Revista Eletrônica de Sistemas de Informação
ISSN 1677-3071

v. 10, n. 2

2011

Sumário

(cc) BY
Este trabalho está licenciado sob uma Licenca Creative Commons Attribution 3.0 .

Revista hospedada em: http://revistas.facecla.com.br/index.php/reinfo
Forma de avaliação: *double blind review*

Esta revista é (e sempre foi) eletrônica para ajudar a proteger o meio ambiente, mas, caso deseje imprimir esse artigo, saiba que ele foi editorado com uma fonte mais ecológica, a *Eco Sans*, que gasta menos tinta.

CATEGORIZAÇÃO AUTOMÁTICA DE MENSAGENS DE *CALL-FOR-PAPERS*

AUTOMATIC CATEGORIZATION OF CALL-FOR-PAPERS MESSAGES

(artigo submetido em julho de 2010)

Daniela Corumba
Tata Consultancy Service (TCS)
daniela.corumba@tcs.com

Hendrik Macedo
Departamento de Computação – Universidade Federal de Sergipe (UFS), Brasil
hendrik@ufs.br

ABSTRACT

Participants of discussion lists receive a great number of messages in their mail boxes. Most of the times, only a small fraction of those messages are useful to the user. An example of such lists is the one used to spread call-for-papers for conferences and scientific journals, which are extremely useful to research groups, professors and students who develop scientific-related activities. The diversity of call-for-papers for different research fields, however, makes the separation of the most relevant ones somewhat difficult. This paper describes an intelligent web service that organizes call-for-papers stored in electronic mail accounts. The service uses a supervised learning technique kNN in order to classify call-for-papers in six major computing areas. Experiments utilizing a test base have shown accuracy of about 89%. An extension of this web service for recommendations of call-for-papers based on automatic information extraction of researchers' Lattes curricula (CNPq) is also presented.

Key-words: recommendation; text mining; categorization; information extraction

RESUMO

Participantes de listas de discussão costumam receber diariamente um grande volume de mensagens em suas caixas de correio eletrônico. Em boa parte dos casos, apenas algumas destas mensagens despertam de fato o interesse do usuário. Um exemplo deste tipo de lista é a assinatura eletrônica de sistemas de chamadas para submissão de artigos científicos a conferências e periódicos (*calls-for-papers*), que são de grande interesse para grupos de pesquisa, professores e estudantes que desenvolvem algum tipo de atividade científica. A diversidade das chamadas entre linhas de pesquisa variadas dificulta o acesso às mais relevantes. Este artigo descreve um serviço Web que organiza de forma inteligente mensagens de *call-for-papers* recebidas em contas de correio eletrônico. O serviço realiza mineração do texto da mensagem e processamento kNN para categorizar os *calls-for-papers* entre seis grandes áreas da computação. Experimentos utilizando uma base de testes mostraram um percentual de acerto na classificação em torno de 89%. Uma extensão desse serviço Web para recomendação de *calls-for-papers* baseado na extração automática de informações de currículos *Lattes* (CNPq) de pesquisadores também é apresentada.

Palavras-chave: recomendação; mineração de texto; categorização; extração de informação

1 INTRODUÇÃO

A correspondência eletrônica (*e-mail*) se tornou uma forma de comunicação bem estabelecida e bastante utilizada em todo o mundo. Correspondências eletrônicas têm sido utilizadas por empresas e organizações para promover e divulgar seus produtos. Não raro, a caixa de entrada de mensagens de um usuário da Internet contém um volume muito grande de mensagens, algumas de seu interesse, outras nem tanto. O fato é que se torna cada vez mais difícil gerenciar sua própria conta de *e-mail*, separando o que é útil do dispensável. Um complicador para este problema são as assinaturas a listas de discussão eletrônica sobre um determinado tema. Neste caso em especial, o volume de mensagens diárias cresce consideravelmente.

Um exemplo importante de listas de discussão eletrônica são as chamadas para submissão de artigos científicos a conferências e periódicos (*call-for-papers*). Estas chamadas são de grande importância para grupos de pesquisa, professores e estudantes que desenvolvem algum tipo de atividade científica. Por meio delas, é possível conhecer veículos apropriados para divulgação de resultados das pesquisas realizadas. Entretanto, mais uma vez, a grande diversidade de *call-for-papers*, que podem variar de acordo com a área de atuação do pesquisador, linha de pesquisa, tipo de veículo de publicação, *deadlines* de submissão, local de realização da conferência, entres outros, dificulta a rápida e correta identificação da mais relevante para o interessado.

Este artigo descreve um serviço Web que organiza automaticamente mensagens de *call-for-papers* recebidas em contas de correio eletrônico. O serviço realiza mineração do texto da mensagem e processamento kNN para classificar os call-for-papers entre seis grandes áreas da computação a saber: *Inteligência Artificial, Banco de Dados, Redes de Computadores, Teoria da Computação, Arquitetura e Engenharia de Software*. Além da categorização, o serviço extrai informações relevantes específicas e recomenda *call-for-papers* baseando-se na similaridade com o currículo *Lattes*[1] do usuário.

