Avaliação de Estratégias para Recuperação de Fontes em Sistemas de Detecção de Plágio

Vicente Bissoli Sessa
Departamento de Informática
Universidade Federal do Espírito Santo
Vitória, Espírito Santo. 29075-910
Email: vicente.sessa@ufes.br
Telefone: +55 (27) 4009-2255

Claudine Badue Santos Gonçalves
Departamento de Informática
Universidade Federal do Espírito Santo
Vitória, Espírito Santo. 29075-910
Email: claudinebadue@lcad.inf.ufes.br
Telefone: +55 (27) 4009-2138

***Resumo*—Neste trabalho, foram investigadas várias estratégias para a etapa de recuperação de documentos fontes em sistemas de detecção extrínseca de plágio. Para avaliar o desempenho desta etapa foi usado um arcabouço desenvolvido pelos organizadores do *International Workshop on Plagiarism Analysis* (PAN), capaz de simular todas as fases envolvidas neste processo. Foram usadas ainda, três métricas: precisão, revocação e F1. Os resultados experimentais mostraram que o desempenho da estratégia que combina o método de seleção de termos menos frequentes, o método de filtragem dos resultados Proximity e o limiar de poda igual a 1 superou o desempenho das demais estratégias avaliadas em termos da métrica F1.**

***Palavras-Chave*—Processamento de Linguagem Natural, Detecção Automática de Plágio, International Workshop on Plagiarism Analysis (PAN), Recuperação de Documentos Fontes, Recuperação de Informação.**

I. Introdução

Métodos para detecção de reutilização de texto são fundamentais em sistemas de detecção de plágio e proteção de direitos autorais. Eles devem ser capazes de lidar com diversos tipos de reutilização de texto, indo desde cópias diretas e citações até paráfrases, traduções e resumos.

Estudos na área datam do início dos anos 90 [1]. Desde então, pesquisadores têm se preocupado com a grande quantidade de conteúdo plagiado encontrado na rede e em meios acadêmicos. Acredita-se que 30% de todo conteúdo disponível na Web seja duplicado, incluindo cópias, sítios espelhados e versionamentos de um mesmo documento [2], [3].

Embora seja um problema antigo, a detecção automática de plágio ainda é uma tarefa desafiadora. Apenas recentemente ferramentas apropriadas para avaliar sistemas de detecção de plágio tornaram-se públicas [4]. Além disso, a estimativa de parâmetros ideais para cada etapa destes sistemas constitui uma área pouco explorada.

Neste trabalho, foram investigadas várias combinações de técnicas de recuperação de informação para a etapa de recuperação de documentos fontes em sistemas de detecção extrínseca de plágio. Para avaliar o desempenho desta etapa, foi usada um arcabouço desenvolvido pelos organizadores do *International Workshop on Plagiarism Analysis* (PAN). Este arcabouço é composto essencialmente por uma base de dados, um mecanismo de busca nos moldes de soluções comerciais e uma coleção de documentos plagiados escritos manualmente. Foram usadas também três métricas de RI: precisão, revocação e F1.

Até onde pôde-se examinar na literatura, a combinação de técnicas empregadas é única e os resultados obtidos são satisfatórios.

Este artigo é organizado como se segue. Após esta introdução, na Seção 2, é apresentado o problema de detecção automática de plágio, incluindo a terminologia, a etapa de recuperação de fontes — foco deste trabalho — e a etapa de identificação de passagens. Na Seção 3, é descrita a metodologia experimental, incluindo o arcabouço PAN, as estratégias para a etapa de recuperação de fontes e as métricas. Na Seção 4, são analisados os resultados experimentais. Finalmente, na Seção 5, são apresentadas as conclusões e as direções para trabalhos futuros.

II. Detecção Automática de Plágio

A. Terminologia

O problema de detecção automática de plágio pode ser definido como a tarefa de encontrar, dado um documento suspeito d_{plg}, o subconjunto D_{src} de documentos que foram usados como fontes, assim como identificar as passagens destes documentos que não foram proferidas pelo referido autor. Quando apoiado por buscas em uma coleção externa D de documentos, este processo passa a ser denominado *detecção extrínseca de plágio*. Detectores nestas condições normalmente implementam duas etapas básicas: recuperação de fontes e identificação de passagens.

