UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

GLAUBER RODRIGUES DA SILVA

VersionsRank: escores de reputação de páginas Web baseados na detecção de versões

Dissertação apresentada como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação

Prof. Dra. Renata Galante
Orientadora

Porto Alegre, novembro de 2009.

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Silva, Glauber Rodrigues da

VersionsRank: escores de reputação de páginas *Web* baseados na detecção de versões / Glauber Rodrigues da Silva – Porto Alegre: Programa de Pós-Graduação em Computação, 2009.

46 f.:il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação. Porto Alegre, BR – RS, 2009. Orientadora: Renata Galante;

1.*Ranking*. 2.Detecção de Versões 3.*PageRank*. I. Galante, Renata. II. VersionsRank: escores de reputação de páginas *Web* baseados na detecção de versões.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Reitor: Prof. Carlos Alexandre Netto
Vice-Reitor: Prof. Rui Vicente Oppermann
Pró-Reitor de Pós-Graduação: Prof. Aldo Bolten Lucion
Diretor do Instituto de Informática: Prof. Flávio Rech Wagner
Coordenador do PPGC: Prof. Álvaro Freitas Moreira
Bibliotecária-Chefe do Instituto de Informática: Beatriz Regina Bastos Haro

AGRADECIMENTOS

Agradeço primeiramente aos meus pais Evaldo Soares da Silva e Maria de Lourdes Velho da Silva pelo apoio incondicional em toda a minha formação.

A minha esposa Elisângela Pinheiro Reginato pela compreensão durante todos os finais de semana dedicados a esse trabalho.

A minha orientadora Renata Galante por todos os ensinamentos e por confiar na minha capacidade de desenvolver esse trabalho. Agradeço também ao professor Marcos André Gonçalves pela valiosa colaboração na elaboração desse trabalho.

Aos colegas e professores do Instituto de Informática e a Universidade Federal do Rio Grande do Sul, o meu muito obrigado pela oportunidade de convívio e aprendizagem proporcionados.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS

HTML	HyperText Markup Language
URL	Uniform Resource Locator
P@10	Precision at 10
MAP	Mean Average Precision
MRR	Mean Reciprocal Rank
MD5	Message-Digest algorithm 5
SHA-1	Secure Hash Algorithm 1
VSR	Version Sum Rank
VAR	Version Average Rank
PR	Page Rank
VR	Version Rank
VPR	Version Page Rank

LISTA DE FIGURAS

LISTA DE TABELAS

RESUMO

Os motores de busca utilizam o *WebGraph* formado pelas páginas e seus *links* para atribuir reputação às páginas *Web*. Essa reputação é utilizada para montar o *ranking* de resultados retornados ao usuário. No entanto, novas versões de páginas com uma boa reputação acabam por distribuir os votos de reputação entre todas as versões, trazendo prejuízo à página original e também as suas versões. O objetivo deste trabalho é especificar novos escores que considerem todas as versões de uma página *Web* para atribuir reputação para as mesmas. Para atingir esse objetivo, foram propostos quatro escores que utilizam a detecção de versões para atribuir uma reputação mais homogênea às páginas que são versões de um mesmo documento. Os quatro escores propostos podem ser classificados em duas categorias: os que realizam mudanças estruturais no *WebGraph* (*VersionRank e VersionPageRank*) e os que realizam operações aritméticas sobre os escores obtidos pelo algoritmo de *PageRank* (*VersionSumRank e VersionAverageRank*).

Os experimentos demonstram que o *VersionRank* tem desempenho 26,55% superior ao *PageRank* para consultas navegacionais sobre a WBR03 em termos de MRR, e em termos de P@10, o *VersionRank* tem um ganho de 9,84% para consultas informacionais da WBR99. Já o escore *VersionAverageRank*, apresentou melhores resultados na métrica P@10 para consultas informacionais na WBR99 e WBR03. Na WBR99, os ganhos foram de 6,74% sobre o *PageRank*. Na WBR03, para consultas informacionais aleatórias o escore *VersionAverageRank* obteve um ganho de 35,29% em relação ao *PageRank*.

Palavras-Chave: *ranking*, detecção de versões, *PageRank.*

ABSTRACT

Search engines use *WebGraph* formed by the pages and their links to assign reputation to *Web* pages. This reputation is used for ranking show for the user. However, new versions of pages with a good reputation distribute your votes of reputation among all versions, damaging the reputation of original page and also their versions. The objective of this work is to specify the new scores to consider all versions of a *Web* page to assign reputation to them. To achieve this goal, four scores were proposed using the version detection to assign a more homogeneous reputation to the pages that are versions of the same document. The four scores proposed can be classified into two categories: those who perform structural changes in *WebGraph* (*VersionRank* and *VersionPageRank*) and those who performs arithmetic operations on the scores obtained by the *PageRank* algorithm (*VersionSumRank* and *VersionAverageRank*).