Mineração de dados, sistemas de recomendação e extração de informações são técnicas bem consolidadas na literatura científica. Em Cho (2002) são recomendados produtos com base em dados de uso da Web e dados relacionados a compras anteriormente efetuadas pelo usuário, enquanto Zhang e Iyengar (2002) recomendam itens personalizados para compra com base na navegação do usuário no site. Loh *et al* (2004) desenvolveu um sistema de recomendação que identifica os assuntos discutidos em um *web chat* privado e recomenda itens de discussões

1A Plataforma Lattes é um sistema de informação desenvolvido e implantado pelo CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) para gerenciar informações relacionadas a pesquisadores e instituições no Brasil. O sistema armazena atualmente algo em torno de 1.140.000 currículos de pesquisadores, docentes, estudantes e profissionais que atuam em ciência, tecnologia e informação.

anteriores e pessoas que possuem conhecimento significativo no assunto tratado. Outros exemplos de sistemas de recomendação que não utilizam técnicas de mineração são descritos em Shadanand e Maes (1995), Miller *et al.* (2004), Konstan *et al.* (1997) e Krishnamurthy *et al.* (2008). Um bom exemplo de extração de informação relevante pode ser encontrado em Álvarez (2007). O trabalho descreve uma ferramenta para analisar e extrair informações presentes no corpo de artigos científicos, tais como título, resumo e referências bibliográficas. Em Ribeiro *et al.* (2005) é desenvolvida uma ferramenta que identifica automaticamente áreas de interesse de indivíduos a partir de seus currículos Lattes. Em Alves *et al.* (2009) é descrito o "LattesMiner", uma API orientada a objetos para a extração de informações de currículos Lattes e identificação de redes sociais acadêmicas. A ferramenta "GeraLattes" (OLIVEIRA *et al.*, 2004) foi desenvolvida para gerenciar informação a partir da informação operacional de currículos Lattes em formato XML. Loh *et al* (2003) fazem uso de técnicas de mineração de textos para criação do perfil do usuário, o qual é obtido por meio da extração de informações presentes no Currículo Lattes.

O restante do artigo está organizado da seguinte forma. A seção 2 apresenta a fundamentação teórica necessária para entendimento da abordagem adotada: a atividade de mineração de textos e os sistemas de recomendação. Na seção 3, o serviços Web desenvolvido é detalhado em sua arquitetura e cada etapa do processo é explicada de forma minuciosa. A seção 4 descreve os experimentos realizados e traz a discussão dos resultados obtidos. Finalmente, na seção 5, a conclusão do trabalho é apresentada.

2 REFERENCIAL TEÓRICO

2.1 SISTEMAS DE RECOMENDAÇÃO

Um Sistema de Recomendação é um sistema que sugere recomendações ao usuário baseado em suas preferências. Estes sistemas desempenham a mesma função que um amigo ou conhecido, o qual recomenda um restaurante ou um filme. As técnicas usadas em Sistemas de Recomendação podem ser classificadas em basicamente três tipos: (i) baseadas em conteúdo, (ii) filtragem colaborativa e (iii) híbridas. Outras técnicas são mais bem exploradas em Nunes (2009).

A recomendação baseada em conteúdo sugere itens similares aos que o usuário já utilizou anteriormente. Este tipo de recomendação baseia-se na recuperação de informação e faz uso de várias técnicas de extração de informação (CARDIE, 1997). Documentos textuais são recomendados baseados na comparação entre seu conteúdo e o perfil do usuário. Estruturas de dados podem ser criadas extraindo características dos textos dos documentos. Geralmente são utilizados métodos para associar pesos para as palavras presentes na coleção dos documentos. Existem vários métodos alternativos para criar pesos às palavras, como por exemplo, a técnica Tf-idf (WEISS, 2005). Um Sistema de Recomendação é considerado

puramente baseado em conteúdo quando as recomendações que são feitas ao usuário são baseadas apenas em conteúdo dos itens que já foram classificados pelo usuário anteriormente (PAZZANI & BILLSUS, 2007).

Segundo Balabanovic e Shoham (1997), a filtragem colaborativa é bastante diferente da técnica de recomendação baseada em conteúdo. Ao invés de recomendar itens por serem similares aos itens que o usuário utilizou anteriormente, ela recomenda itens que usuários similares utilizaram anteriormente. Dessa forma, é encontrado um conjunto de usuários cujos gostos são similares aos gostos do usuário em questão, os quais são denominados de vizinhos mais próximos. A pontuação dos itens que não foram utilizados pelo usuário é dada pela combinação das pontuações conhecidas do conjunto de vizinhos mais próximos. Em um Sistema de Recomendação puramente colaborativo, as recomendações para os usuários são feitas baseadas nas similaridades com outros usuários (NUNES, 2009). Fazendo uso das recomendações de outros usuários, é possível lidar com qualquer tipo de conteúdo e receber itens com conteúdo diferente daqueles vistos anteriormente.

Diversos sistemas de recomendação usam a abordagem híbrida combinando técnicas dos métodos colaborativos e baseados em conteúdo, o que ajuda a melhorar o desempenho da recomendação. Existem diferentes formas para combinar as duas técnicas e gerar o método híbrido, segundo Adomavicius e Tuzhilin (2005), podem ser exploradas as seguintes possibilidades: (1) implementar o método colaborativo e o método baseado em conteúdo separadamente e depois comparar predições, (2) incorporar algumas características do método colaborativo no método baseado em conteúdo e vice-versa, e (3) construir um modelo unificado geral que incorpora características dos métodos colaborativo e baseado em conteúdo.

2.2 MINERAÇÃO DE TEXTOS

A mineração de textos, tipo especial de mineração de dados, é uma tecnologia emergente para análise de grandes coleções de documentos não estruturados, visando à extração de padrões ou conhecimentos interessantes e não triviais (VISA, 2001).

Assim como a mineração de dados convencional, a mineração de texto possui etapas inerentes ao processo tal como o pré-processamento. Uma vez montada a base com os textos a serem minerados, faz-se necessário converter cada documento para um formato compreensível por algoritmos computacionais. Essa tarefa é atribuída à etapa de pré-processamento.

Na literatura científica, existem duas abordagens principais para o processamento de texto em língua natural: uma abordagem estatística, onde se faz uso de modelos vetoriais para representação e uma abordagem baseada em *parsing* sintático. Para nossa aplicação, onde não há necessidade de compreensão semântica do texto, o modelo vetorial é extremamente adequado e eficiente.