B. Recuperação de Fontes

A etapa de recuperação de fontes consiste em obter o subconjunto $D_{src} \subseteq D$ de documentos fontes de d_{plg} ao mesmo tempo em que deseja-se minimizar os custos desta recuperação. Em um caso típico de plágio, no qual trechos são extraídos de documentos da Web, recuperar precisamente cada documento fonte $d_{src} \in D_{src}$ sem trazer junto quantidades significativas de documentos irrelevantes é uma tarefa difícil. Em cenários como este, a etapa de recuperação

de fontes acaba funcionando como uma espécie de pré-processamento a fim reduzir os custos de uma posterior inspeção detalhada entre d_{plg} e cada documento $d \in D$. Nestas situações, buscas preliminares em D são realizadas com trechos extraídos de d_{plg} e os resultados são usados na construção de um subconjunto intermediário $D_{cand} \subseteq D$ de documentos com maior probabilidade de serem os fontes de d_{plg}. Fica claro que, para grandes coleções, deseja-se que $|D|_{cand} \ll |D|$. Por esta razão, decisões acerca de quais trechos analisar (segmentação), quais termos dos trechos submeter à busca (extração de termos-chave), como combiná-los para formular consultas (formulação de consultas) e quais resultados conisderar (filtragem dos resultados), passam a ter um papel fundamental na construção de D_{cand} e, em última instância, de D_{src}. A Figura 1 ilustra, de maneira genérica, as fases do processo de detecção extrínseca de plágio. A seguir, são discutidos em maiores detalhes algumas questões associadas às fases de obtenção de D_{cand}.

1) Segmentação: A fase de segmentação divide o documento suspeito em segmentos (sobrepostos ou não) para serem processados individualmente. Dessa forma, espera-se dar igual atenção às partes do documento e evitar que os métodos empregados nas fases seguintes tornem-se suceptíveis a alguma especificidade.

2) Extração de Termos-Chave: A fase de extração de termos-chave seleciona termos ou passagens de cada segmento a serem submetido à busca. São empregados métodos de seleção de termos que maximizem as chances de recuperar somente documentos pertinentes e que, ao mesmo tempo, limitam a quantidade de consultas realizadas. Talvez este seja o passo mais importante e decisões tomadas aqui afetam diretamente o desempenho geral do sistema.

3) Formulação de Consultas: A fase de formulação de consultas usa os termos selecionados na fase anterior para formular consultas. Para isso, são considerados aspectos como o tamanho, a quantidade e a composição dos termos, além das limitações impostas pelo mecanismo de busca. Dentre elas, estão a disponibilidade de operadores lógicos, a quantidade máxima de termos por consulta e o suporte ao casamento exato de padrões.

4) Controle de Busca: A fase de controle de busca escalona a submissão das consultas de forma que uma consulta seja enviada somente após o processamento dos resultados da consulta anterior. Dessa maneira, espera-se obter informações relevantes que justifiquem ajustes dinâmicos, tais como o descarte de consultas, a formulação de novas consultas ou a reformulação das já existentes.

5) Filtragem dos Resultados: A fase de filtragem dos resultados desconsidera documentos retornados pelo mecanismo de busca que, por alguma razão, não são relevantes o suficiente para serem inspecionados detalhadamente em busca de passagens plagiadas. Os documentos remanescentes formam o conjunto D_{cand} de documentos. Dessa forma, espera-se reduzir ao máximo o tamanho de D_{cand} e diminuir o custo computacional da etapa seguinte.

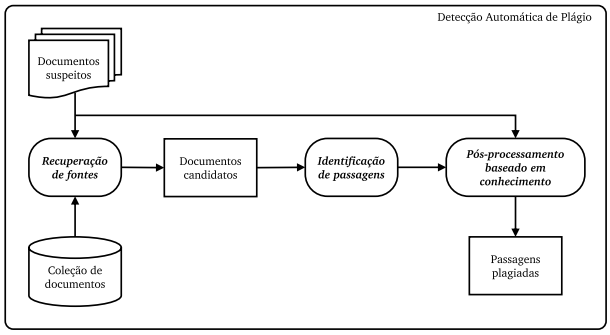

Figura 1. Processo genérico para detecção de plágio

C. Identificação de Passagens

A etapa de identificação de passagens consiste em encontrar, para cada $d_{cand} \in D_{cand}$, os trechos reutilizados na construção de d_{plg}. Para isso, é feita uma inspeção detalhada em d_{cand} e d_{plg} em busca de passagens copiadas ou modificadas que possuem pouca ou nenhuma similaridade com o texto original. Documentos cujas passagens foram identificadas como plágio formam o conjunto D_{src} e, juntamente com as anotações de início e término dos trechos reusados, são encaminhados para avaliação por um especialista para pós-processamento baseado em conhecimento.

III. Metodologia Experimental

A. Arcabouço PAN

Apesar de muitos estudos terem sido realizados na área, sempre houve uma enorme carência de ferramentas apropriadas para apoiar a tarefa de detecção de plágio. Como consequência, muitos trabalhos nunca puderam ser reproduzidos ou comparados entre si [5]. Diante disso, pesquisadores decidiram criar o *International Workshop on Plagiarism Analysis (PAN)* — competição anual para trabalhos nas áreas de detecção de plágio, identificação de autoria e perfilação de autores — e, juntamente, disponibilizar um arcabouço ferramental para padronizar diversos aspectos que divergiam em pesquisas realizadas até então.