The experiments show that the performance *VersionRank* is 26.55% higher than the *PageRank* for navigational queries on WBR03 in terms of MRR, and in terms of P@10, the *VersionRank* has a gain of 9.84% for the WBR99 informational queries. The score *VersionAverageRank* showed better results in the metric P@10 for WBR99 and WBR03 information queries. In WBR99, it had a gain of 6.74% compared to PageRank. In WBR03 for random informational queries, *VersionAverageRank* showed an increase of 35.29% compared to *PageRank.*

Keywords: *ranking, version detection, PageRank.*

1 INTRODUÇÃO

A natureza distribuída das informações disponíveis na Internet levou à busca de maneiras eficientes de executar consultas sobre uma grande coleção de documentos. Os motores de busca realizam essa tarefa. Entretanto, o conjunto de documentos relevantes para uma consulta pode facilmente ter milhões de itens. Nesse contexto, a montagem de um *ranking* de resultados eficaz passou a ser uma tarefa crucial. A estrutura de *links*, comum nos documentos HTML, permite estimar quais são as páginas mais "populares", partindo da premissa de que quanto maior o número de *links* que apontam para certa página (*links* esses vindos também de páginas populares), mais popular é essa página.

Os algoritmos que levam em conta essa estrutura de *links* na montagem do *ranking* dos resultados de uma pesquisa são chamados de algoritmos de análise de *links*. O algoritmo de *PageRank* (PAGE et al., 1998) é o que mais tem destaque nesta família de algoritmos. O termo *PageRank* é comumente atribuído a um número que mede a reputação de página *Web* na Internet, ou seja, quanto maior esse número, maior é a reputação da página.

A velocidade com que a estrutura da Internet se modifica acaba por trazer alguns desafios ao algoritmo de *PageRank*. Novas versões de uma página *Web* acabam por distribuir o escore de reputação da página para todas as suas versões. Por exemplo, considerando uma página *Web p*, um fator importante para constituir o *ranking* pelos motores de busca é a quantidade de outras páginas que apontam para *p*. Com o surgimento de inúmeras versões da página *p*, os votos de reputação para a mesma tendem a ficar espalhados entre as versões, causando um efeito indesejado de distribuição heterogênea do escore de reputação entre as versões da página. O ideal é que os votos de reputação sejam dados ao documento e não separadamente a cada uma de suas versões.

Além de melhorar o escore da página *p*, a detecção de versões ainda pode contribuir para melhorar o posicionamento no *ranking* de versões de *p*, baseado nos *links* que apontam para *p*. Por exemplo, considere a página *www.tempoglauber.com.br/glauber/Biografia/vida.htm* que por anos foi apontada por outras páginas como a página mais relevante sobre a biografia do cineasta Glauber Rocha. Suponha agora que o domínio *tempoglauber.com.br* seja adquirido por uma empresa que deseja colocar outro tipo de conteúdo no site e que a página sobre a biografia de Glauber Rocha tenha migrado para *www.glauberrocha.com.br/glauber-bio.html*, sofrendo pequenas alterações relativas ao conteúdo do site. As páginas que apontam para a antiga versão, além de terem *links* quebrados, não poderão atribuir reputação para a nova versão da página, visto que não sabem da existência da nova versão. Ao detectar que as duas páginas são, na verdade, versões de um mesmo documento lógico, é possível atribuir um escore de reputação único para o documento,

propiciando um melhor posicionamento no *ranking* da nova versão da página da biografia de Glauber Rocha.

Com a finalidade de classificar as consultas submetidas a um motor de busca, Broder (2002) propôs a definição de três classes de consultas: consultas navegacionais, consultas informacionais e consultas transacionais. Consultas navegacionais são consultas com fins navegacionais, ou seja, o usuário deseja ir a algum site e somente um resultado já satisfaz a consulta. Por exemplo, o usuário consulta "receita federal", na qual provavelmente o site da receita federal irá satisfazer o usuário. Já as consultas informacionais têm a intenção de obter alguma informação sobre um tópico específico, e normalmente mais de um resultado satisfazem a consulta. Por exemplo, a consulta "aprender idiomas", na qual vários resultados podem satisfazer o usuário. Por fim, consultas transacionais têm como objetivo realizar alguma ação, como realizar uma compra ou baixar algum arquivo. Por exemplo, a consulta "comprar DVD", na qual quaisquer páginas *Web* que ofereçam esse serviço serão relevantes para o usuário.

Alguns trabalhos tiveram sucesso ao adotar outra perspectiva para atribuir reputação às páginas *Web*. Existem trabalhos que conseguem melhores resultados com consultas navegacionais (BERLT et al., 2007), outros que conseguem melhores resultados em consultas informacionais (BAEZA-YATES et al., 2006) e outros ainda que conseguem melhores resultados em ambas (LIU et al., 2008; CARVALHO et al., 2006).