O modelo vetorial vale-se da geometria para representação dos documentos. Introduzido por Salton *et al.* (1975), esse modelo foi desenvolvido para ser utilizado em um sistema de recuperação de informações chamado SMART. Segundo o modelo vetorial, cada documento é representado por um vetor de termos e cada termo possui um peso associado que indica seu grau de importância no documento (SALTON *et al.*, 1975). Em outras palavras, cada documento possui um vetor associado, o qual é constituído por elementos organizados por uma tupla de valores da forma: $d_j = \{w_{1j}, ..., w_{ij}\}$, onde d_j representa um documento e w_{ij} representa um peso associado a cada termo indexado de um conjunto de t termos do documento.

Cada elemento do vetor de termos é considerado uma coordenada dimensional. Desta forma, os documentos podem ser colocados em um espaço euclideano de *n* dimensões (onde *n* é o número de termos) e a posição do documento em cada dimensão é dada pelo peso do termo associado a aquela dimensão.

No modelo do espaço vetorial, as consultas também são representadas por vetores. Assim, os vetores dos documentos podem ser comparados ao vetor da consulta e o grau de similaridade entre eles pode ser calculado. Os documentos mais similares (aqueles que apresentarem os vetores mais próximos ao vetor da consulta) são considerados relevantes, retornando como resposta para o usuário. Além disso, os documentos que apresentarem os vetores mais próximos são considerados similares entre si.

De uma forma geral, nem todos os termos que compõe um documento são relevantes quando se almeja extrair informações de alto nível. Assim, para compor o vetor de termos de um texto, é necessário identificar as palavras de forte conteúdo semântico, selecionando apenas aquelas que realmente carregam em si um significado relevante para o propósito em questão.

A atividade de *extração de termos* de um documento é composta de vários passos, contribuindo todos eles para o propósito final (HIEMSTRA & JONG, 2001). São eles:

1. Análise léxica: nem sempre o documento original se encontra em formato puramente textual. Em função disso, é necessário converter estes formatos, eliminando quaisquer atributos de formatação de apresentação para um formato padronizado.
2. Conversão de caracteres para maiúsculo ou minúsculo: este procedimento possibilita que palavras iguais, porém escritas com algum caractere em formato maiúsculo ou minúsculo diferente possam ser interpretadas como o mesmo termo.
3. Uso de uma lista de palavras a serem desconsideradas: comumente chamadas de *stopwords*. Essa lista consiste em uma relação de palavras que não têm conteúdo semântico significativo (como por exemplo, preposições, conjunções, artigos, nume-

rais e outros) e, por consequência, não são relevantes na análise do texto.

4. Normalização morfológica: com o objetivo de agrupar termos com o mesmo significado conceitual, a exemplo das palavras computar e computação, pode ser aplicado um algoritmo de conversão de termos em radicais. No caso exemplificado, as palavras possuem o mesmo radical *comput*, e, por isto, podem ser resumidas a este termo.
5. Normalização de sinônimos: palavras de mesmo significado podem ser reduzidas a um mesmo termo, a exemplo da sigla IA e a composição Inteligência Artificial, as quais possuem o mesmo significado.

O processo de associar valores numéricos a cada termo previamente extraído é conhecido como *atribuição de pesos*. Em geral, a determinação do peso de um termo em um documento pode ser efetuada com base em dois critérios (WEISS, 2005):

(1) quanto mais vezes um termo aparece no documento, mais relevante ele é para o tópico do documento;

(2) quanto mais vezes um termo ocorre dentre todos os documentos de uma coleção, menos importante ele é para diferenciar os documentos.

Partindo deste princípio: duas são as abordagens passíveis de aplicação para cálculo de pesos:

- binária ou booleana - os valores 0 e 1 são utilizados para representar, respectivamente, a ausência ou presença de um termo no documento.
- numérica – baseia-se em técnicas estatísticas relacionadas à frequência dos termos no documento.

Os pesos numéricos podem ser representados conforme as medidas a seguir:

Frequência dos termos (*term frequency* - tf): Método simples, que consiste no cálculo do número de vezes que um termo *wi* ocorre em um documento *d*. Esse método está baseado na premissa de que a frequência do termo no documento fornece informação útil sobre a importância desse termo para o documento.

Frequência do documento (*document frequency* - DF): é o número de documentos no qual o termo w_i ocorre pelo menos uma vez.

Frequência inversa do documento (*inverse document frequency* - idf): define a importância de um termo dentre um conjunto de documentos. Quanto maior for este índice, mais representativo é o termo para o documento que o possui. A fórmula para cálculo da *idf* é:

doi:10.5329/RESI.2011.1002006

$$\mathrm{idf_i} = \log \frac{|D|}{|\{d : t_i \in d\}|}$$

Onde |D| representa o total de documentos e $|\{d : t_i \in d\}|$ representa o número de documentos em que o termo t_i aparece.

tf-idf: combina a frequência de um termo com sua frequência inversa de documento, a fim de obter um índice maior de sua representatividade. A fórmula para cálculo do peso *tf-idf* é:

$$(\mathrm{tf\text{-}idf})_{i,j} = \mathrm{tf_{i,j}} \times \mathrm{idf_i}$$

3 SERVIÇO WEB INTELIGENTE

O serviço desenvolvido neste trabalho apresenta uma forma de classificar chamadas de trabalho (*call-for-papers*), categorizando-as em subáreas da Computação (Teoria da Computação, Inteligência Artificial, Redes de Computadores, Banco de Dados, Arquitetura de Computadores e Engenharia de Software). O serviço também dispõe de funcionalidades que provêm a qualidade do veículo de publicação e a data em que os trabalhos devem ser submetidos para conferência ou periódico, além de recomendar chamadas de trabalho ao pesquisador. Dessa forma, esse serviço Web combina técnicas de mineração de textos e sistemas de recomendação.

As principais funcionalidades do serviço Web são:

- listar todas as chamadas de trabalho em ordem crescente de data em que os trabalhos devem ser submetidos para o evento;
- listar todas as chamadas de trabalho de uma subárea previamente definida e escolhida pelo usuário;
- listar todas as chamadas de trabalho em ordem crescente de qualidade do veículo de publicação;
- recomendar uma chamada de trabalho considerando o currículo Lattes de um pesquisador.