Este arcabouço é formado por uma coleção de documentos indexados por um mecanismo de busca, um conjuntos de documentos plagiados escritos manualmente e métricas de avaliação clássicas da área de Recuperação de Informação. Seus componentes estão disponibilizados gratuitamente no sítio dos organizadores para que qualquer pessoa interessada no assunto realize experimentos sob as mesma condições e possa comparar seus resultados com os dos demais pesquisadores.

O componente de dados deste arcabouço está a cargo da ClueWeb09, uma coleção de documento criada pelo Language Technologies Institute da Carnegie Mellon University para auxiliar nas pesquisas de recuperação de informação [4]. Sua escolha não foi ao acaso; a ClueWeb09 é considerada uma aproximação real da Web pública e grande o suficiente para tornar inviável a comparação exaustiva entre

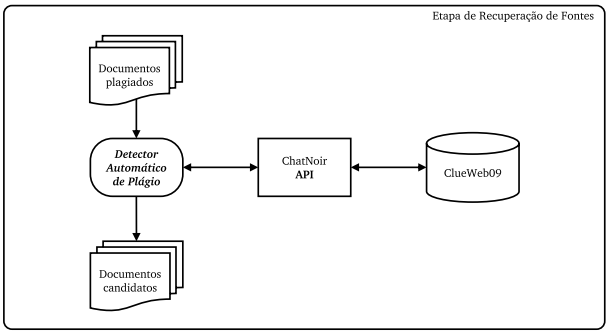

Figura 2. Etapa de recuperação de fontes simulada pelo arcabouço PAN.

pares de documentos, tornando imprescindível a etapa de recuperação de fontes. De fato, ela é composta por mais de 1 bilhão de páginas em 10 idiomas, sendo mais da metade em inglês, coletadas da Web entre janeiro e fevereiro de 2009, totalizando cerca de 25 terabytes de dados [6].

Esta coleção está indexada pelo ChatNoir [6], um mecanismo de busca nos moldes de soluções comerciais e que foi contruído especificamente para este fim. Ele adota o modelo de recuperação BM25F [7] e conta com ordenação por PageRank [8] e SpamRank [8].

Por fim, os documentos plagiados foram cuidadosamente elaborados por 28 profissionais, fluentes em inglês, contratados através da ferramenta de *crowdsourcing* oDesk. A eles coube a tarefa de escrever 300 documentos de aproximadamente 5000 palavras e com até 30 passagens plagiadas sobre tópicos selecionados da coleção TREC WebTrack. Esta atividade foi apoiada por um editor de texto capaz de manter registros sobre os documentos fontes usados, páginas consultadas, *links* visitados, buscas realizadas, além de um histórico de edição.

O arcabouço PAN oferece aos seus usuários meios para avaliar os sistemas de detecção de plágio em apenas uma ou em ambas as etapas. Neste trabalho, optou-se em avaliar apenas a etapa de recuperação de fontes. A Figura 2 apresenta uma visão geral da etapa de recuperação de fontes simulada pelo arcabouço PAN, ilustrando as interações realizadas entre seus componentes e os sistemas de detecção de plágio que a utilizam.

Para avaliar o desempenho da etapa de recuperação de fontes, foi utilizado um conjunto D_{plg} com 40 documentos plagiados, escritos em HTML, e um conjunto D_{src} de arquivos JSON associados contendo endereços para os documentos fontes.

B. Etapa de Recuperação de Fontes

1) Pré-processamento: Na fase de pré-processamento, os documentos foram divididos em sentenças e os termos das sentenças foram identificados, sendo que todos os termos gráficos (pontuações, símbolos de moeda, datas, etc.), com exceção de substantivos, verbos e adjetivos, foram desconsiderados. Para isso, foi usada a ferramenta Stanford Natural Language Processing (NLP), em Java, desenvolvida pela Stanford University.

A remoção de termos ocorreu de acordo com a classificação dada pelo *Part-Of-Speech Tagger,* incluído na ferramenta Stanford NLP, que as classificou segundo o *Penn Treebank's Tagset* [9]. Se alguma palavra não pertencia às classes gramaticais de interesse ou se sua forma canônica não estava contida em um dicionário da língua inglesa, ela era sumariamente removida. O dicionário, obtido junto ao projeto SCOWL [10], continha cerca de 160 mil termos e é o mesmo usado no verificador ortográfico Aspell [11].

2) Segmentação: Na fase de segmentação, os documentos foram divididos em segmentos com 10, 20, 30, 40 ou 50 sentenças.

3) Extração de Termos-Chave: Na fase de extração de termos-chave, foram usados um dentre 6 métodos de seleção de termos. Exceto o primeiro (método sequencial), todos os outros estimaram, para cada segmento, a importância dos termos e selecionaram os mais importantes para que servissem de âncoras na formulação das consultas. A seguir, são descritos cada um dos métodos.

a) Sequencial: O método sequencial seleciona os termos subsequentes a partir do início de cada segmento. Este é o método mais simples e o desempenho das estratégias que o utilizam foi usado com referência para o desempenho das demais.

b) Term Frequency—Inverse Segment Frequency (TF-ISF): O método TF-ISF é uma variação do conhecido *Term Frequency—Inverse Document Frequency* (TF-IDF) e estima a importância de um termo w em um segmento s em um documento d pela frequência $f_{w,s}$ de w em s e pela quantidade de segmentos onde w ocorre. Para um termo w esta importância é dada por:

$$tf\text{-}isf(w) = f_{w,s} \times \log\left(\frac{N}{n_d}\right)$$

onde N é o total de segmentos de d e n_d é o total de segmentos de d onde w ocorre pelo menos uma vez. Este método tem a propriedade de dar maior importância aos termos frequentemente usados em um segmento e que, ao mesmo tempo, possuem baixa ocorrência nos demais [12].

c) Hyperlink-Induced Topic Search (HITS): O método HITS baseia-se no conceito de autoridades e centros, similiar ao *PageRank* [13], e foi elaborado originalmente para estimar a importância de páginas Web [14]. Na adaptação deste método para a seleção de termos, cada documento d_{plg} é transformado em um grafo direcionado $G(V, E)$, onde os termos representam vértices e a precedência entre eles, arestas. Se um termo qualquer a precede imediatamente outro termo b em alguma sentença $s \in d_{plg}$, então uma aresta direcionada $E(v_a, v_b)$ do vértice v_a para o vértice v_b é adicionada em G.

O algoritmo inicia-se atribuindo 1 aos valores de autoridade *(auth)* e centro *(hub)* de todos os vértices e, a cada iteração, atualiza-os pela soma de *auth* e *hub* dos vértices adjacentes. Mais formalmente, para um vértice v_i em uma

iteração qualquer, os valores de *auth* e *hub* são atualizados por:

$$\left.\begin{aligned} auth(v_i) &= \sum_{j=1}^{n} hub(v_j) \\ hub(v_i) &= \sum_{j=1}^{n} auth(v_j) \end{aligned}\right\} \Leftrightarrow \exists E(v_j, v_i) \in G$$

O algoritmo é interrompido após a execução de um número máximo de iterações. A importância dos termos é dada pela média dos valores de *auth* e *hub* dos respectivos vértices, ressaltando termos com a maior diversidade de co-ocorrência. Neste trabalho, o número máximo de iterações foi de 5, obtido empiricamente.

d) Chisquare (χ^2): O método χ^2 usa informações estatísticas para estimar a importância de um termo w em um documento d. A partir do conjunto G dos termos mais frequentes de d, o método calcula a importância de w em termos de sua frequência de co-ocorrência com cada termo $g \in G$ por:

$$\chi^2(w) = \sum_{g \in G} \frac{\left(freq(w,g) - n_w p_g\right)^2}{n_w p_g}$$

onde $freq(w,g)$ é a frequência de co-ocorrência entre w e g; n_w é o número total de termos nas sentenças onde w ocorre; e p_g é o número total de termos nas sentenças onde g aparece dividido pelo número total de termos em d. Diz-se que dois termos co-ocorrem se ambos aparecem em uma mesma sentença.

O método *chisquare* tem a vantagem apresentar o desempenho similar ao do TF-IDF clássico sem a necessidade de uma base de dados externa [15]. Neste trabalho, o conjunto G foi formado por 30% dos termos mais frequentes de cada documento.

e) Termos Menos Frequentes: Este método seleciona os termos menos frequentes do documento e usa-os como âncoras em cada segmento. Dessa forma, espera-se obter palavras que fogem ao vocabulário típico do autor, na esperança de que tais palavras representem, de fato, trechos extraídos de outras obras.

f) Termos Igualmente Espaçados: Este método seleciona termos equidistantes para servirem de âncoras. Para isso, dados q consultas a serem formuladas e um segmento com t termos, o algoritmo divide o segmento em q trechos de $n = \frac{t}{q}$ palavras. Para cada trecho i, $1 \leq i \leq q$, o termo intermediário t_i na posição $(i-1)n + \frac{n}{2}$ é adicionado ao conjunto de âncoras do segmento. Ao contrário dos métodos anteriores, não faz-se nenhuma suposição em torno dos termos e assume-se que os trechos plagiados podem estar igualmente distribuídos ao longo do texto com a mesma probabilidade [16].

4) Formulação de Consultas: Na fase de formulação de consultas, foram adicionadas quantidades iguais de termos à esquerda e à direita de cada âncora até atingir o total de 10 termos. Consultas com menos do que 5 termos foram descartadas.