Estas funcionalidades serão discutidas em mais detalhes nas próximas seções.

3.1 ARQUITETURA DO SERVIÇO

A Figura 1 traz um diagrama de componentes referente à arquitetura do serviço Web, salientando os componentes considerados mais relevantes para o entendimento do sistema.

Dentro do Servidor de Email, encontra-se o componente Conta Email, que é responsável pela coleta de mensagens presentes na caixa de entrada de uma conta de e-mail. A comunicação entre o Conta Email e o RecebedorCFP se dá por meio do protocolo *IMAP*, que é um protocolo de gerenciamento de correio eletrônico. O componente RecebedorCFP repassa as mensagens de e-mail ao componente ClassificadorCFP, o qual é responsável pelas etapas de extração de informação e classificação da

mensagem em uma subárea. Dessa forma, o banco de dados conterá as mensagens de e-mail já classificadas e com as informações, *deadline* e avaliação *Qualis*, necessárias ao serviço web extraídas. O banco de dados é responsável pela alimentação do serviço web listando os *call-for-papers* de acordo com as preferências do usuário.

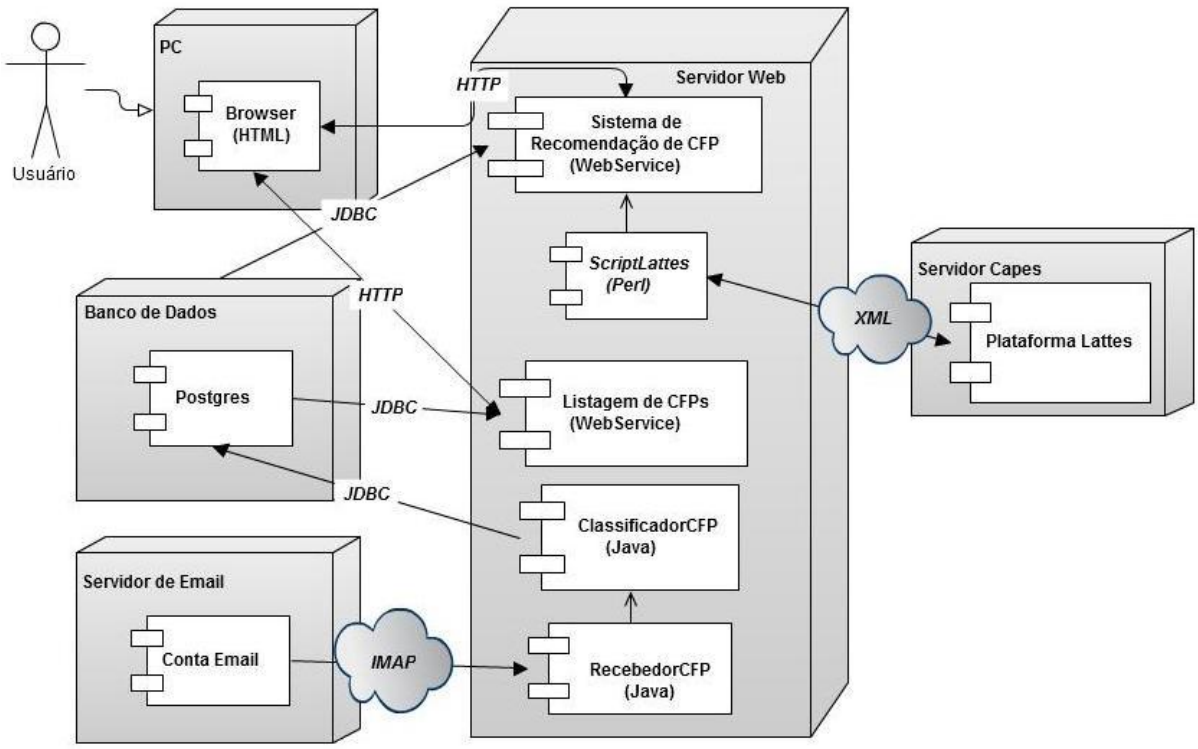

Figura 1. Diagrama de componentes que representa a arquitetura do sistema

Fonte: elaborado pelo autor

Adicionalmente a essa funcionalidade do serviço Web, existe o sistema de recomendação implementado neste trabalho. O usuário interage com o componente sistema de recomendação de CFP passando um ID. Esse ID é utilizado pelo componente ScriptLattes, a fim de obter informações sobre o Currículo Lattes do usuário. Neste momento, o ScriptLattes acessa a Plataforma Lattes fazendo uso das informações disponíveis pelo site da Capes.

3.2 COLETA DE DADOS

Atualmente a base de dados dispõe de 576 *calls-for-papers* obtidas a partir de listas de discussão da área de Computação, em particular, a lista da SBC (Sociedade Brasileira de Computação), a *sbc-l.* Para utilização dos *call-for-papers* durante a execução do programa, foi utilizada a API JavaMail (JAVAMAIL, 2010), que dá suporte à implementação de funções de e-mail em aplicações Java.

A *sbc-l* é um veículo de discussão para pesquisadores, profissionais e estudantes de computação do país. Os assinantes da lista recebem

doi:10.5329/RESI.2011.1002006

diariamente não apenas mensagens de *call-for-papers*. Diversos outros tipos de mensagens circulam frequentemente. Para o trabalho, apenas as mensagens de *call-for-papers* são relevantes. Desta forma, foi aplicada uma etapa de mineração do título das mensagens de forma a filtrá-las apropriadamente. A mineração se deu por meio da exclusão de mensagens cujos títulos não estivessem relacionados a palavras que remetem a chamadas de trabalho tais como "CFP", "CFPs", "*Call for papers*", "Chamada de trabalhos", "Chamada de artigos" e "Envio de trabalhos".