5) Filtragem dos Resultados: Na fase de filtragem dos resultados, as consultas foram configuradas para retornar, no máximo, 10 documentos. Para cada consulta, os documentos retornados pelo mecanismo de busca foram re-ordenados segundo o peso a eles atribuídos por um dentre 6 métodos de ordenação. Além disso, foi definido um limiar de poda do resultado; os documentos posicionados abaixo da posição 1, 3, 5 ou 10 foram desconsiderados. A seguir, são descritos cada um dos métodos de ordenação de documentos.

a) Pagerank: O método *PageRank* estima a importância de um documento qualquer da Web através de um algoritmo recursivo que considera os valores de *PageRank* dos documentos que o referenciam. Para isso, é construída uma representação em grafos da rede com os documentos representando vértices e os *links* entre eles, arestas. Cada *link* é considerado um indicativo de relevância e documentos centrais possuem importâncias diferenciadas [13].

b) SpamRank: O método *SpamRank* determina a importância de um documento em termos da fração do documento que pode ser considerada *spam*. Documentos que utilizam mecanismos de autopromoção e promoção mútua para melhorar sua posição no resultado dos mecanismos de busca recebem os valores mais baixos. Os valores de *Spamrank* para as páginas da coleção ClueWeb09 foram calculados pela Waterloo University [17].

c) BM25F: O método BM25F [7] é uma modificação do método BM25 que atribui pesos distintos à importância dos termos da consulta que ocorrem em diferentes partes do documento, tais como títulos e âncoras [8]. Os métodos da família BM25 calculam a importância de um documento em relação a uma consulta como a soma da importância dos termos da consulta no documento; esta importância é calculada por métodos do tipo TF-IDF, dentre os quais os métodos da a família BM25 são considerados o estado da arte [18].

d) Proximity: O método *Proximity* calcula a importância de um documento em relação a uma consulta pela distância entre termos consecutivos da consulta no documento. Esta distância é dada pelo número de termos que os separam. Este método exige, ainda, que a ordem dos termos na consulta seja a mesma no documento [8].

Se não for especificado nenhum método de re-ordenação, a ordem padrão dos resultados é mantida, com a importância do documento d em relação a uma consulta q sendo dada por:

$$\begin{aligned} weight_{d,q} &= \big(bm25f_{d,q} + 0.1 \times proximity_{d,q} + \\ &\quad pagerank_d\big) \times spamrank_d \end{aligned}$$

onde $bm25f_{d,q}$ denota a importância de d em relação a q calculada pelo método de filtragem de documentos BM25F, $proximity_{d,q}$ a importância de d em relação a q pelo método *Proximity*, $pagerank_d$ a importância de d pelo método *PageRank* e $spamrank_d$ a importância de d pelo método *SpamRank*.

Tabela I
PERCENTUAIS DE SUBMISSÃO DOS DOCUMENTOS SUSPEITOS E RESPECTIVAS COMBINAÇÕES ENTRE QUANTIDADE DE CONSULTAS E TAMANHO DE SEGMENTOS.

Percentual de Submissão					
10%		30%		50%	
Quantidade de Consultas	Tamanho do Segmento*	Quantidade de Consultas	Tamanho do Segmento*	Quantidade de Consultas	Tamanho do Segmento*
1	10	3	10	5	10
2	20	6	20	10	20
3	30	9	30	15	30
4	40	12	40	20	40
5	50	15	50	25	50

* Em sentenças

Avaliamos o desempenho de todas as combinações entre os 6 métodos de seleção de termos, 5 métodos de filtragem de resultados e 4 limiares de poda dos resultados mencionados acima, bem como 3 percentuais de submissão do conteúdo dos documentos plagiados e 5 combinações entre quantidades de consultas e tamanhos de segmentos, mostrados na I. Ao todo, foram $6 \times 5 \times 4 \times 3 \times 5 = 1800$ configurações distintas de experimentos.

C. Métricas de Avaliação

Para avaliar o desempenho da etapa de recuperação de fontes, foram usadas as métricas precisão, revocação e F1. Na recuperação de informação, a métrica precisão é definida como o número de documentos relevantes recuperados por um mecanismo de busca dividido pelo número total de documentos recuperados; a métrica revocação é definida como o número de documentos relevantes recuperados dividido pelo número total de documentos relevantes; e a métrica F1 é definida como a média harmônica dessas duas métricas. No contexto de recuperação de fontes em sistemas de detecção de plágio, documentos relevantes são aqueles usados como fontes de plágio [19].