3.3 EXTRAÇÃO DA DATA DE SUBMISSÃO

O problema da extração da data de submissão de trabalhos científicos (*deadline*) consiste em encontrar um formato que combine com o padrão utilizado nas chamadas de trabalho. Pode-se definir o problema mais formalmente da seguinte forma.

Seja *D* uma grande base de dados que contém informações semi-estruturadas, como é o caso de chamadas de trabalho e seja $R = r_1, ..., r_n$ a relação que se pretende alcançar, cada tupla *t* de *R* ocorre uma ou mais vezes em *D*. Cada uma dessas ocorrências contém todos os campos presentes em *t,* representados como um conjunto de palavras que determinarão a data de submissão.

Neste trabalho a relação que se pretende alcançar é (dia, mês, ano) referente à data. Claramente, essa relação não é bem definida dentro de uma chamada de trabalho, principalmente levando-se em consideração que o dia pode aparecer antes do mês ou depois do mês, letras maiúsculas e minúsculas, números escritos por extenso ou representados por numerais, presença ou não de separadores, como vírgulas e barras. Dessa forma, observou-se a necessidade de definir um padrão para extrair esse tipo de informação.

Intuitivamente um padrão combina um formato particular de ocorrências de tuplas da relação que se pretende alcançar. Supondo condições ideais, o padrão é suficientemente específico para não encontrar nenhuma tupla que não deveria estar na relação, entretanto alguns falsos-positivos podem ocorrer. Assim, os padrões devem conter várias representações. Neste trabalho foram utilizadas *expressões regulares* para que fossem extraídos os *deadlines*.

A Figura 2 mostra um trecho de uma chamada de trabalho que contém a data de submissão.

Datas importantes:
A data limite para apresentação de propostas é 17 de agosto de 2010 às 23:59, horário de Brasília.
A notificação de aceitação ocorrerá até o dia 8 de setembro de 2010.
A versão final das apresentações deverá ser enviada até o dia 30 de setembro de 2010.

Figura 2. Trecho de uma chamada de trabalho e sua data de submissão
Fonte: elaborado pelo autor

Dessa forma, a *expressão regular* utilizada neste trabalho foi definida primeiramente com uma tupla (*início*, *dia*, *mês*, *ano*), sendo *início* um valor booleano que indica a partir de que momento deve-se tentar achar o padrão. Assim, o padrão só será procurado a partir do momento em que o conjunto de palavras "Datas importantes" ocorrer no documento e será encontrado caso ocorra a seguinte expressão regular:

dia, mês, ano

O *dia* é restrito a [1-31], o *mês* é restrito a [jan, fev, mar, abr, mai, jun, jul, ago, set, out, nov, dez], podendo ocorrer letras iniciais maiúsculas ou minúsculas, assim como todas as letras maiúsculas ou minúsculas. Foi definido também um valor de 6 caracteres que deve ser desconsiderado para o casamento de padrões após encontrar os valores de *mês*, assim estarão sendo considerados também os nomes dos meses escritos até a última letra, como por exemplo, "dezembro". O mês pode também assumir valores inteiros, sendo eles os numerais equivalentes a cada mês do calendário. O *ano* será encontrado após a ocorrência do *mês* e serão considerados anos compostos por 2 ou 4 dígitos.

Analogamente, foi definida uma expressão regular para as datas presentes em chamadas de trabalho em inglês, respeitando as normas gramaticais da língua. Entretanto, nesse momento, o padrão só é procurado após a ocorrência do conjunto de caracteres "*Important dates*". Assim, para a língua inglesa, a expressão regular definida foi:

mês, dia, ano

O *mês* é restrito as 3 primeiras iniciais dos meses em inglês, sendo a primeira letra sempre maiúscula, e considerando-se uma margem de 6 caracteres para o caso de haver o nome do mês inteiro, como por exemplo, "*November*". O *dia* é restrito a [1-31], e foram considerados dois caracteres após o dia, caso a ocorrência do dia seja, por exemplo, "2nd". O *ano* segue a mesma regra do português.

Não foram considerados os casos em que as datas são expressas apenas por números e com um separador para cada atributo da tupla, uma vez que a ocorrência de datas escritas nesse formato não é predominante nas chamadas de trabalho utilizadas.

3.4 EXTRAÇÃO DA QUALIDADE DO VEÍCULO DE PUBLICAÇÃO

A credibilidade nas conferências e periódicos nacionais e internacionais e, consequentemente, do grupo de pesquisadores que neles atuam, é avaliada pela comunidade científica principalmente por meio da qualidade de sua produção científica e tecnológica. A qualidade científica de uma conferência científica é dada com base no documento *Qualis* da *Capes*.

Entretanto, o gerenciamento e a manutenção das informações atualizadas sobre a qualidade do veículo de publicação requerem um esforço de coleta de dados, assim como demandam uma quantidade de tempo considerável para ser realizada manualmente pelos pesquisadores. Assim,

o objetivo da extração da qualidade de uma conferência é facilitar o acesso a essa informação.

Neste trabalho, o processo de extração da qualidade do veículo de publicação foi dado através do *documento de área da Capes*, disponibilizado a cada três anos, onde está disponível a lista completa das publicações qualificadas em veículos internacionais e nacionais. O documento possui uma lista contendo a sigla da conferência, o nome e o estrato do *Qualis*, que pode assumir 8 valores variando entre A1, A2, B1, B2, B3, B4, B5 e C, sendo A1 o estrato de maior valor e C o de menor valor. A Figura 3 mostra uma parte do documento analisado.

1003	SBBD	Simpósio Brasileiro de Bancos de Dados	B3
1004	SBCCI	Symposium on Integrated Circuits and Systems Design	B3
1005	SBES	Simpósio Brasileiro de Engenharia de Software	B3
1006	SBIA	Brazilian Symposium on Artificial Intelligence	B3
1007	SBLP	Simpósio Brasileiro de Linguagens de Programação	B3
1008	SBMF	Simpósio Brasileiro de Métodos Formais	B3

Figura 3. Trecho ilustrativo de estratos do *Qualis* da Computação.
Fonte: adaptado do documento de área da Capes (Computação), triênio 2007-2009

Para extração do *Qualis* foram analisadas as siglas das conferências que estavam na base de dados, uma vez que as chamadas de trabalho contêm tanto a sigla quanto o nome da conferência. Assim, a extração consistiu em casar a sigla dentro do corpo da chamada de trabalho e, uma vez encontrada a sigla, retornar o valor do estrato do *Qualis* associado a ela. Os *Qualis* dos periódicos são obtidos no *site* da Capes (Capes, 2010), portanto eles não foram considerados neste trabalho, até o presente momento.

3.5 CATEGORIZAÇÃO DE UMA CHAMADA DE TRABALHO EM UMA SUBÁREA

A categorização considera a subárea da Computação (Teoria da Computação, Inteligência Artificial, Redes de Computadores, Banco de Dados, Arquitetura de Computadores e Engenharia de Software) à qual a chamada se refere. Dessa forma, é possível restringir o domínio de um *call-for-paper* a uma das seis subáreas pré-definidas no trabalho, a fim de minimizar o tempo gasto procurando por chamadas de trabalho referentes a uma determinada área do pesquisador, além de gerenciar o grande volume de *calls-for-papers* recebido.

Para classificar uma chamada de trabalho em uma subárea da Computação foram realizados os seguintes passos principais para a mineração dos textos, como mostra a Figura 4.

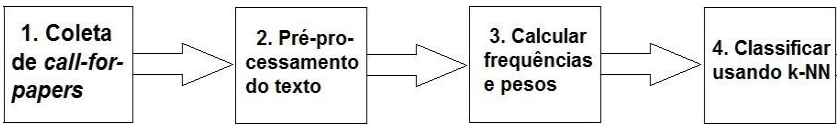

Figura 4. Etapas realizadas para a classificação dos textos das mensagens de *call-for-papers.*
Fonte: elaborado pelo autor

O passo 1 inclui a etapa da coleta de dados eliminando quaisquer mensagens enviadas para a lista de discussão da SBC que não se refiram a chamadas de trabalho. Após coletar os dados, eles são gravados em um Banco de Dados para utilização nos passos seguintes. O banco de dados apenas guarda textos referentes a *calls-for-papers*. Inicialmente foi necessário um trabalho manual para classificação de *calls-for-papers* que fariam parte da base de treinamento do classificador. A Tabela 1 traz a quantidade de *calls-for-papers* presentes na base de treinamento montada.

Tabela 1. Número de chamadas de trabalhos presentes na base de treinamento, classificadas por subárea da Computação.

Subárea da Computação	Quantidade
Arquitetura	15
Engenharia de software	11
Inteligência artificial	10
Banco de dados	15
Redes	14
Teoria	11

Fonte: elaborada pelo autor.

No passo 2 o sistema faz uso das técnicas de pré-processamento de texto a fim de preparar o texto que será classificado. Esse passo inclui a etapa de eliminação de *stopwords*.

No passo 3, o sistema usa técnicas de Mineração de Textos para atribuir pesos aos atributos presentes no corpo das mensagens, de forma que palavras importantes para a coleção terão valores maiores que aquelas consideradas menos relevantes para a coleção. Para conclusão deste passo utilizou-se a técnica de *Tf-Idf*, descrita na seção 2.

Por fim, no passo 4, o sistema está pronto para realizar a classificação de um novo *call-for-paper*. Nesse momento, é passado um *call-for-paper* que não está presente na base de dados e ele é classificado dentro de uma das 6 subáreas pré-definidas. A técnica utilizada para classificação foi o k-NN.

A técnica dos *k-Nearest Neighbours* pode ser definida da seguinte forma. Considere um conjunto D de tuplas de treinamento. Cada elemento de D é uma tupla (x_1, x_2, ..., x_n, c), onde c é a classe à qual pertence a tupla (x_1, ..., x_n). A tupla (x_1, ..., x_n) pode ser vista como um ponto no espaço n-dimensional.

Seja Y = (y_1, ..., y_n) uma nova tupla, ainda não classificada. A fim de classificá-la, calculam-se as distâncias de Y a todas as tuplas de treinamento e considera-se a tupla de treinamento mais próxima de Y. A distância entre duas tuplas pode ser calculada através da distância euclidiana.

$$d(X,Y) = \sqrt{\Sigma_{i=1}^{n}(x_i - y_i)^2}$$

Geralmente é preciso normalizar os valores de cada atributo para que todos estejam em um mesmo intervalo de variação, não havendo muita discrepância entre os valores dos diferentes atributos, o que poderia influir tendenciosamente no cálculo da distância.

3.6 RECOMENDAÇÃO BASEADA NA EXTRAÇÃO DE INFORMAÇÕES DE CURRÍCULOS LATTES

O Sistema de Recomendação foi desenvolvido fazendo uso da ferramenta "scriptLattes" (SCRIPTLATTES, 2010). Para utilização dessa ferramenta, é necessária a criação de um arquivo no formato de texto, contendo a informação correspondente ao pesquisador a quem a recomendação será destinada. Esse arquivo deve conter o código de 16 dígitos que o CNPq utiliza como ID para cada currículo Lattes, podendo conter informações tais como o nome completo do pesquisador e o ano a partir do qual se pretende obter as informações. Dessa forma, o Sistema de Recomendação recebe como entrada o ID do Lattes do usuário, que acessa o sistema Currículo Lattes por meio do scriptLattes e retorna arquivos HTML contendo as publicações do pesquisador. Esses arquivos HTML são analisados procurando-se identificar similaridades entre as publicações do pesquisador e os *calls-for-papers* presentes na base de treinamento.

4 EXPERIMENTOS E RESULTADOS

A coleta de mensagens de *call-for-papers* foi realizada durante um período de 3 meses compreendido entre 28/04/2010 e 28/07/2010, perfazendo um total de 921 mensagens para a área de computação.