Mais especificamente, dado um documento d_{plg} com passagens plagiadas de um conjunto D_{src} de documentos fontes e um conjunto D_{ret} de documentos recuperados pelo mecanismo de busca, as métricas precisão, revocação e F1 são dadas por:

$$\begin{aligned} \textit{precisão} &= \frac{|D_{ret} \cap D_{src}|}{|D_{ret}|} \\ \textit{revocação} &= \frac{|D_{ret} \cap D_{src}|}{|D_{src}|} \\ F1 &= 2 \times \frac{\textit{precisão} \times \textit{revocação}}{\textit{precisão} + \textit{revocação}} \end{aligned}$$

IV. RESULTADOS EXPERIMENTAIS E DISCUSSÃO

A Figura II mostra os melhores resultados obtidos em termos de F1. O desempenho geral de cada configuração foi avaliado em termos da métrica F1 por equilibrar precisão e revocação. A última coluna da Figura II representa o desempenho em relação à estratégia que utiliza o método sequencial, usado como referência para as demais estratégias. Nesta coluna, células sombreadas com cores mais escuras apresentam ganhos mais expressivos.

Os resultados experimentais mostram que as estratégias que usam o método que seleciona os termos menos frequentes como âncoras para formulação de consultas apresentaram desempenho superior às demais. Isto revela que palavras incomuns são boas fontes de evidência de vocabulários reusados de outras fontes e que autores de plágio não se esforçam o suficiente para alterá-las. Esta idéia ganha força quando observa-se que, dentre estas estratégias, aquelas que usam o método de ordenação pelo valor de *Proximity* apresentaram melhor desempenho. De fato, a forma de plágio mais facilmente detectável é aquela que preserva a ordem original das palavras ou cujos autores não empenharam-se o suficiente para disfarçá-lo, realizando cópias quase literais.

Nota-se também, que focar-se apenas no documento do topo da ordenação é a melhor maneira de equilibrar precisão e revocação. Isto indica que trechos plagiados normalmente possuem apenas uma fonte e que autores de plágio, na busca por fontes, atêm-se apenas aos primeiros resultados exibidos pelo mecanismo de busca.

Finalmente, observa-se que o melhor desempenho foi obtido ao submeter-se 10% do conteúdo dos documentos plagiados. No entanto, a estratégia que obteve o segundo melhor desempenho submeteu 30% do conteúdo. Esta última estratégia destaca-se por alcançar valores de F1 semelhantes à primeira, mas com níveis de revocação mais elevados, próximos daqueles obtidos por estratégias que submetem quantidades maiores de informação.

V. CONCLUSÕES E TRABALHOS FUTUROS

Neste trabalho, foram investigados o desempenho de várias estratégias para recuperação de fontes em sistemas de detecção de plágio usando-se o arcabouço PAN e três métricas de RI: precisão, revocação e F1. Os resultados experimentais mostraram que a estratégia que combina o método de seleção de termos menos frequentes, o método de filtragem dos resultados *Proximity* e o limiar de poda igual a 1, superou o desempenho das demais estratégias analisadas. Os resultados mostraram também que, apesar da estratégia que submete 10% do conteúdo dos documentos plagiados ter alcançado o melhor desempenho, aquelas que submetem 30% do conteúdo, nas mesmas condições, atingem níveis de revocação mais elevados a pequenos custos de precisão, pondo-se como uma alternativa interessante.

Uma direção para trabalho futuro seria considerar quantidades maiores de resultados para cada consulta, ao invés de focar-se apenas nos 10 primeiros, o que poderá revelar outros aspectos do comportamento dos autores de plágio. Outra vertente para pesquisas futuras seria usar mecanismos de tradução de documentos que estejam em outros idiomas, os quais poderão permitir a detecção de plágio de tradução. Por fim, outra sugestão para investigações futuras seria usar

Tabela II
MELHORES RESULTADOS EM TERMOS DA MÉTRICA F1.