Nesta seção, visa-se a observar os resultados independentes da (i) filtragem de chamadas de trabalho executada pelo componente "RecebedorCFP"; (ii) a detecção de *deadlines* contidos nos corpos dessas mensagens e, por fim, (iii) a categorização desses e-mails nas 6 subáreas.

A Tabela 2 mostra os resultados obtidos a partir da filtragem dos e-mails que chegaram às listas de discussão. É possível perceber um acerto de quase 94%, considerando tanto o número de falsos-positivos (*calls-for-papers* não-válidos considerados válidos pelo sistema) quanto de falsos-negativos (*calls-for-papers* válidos considerados não-válidos pelo sistema).

Tabela 2. Demonstrativo de resultados das chamadas de trabalho.

	Número de acertos	Número de erros	% acertos	% erros
CFP encontrado	566	11	98,1	1,9
CFP não encontrado	299	45	86,9	13,1
Total	865	56	93,9	6,1

Fonte: elaborada pelo autor.

Após filtragem das mensagens para chegar-se a uma lista de supostas chamadas de trabalhos, tem-se um universo de 576 mensagens. A partir desta etapa, o componente "RecebedorCFP" captura aqueles *deadlines* que, em sua quase totalidade, já estão presentes nas chamadas de trabalho, como mostra a Tabela 3.

Tabela 3. Demonstrativo de resultados da extração do *deadline*.

	Número de acertos	Número de erros	% acertos	% erros
Deadline encontrado	311	45	87,4	12,6
Deadline não encontrado	183	37	83,2	16,8
Total	494	82	85,8	14,2

Fonte: elaborada pelo autor.

O sistema conseguiu obter uma taxa de 85,8% de acertos ao capturar o *deadline* da chamada de trabalho quando este existia de fato e de não capturar quando não existia. A taxa de erro em *deadlines* não encontrados (16,8%) encontra-se elevada devido à quantidade de chamadas de trabalho analisadas que continham anexos, o que impossibilitou encontrar a informação procurada no corpo da mensagem.

Em relação à extração do *Qualis* da Capes (Tabela 4), os resultados obtidos foram satisfatórios, visto que 91,5% dos Qualis conseguiram ser capturados. O somatório dos falso-positivos e falso-negativos ficou em torno dos 8,5%.

Tabela 4. Demonstrativo de resultados da extração do *Qualis.*

	Número de acertos	Número de erros	% acertos	% erros
Qualis encontrado	362	27	93,1	6,9
Qualis não encontrado	165	22	88,3	11,7
Total	527	49	91,5	8,5

Fonte: elaborada pelo autor.

Para a categorização das chamadas de trabalho usando a técnica de k-*Nearest-Neighbor*, foi utilizado o valor de k = 7. Com esse valor, o classificador obteve sucesso em suas classificações ao categorizar *calls-for-papers* de determinadas subáreas, como mostra a Tabela 5.

Tabela 5. Demonstrativo de resultados da classificação dos call-for-papers.

	Número de acertos	Número de erros	% acertos	% erros
Inteligência artificial	87	24	78,4	21,6
Banco de dados	15	12	55,6	44,4
Redes	27	9	75,0	25,0
Teoria da computação	30	12	71,4	28,6
Arquitetura	21	9	70,0	30,0
Engenharia de software	21	6	77,8	22,2
Total	201	72	73,6	26,4

Fonte: elaborada pelo autor.

A melhor classificação obtida foi em Inteligência Artificial com o maior número de acertos. Por outro lado, o menor número de acertos aconteceu na área de Banco de Dados. Essa diferença está relacionada a palavras com alta relevância na área de Banco de Dados que também são utilizadas em outras subáreas, como por exemplo, Redes e Arquitetura. Os termos relevantes de Inteligência Artificial distinguem-se dos demais termos equivalentes a outras subáreas. De uma maneira geral, a classificação obteve sucesso com 73,6% dos *calls-for-papers* sendo classificados corretamente. A fim de aumentar a taxa de acertos, uma alternativa é fazer uso de uma base de dados maior, com *calls-for-papers* significativos em cada uma das subáreas.

Considerando o Sistema de Recomendação baseado na extração de informações de Currículos Lattes, é importante salientar que a Plataforma Lattes é utilizada principalmente por brasileiros, assim a grande maioria dos currículos disponíveis está em português. Em virtude do pequeno número de *calls-for-papers* escritos em português utilizados neste trabalho, a recomendação obteve resultados inferiores às demais etapas. Entretanto, o sistema de recomendação seguramente terá melhores resultados à medida que a base de dados aumente, principalmente fazendo uso de chamadas de trabalho escritas na língua portuguesa.

5 CONCLUSÃO

Este trabalho descreve um serviço Web de classificação automática de chamadas de trabalhos científicos (*calls-for-papers*) realizadas através de mensagens eletrônicas de e-mail. A classificação leva em consideração as grandes subáreas da computação, deadlines de submissão e avaliação Qualis da Capes. Um sistema de recomendação de chamadas de trabalho também foi desenvolvido. O sistema utiliza o currículo Lattes do pesquisador para recomendar *call-for-papers* relacionados com assuntos de pesquisa do orientador, refletidas em diversas seções do currículo, como listas de publicações e orientações.

Fazendo uso de um modelo vetorial de representação dos textos e o algoritmo de classificação k-NN, o serviço Web apresentou taxas de acerto em torno de 73% na classificação de acordo com a subárea, 86% de acordo com o deadline e 91% na classificação que levou em consideração a avaliação Qualis da Capes. Uma vez que a quantidade de chamadas de trabalhos para periódicos e, em particular, para eventos científicos em computação (conferências, simpósios, encontros e *workshops*) se mostra bem extensa e frequente e a filtragem manual dos melhores veículos para uma eventual submissão é consequentemente uma tarefa árdua, estes valores percentuais se mostram bem relevantes.