Resultados									
PS	**MST**	**LP**	**QC**	**TS**	**MFR**	**P**	**R**	**F1**	**%**
10%	Sequencial	1	1	10	SpamRank	0,0535	0,1337	0,0726	0,00
		3	1	10	Padrão	0,0327	0,2154	0,0554	0,00
		5	1	10	Padrão	0,0247	0,2459	0,0441	0,00
		10	1	10	Padrão	0,0158	0,2837	0,0295	0,00
	TFISF	1	3	30	Padrão	0,0574	0,1526	0,0788	8,54
		3	3	30	Proximity	0,0330	0,2522	0,0568	2,53
		5	3	30	Proximity	0,0248	0,2948	0,0449	1,81
		10	3	30	Padrão	0,0159	0,3284	0,0299	1,36
	HITS	1	5	50	PageRank	0,0495	0,1296	0,0686	-5,51
		3	5	50	Proximity	0,0302	0,2127	0,0516	-6,86
		5	5	50	Proximity	0,0223	0,2530	0,0403	-8,62
		10	5	50	Proximity	0,0135	0,2788	0,0254	-13,90
	Chisquare	1	2	20	Proximity	0,0635	0,1532	0,0867	19,42
		3	2	20	Proximity	0,0361	0,2551	0,0618	11,55
		5	2	20	Proximity	0,0256	0,2866	0,0462	4,76
		10	2	20	Proximity	0,0161	0,3250	0,0303	2,71
	Termos Menos Frequentes	1	2	20	Proximity	0,0800	0,1936	0,1091	50,28
		3	3	30	Padrão	0,0428	0,2845	0,0721	30,14
		5	2	20	Proximity	0,0313	0,3219	0,0559	26,76
		10	2	20	Proximity	0,0201	0,3562	0,0375	27,12
	Termos Igualmente Espaçados	1	4	40	Padrão	0,0541	0,1437	0,0754	3,86
		3	1	10	SpamRank	0,0314	0,2237	0,0536	-3,25
		5	1	10	SpamRank	0,0251	0,2670	0,0451	2,27
		10	1	10	SpamRank	0,0157	0,3016	0,0295	0,00
30%	Sequencial	1	9	30	Proximity	0,0352	0,2529	0,0605	0,00
		3	9	30	Proximity	0,0190	0,3635	0,0358	0,00
		5	9	30	Proximity	0,0134	0,3913	0,0257	0,00
		10	9	30	Proximity	0,0079	0,4013	0,0155	0,00
	TFISF	1	9	30	Proximity	0,0473	0,3418	0,0815	34,71
		3	9	30	Proximity	0,0241	0,4923	0,0455	27,09
		5	9	30	Proximity	0,0165	0,5280	0,0319	24,12
		10	9	30	Proximity	0,0097	0,5581	0,0189	21,94
	HITS	1	6	20	Proximity	0,0443	0,3177	0,0763	26,12
		3	6	20	Proximity	0,0230	0,4681	0,0435	21,51
		5	6	20	Proximity	0,0159	0,5206	0,0307	19,46
		10	6	20	Proximity	0,0093	0,5519	0,0183	-51,33
	Chisquare	1	3	10	Padrão	0,0367	0,2693	0,0633	4,63
		3	6	20	Proximity	0,0199	0,3991	0,0376	5,03
		5	12	40	Proximity	0,0141	0,4572	0,0272	5,84
		10	12	40	Proximity	0,0084	0,4949	0,0165	6,45
	Termos Menos Frequentes	1	15	50	Proximity	0,0561	0,3815	0,0961	58,84
		3	15	50	Proximity	0,0269	0,5002	0,0507	41,62
		5	15	50	Proximity	0,0186	0,5411	0,0358	39,30
		10	15	50	Proximity	0,0114	0,5762	0,0222	43,23
	Termos Igualmente Espaçados	1	15	50	Proximity	0,0334	0,2601	0,0580	-4,13
		3	15	50	Proximity	0,0167	0,3643	0,0317	-11,45
		5	15	50	Proximity	0,0118	0,4012	0,0228	-11,28
		10	15	50	Proximity	0,0070	0,4213	0,0138	-10,97
50%	Sequencial	1	15	30	Proximity	0,0302	0,3524	0,0549	0,00
		3	15	30	Proximity	0,0153	0,4734	0,0295	0,00
		5	15	30	Proximity	0,0103	0,4983	0,0201	0,00
		10	15	30	Proximity	0,0061	0,5114	0,0120	0,00
	TFISF	1	20	40	Proximity	0,0321	0,3717	0,0583	6,19
		3	20	40	Proximity	0,0152	0,4676	0,0293	-0,68
		5	20	40	Proximity	0,0104	0,5163	0,0204	1,49
		10	40	40	Proximity	0,0060	0,5351	0,0119	-0,83
	HITS	1	25	50	Proximity	0,0347	0,3968	0,0631	14,94
		3	25	50	Proximity	0,0163	0,5242	0,0315	6,78
		5	25	50	Proximity	0,0110	0,5537	0,0215	6,97
		10	25	50	Proximity	0,0063	0,5755	0,0125	4,17
	Chisquare	1	20	40	Proximity	0,0302	0,3395	0,0548	-0,18
		3	25	50	Padrão	0,0170	0,4603	0,0303	2,71
		5	25	50	Padrão	0,0115	0,5068	0,0214	6,47
		10	25	50	Padrão	0,0066	0,5357	0,0127	5,83
	Termos Menos Frequentes	1	20	40	Proximity	0,0484	0,3401	0,0781	42,26
		3	20	40	Proximity	0,0224	0,4424	0,0411	39,32
		5	25	30	Proximity	0,0162	0,4469	0,0294	46,27
		10	25	30	Proximity	0,0097	0,4737	0,0184	53,33
	Termos Igualmente Espaçados	1	5	10	Proximity	0,0273	0,3359	0,0498	-9,29
		3	25	30	Proximity	0,0139	0,4663	0,0268	-9,15
		5	25	30	Proximity	0,0095	0,5088	0,0185	-7,96
		10	25	30	Proximity	0,0055	0,5264	0,0108	-10,00

PS: Percentual de Submissão
MST: Método de Seleção de Termos
LP: Limiar de Poda
QC: Quantidade de Consultas
TS: Tamanho do Segmento (em sentenças)
MFR: Método de Filtragem dos Resultados
P: Precisão
R: Revocação

ferramentas de análise de vocabulário, as quais poderão permitir o levantamento do perfil dos autores e, consequentemente, a detecção de variações linguísticas significativas que sugiram indícios de plágio

REFERÊNCIAS

[1] S. Brin, J. Davis, and H. Garcia-Molina, "Copy detection mechanisms for digital documents," in *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '95, C. M. J. and S. D. A., Eds. New York, NY, USA: ACM, 1995, pp. 398–409.