A grande dificuldade encontrada, para que uma melhor avaliação da efetividade do sistema de recomendação desenvolvido fosse feita, foi a falta de uma base maior de mensagens de *call-for-papers* escritos em

português, uma vez que a grande maioria dos currículos Lattes disponíveis não apresentam sua versão em inglês.

Algumas modificações podem ser realizadas a fim de melhorar o desempenho do sistema. Dessa forma, como trabalhos futuros, visa-se realizar uma categorização múltipla das subáreas, visto que foi observado que uma quantidade considerável de *call-for-papers* abrangem mais de uma subárea da Computação.

Outro ponto a ser considerado é a predominância de *calls-for-papers* escritos em inglês, o que dificulta a classificação daqueles escritos em outra língua. Assim, capturar e-mails de outras fontes a fim de diversificar as línguas torna-se um trabalho futuro promissor.

6 REFERÊNCIAS

ADOMAVICIUS, G., TUZHILIN, A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions*. IEEE Transactions on knowledge and data engineering*, p. 734-749, 2005. doi:10.1109/TKDE.2005.99

ÁLVAREZ, A. C. Extração de informação de artigos científicos: uma abordagem baseada em indução de regras de etiquetagem. Dissertação de Mestrado, Instituto de Ciências Matemáticas e de Computação (ICMC), USP, 2007.

ALVES, A. D., YANASSE, H. H., SOMA, N. Y. Extração de Informação na plataforma Lattes para identificação de redes sociais acadêmicas. Workshop dos Cursos de Computação Aplicada do INPE, 9., São Jose dos Campos. *Anais...*, INPE, 2009.

BALABANOVIC, M., SHOHAM, Y. Fab: content-based, collaborative recommendation. *Communications of the ACM*, v. 40, n. 3, p. 72, 1997. doi:10.1145/245108.245124

LOH, S. BORGES, T., RIBEIRO JR, L. C., PILTCHER, G., LITCHNOW, D., KICKHOFEL, R. B., GOUVEIA, C., GARIN, R. S. Identificação automática de expertise analisando currículos no formato Lattes. Simpósio Brasileiro de Sistemas de Informação, 1., Porto Alegre. *Anais*... Porto Alegre: SBSI, 2004.

CAPES. Coordenação de Aperfeiçoamento de Pessoal de Nível Superior. s.d. Disponível em: http://www.capes.gov.br/. Acesso em: 10/07/2010.

CARDIE, C. Empirical methods in information extraction. *AI Magazine.* v. 18, n. 4, p. 65, 1997.

CHO, Y. H., KIM, J. K., KIM, S. H. A personalized recommender system based on web usage mining and decision tree induction, *Expert Systems with Applications*, v. 23, n. 3, p. 329–342, 2002. doi:10.1016/S0957-4174(02)00052-0

HIEMSTRA, D., JONG, F. Statistical language models and information retrieval: natural language processing really meets retrieval. *Glot International*, v. 5, n. 8, 2001.

JAVAMAIL. Java mail API 1.4.3. Sun Microsystems, Inc. 2009. Disponível em: http://java.sun.com/products/javamail/index.jsp. Acesso em: 10 jul 2010.

KONSTAN, J. A.; MILLER, B. N.; MALTZ, D.; HERLOCKER, J. L.; GORDON, L. R.; RIEDL, J. GroupLens: applying collaborative filtering to Usenet news. *Communications of the ACM*, v. 40, n. 3, p. 87, 1997. doi:10.1145/245108.245126

KRISHNAMURTHY, B., GILL, P., ARLITT, M. A few chirps about twitter. Workshop on Online Social Networks, 1., p. 19-24. *Proceedings...* ACM, 2008. doi:10.1145/1397735.1397741

MILLER, B. N., ALBERT, I., LAM, S. K., KONSTAN, J. A., RIEDL, J. Movielens unplugged: experiences with a recommender system on four mobile devices. *People and Computers*, p. 263-280, 2004.

NUNES, M. *Recommender systems based on personality traits*: could human psychological aspects influence the computer decision-making process? Berlin: VDM Verlag Dr. Muller, 2009.

OLIVEIRA, E., BERMEJO, P. H. de S., KERN, V. M. GeraLattes: extração de informação gerencial de currículos de pesquisadores usando XML. Workshop de Computação da Região Sul (WorkCompSul 2004), 1., Florianópolis. *Anais...* UNISUL, 2004.

PAZZANI, M., BILLSUS, D. Content-based recommendation systems. The adaptive web, p. 325-341. Springer-Verlag, 2007.

RIBEIRO, L.; BORGES, T.; LICHTNOW, D.; LOH, S.; SALDAÑA, R. Identificação de áreas de interesse a partir de extração de informações de currículos Lattes/XML. In: I Escola Regional de Banco de Dados, Porto Alegre, 2005.

SALTON, G., WONG, A., YANG, C. S. A vector space model for automatic indexing. *Communications of the ACM*, v. 18, 1975.

SCRIPTLATTES. ScriptLattes. s.d. Disponível em: http://scriptlattes.sourceforge.net/. Acesso em: 10/07/2010.

SHADANAND, U., MAES, P. Social information filtering: algorithms for automating “word of mouth”. SIGCHI Conference on Human Factors in Computing Systems, p. 210-217. *Proceedings...* ACM Press/Addison-Wesley Publishing Co., 1995.

VISA, A. Technology of text mining. International Workshop on Machine Learning and Data Mining in Pattern Recognition (MLDM 2001), 2., Leipzig, *Proceedings...* MLDM, 2001.

WEINBERGER, K., SAUL, L. Distance metric learning for large margin nearest neighbor classification. *The Journal of Machine Learning Research*, v. 10, p. 207-244, 2009.

WEISS, S. *Text mining*: predictive methods for analyzing unstructured information. Springer-Verlag New York Inc., 2005.

ZHANG, T., IYENGAR, V. Recommender systems using linear classifiers. *Journal of Machine Learning Research*, v. 2, p. 313-334, 2002.