[2] A. Z. Broder, N. Eiron, M. Fontoura, M. Herscovici, R. Lempel, J. McPherson, R. Qi, and E. Shekita, "Indexing shared content in information retrieval systems," in *Advances in Database Technology - EDBT 2006, 10th International Conference on Extending Database Technology*, Munich, Germany, March 2006, pp. 313–330.

[3] D. Fetterly, M. Manasse, and M. Najork, "On the evolution of clusters of near-duplicate web pages," in *Proceedings of the First Conference on Latin American Web Congress*, ser. LA-WEB '03. Washington, DC, USA: IEEE Computer Society, 2003, pp. 37–.

[4] M. Potthast, T. Gollub, M. Hagen, J. Kiesel, M. Michel, A. Oberländer, M. Tippmann, A. Barrón-Cedeño, P. Gupta, P. Rosso, and B. Stein, "Overview of the 4th international competition on plagiarism detection," in *CLEF 2012 Evaluation Labs and Workshop, Online Working Notes, Rome, Italy, September 17-20, 2012*, ser. CEUR Workshop Proceedings, P. Forner, J. Karlgren, and C. Womser-Hacker, Eds., vol. 1178. CEUR-WS.org, 2012.

[5] M. Potthast, B. Stein, A. Barrón-Cedeño, , and P. Rosso, "An evaluation framework for plagiarism detection," in *23rd International Conference on Computational Linguistics (COLING '10)*, I. C.-R. Huang and D. Jurafsky, Eds. Stroudsburg, PA, USA: Association for Computational Linguistics, August 2010, pp. 997–1005.

[6] M. Potthast, M. Hagen, B. S. J. Graßegger, M. Michel, M. Tippmann, and C. Welsch, "Chatnoir: A search engine for the clueweb09 corpus," in *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '12. New York, NY, USA: ACM, 2012, pp. 1004–1004.

[7] S. Robertson, H. Zaragoza, and M. Taylor, "Simple bm25 extension to multiple weighted fields," in *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management*, ser. CIKM '04. New York, NY, USA: ACM, 2004, pp. 42–49.

[8] J. Graßegger, M. Michel, M. Tippmann, M. Hagen, M. Potthast, and B. Stein, "The chatnoir ranking," Aug. 2012.

[9] M. P. Marcus, M. A. Marcinkiewicz, and B. Santorini, "Building a large annotated corpus of english: The penn treebank," *Comput. Linguist.*, vol. 19, no. 2, pp. 313–330, Jun. 1993.

[10] "Spell checker oriented word lists," http://wordlist.aspell.net/, acessado em 12-04-2015.

[11] "Aspell," http://aspell.net/, acessado em 16-04-2015.

[12] J. L. Neto, A. D. Santos, C. Kaestner, and A. Freitas, "Generating text summaries through the relative importance of topics," in *Advances in Artificial Intelligence*, ser. Lecture Notes in Computer Science, M. Monard and J. Sichman, Eds. Springer Berlin Heidelberg, 2000, vol. 1952, pp. 300–309.

[13] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web," 1998.

[14] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *ACM*, vol. 46, no. 5, pp. 604–632, Sep. 1999.

[15] Y. Matsuo and M. Ishizuka, "Keyword extraction from a single document using word co-occurrence statistical information," *International Journal on Artificial Intelligence Tools*, vol. 13, 2004.

[16] A. Pereira and N. Ziviani, "Retrieving similar documents from the web," *Journal of Web Engineering*, vol. 2, no. 4, pp. 247–261, Oct. 2003.

[17] G. Cormack, M. Smucker, and C. Clarke, "Efficient and effective spam filtering and re-ranking for large web datasets," *Information Retrieval*, vol. 14, no. 5, pp. 441–465, 2011.

[18] H. Zaragoza, N. Craswell, M. Taylor, S. Saria, and S. Robertson, "Microsoft cambridge at trec-13: Web and hard tracks," in *In Proceedings Of TREC 2004*, 2004.

[19] F. Sebastiani, "Machine learning in automated text categorization," *ACM Comput. Surv.*, vol. 34, no. 1, pp. 1–47, Mar. 2002.