O trabalho de Liu et al. (2008) considera a navegação efetuada pelo usuário pelas páginas como indicativo de reputação das mesmas. O *ranking* obtido dessa navegação é denominado *BrowseRank*. O *BrowseRank* obteve resultados melhores que o *PageRank*, principalmente, nos casos em que o *PageRank* é afetado por *spams* – prejudicando consultas navegacionais e informacionais – o que não acontece com o *BrowseRank*. Já no trabalho de Berlt et al. (2007) foi proposto um modelo em que páginas de um mesmo *host* ou domínio são agrupadas para montar um *WebGraph*. Esse *WebGraph* agrupado foi denominado *hipergrafo*. A partir do *hipergrafo*, algoritmos de análise de *links*, como, por exemplo, o *PageRank*, são aplicados para obter novos escores. Os novos escores obtidos por Berlt et al. tiveram resultados melhores para consultas navegacionais quando comparados ao *PageRank*. O trabalho proposto nesta dissertação, por sua vez, utiliza a detecção de versões para atribuir às páginas *Web* escores homogêneos de reputação a todas as versões de um mesmo documento. Os novos escores acabam por beneficiar principalmente consultas informacionais, visto que se baseiam no conteúdo das páginas.

O objetivo deste trabalho é, portanto, especificar novos escores que considerem todas as versões de uma página *Web* para atribuir reputação para as mesmas. Para atingir esse objetivo, foram propostos quatro escores que utilizam a detecção de versões para atribuir uma reputação mais homogênea as páginas que são versões de um mesmo documento. Os quatro escores propostos podem ser classificados em duas categorias: os que realizam mudanças estruturais no *WebGraph* e os que realizam operações aritméticas sobre os escores obtidos pelo algoritmo de *PageRank*.

Os escores que realizam alterações na estrutura do *WebGraph* são o *VersionRank* e o *VersionPageRank*. O escore denominado *VersionRank* é baseado no algoritmo de *PageRank*, porém diferente do *PageRank*, que tem como modelo o *WebGraph*, o *VersionRank* tem o seu funcionamento baseado no *VersionGraph*. No *VersionGraph*, as versões de um mesmo documento *Web* são representadas como um único vértice no grafo direcionado formado pelas páginas *Web* e seus *links*. Já o escore

VersionPageRank utiliza os valores de reputação atribuídos pelo *VersionRank* e pelo *PageRank* para atribuir uma reputação baseada nos dois valores: quando a página é única na coleção é utilizado o escore *PageRank*, quando a página tem versões na coleção é utilizado o escore *VersionRank*. A principal contribuição desses dois escores é a utilização do modelo de *VersionGraph* para atribuir reputação às páginas *Web*. Através do modelo de *VersionGraph* é possível atribuir reputação aos documentos *Web* e não às versões dos mesmos – representadas pelas páginas *Web*.

Os escores que aplicam operações aritméticas sobre o *PageRank* são o *VersionAverageRank* e o *VersionSumRank*. O escore *VersionAverageRank* atribui a média dos *PageRank's* das versões de um documento a todas as suas versões. O escore *VersionSumRank* atribui a soma dos *PageRank's* das versões a todas as versões que compõem o documento. A principal contribuição desses dois escores é tornar homogênea a atribuição de reputação a todas as versões de uma página *Web* através de operações aritméticas sobre as versões da mesma.

Esta dissertação também apresenta um conjunto de experimentos que tem o objetivo de verificar a eficiência dos escores propostos em comparação com o principal escore de reputação da literatura, o *PageRank*, e os escores propostos por Berlt et al. Os experimentos foram realizados utilizando coleções extraídas de motores de busca reais: WBR99 e WBR03. A eficiência dos escores foi medida utilizando as métricas utilizadas pela área de recuperação de informação para a *Web*. Os experimentos demonstram que o *VersionRank* tem desempenho 26,55% superior ao *PageRank* para consultas navegacionais sobre a WBR03 em termos de MRR, e em termos de P@10, o *VersionRank* tem um ganho de 9,84% para consultas informacionais da WBR99. Já o escore *VersionAverageRank* apresentou melhores resultados na métrica P@10 para consultas informacionais na WBR99 e WBR03. Na WBR99, os ganhos foram de 6,74% sobre o *PageRank*. Na WBR03, para consultas informacionais aleatórias o escore *VersionAverageRank* obteve um ganho de 35,29% em relação ao *PageRank*. Já para consultas informacionais populares, o *VersionAverageRank* obteve um ganho de 14,79% na métrica P@10.

A principal contribuição deste trabalho é, portanto, a apresentação de novos escores de reputação que, ao realizar a detecção de versões de páginas *Web*, acabam por ser uma alternativa eficiente ao *PageRank* e aos escores propostos por Berlt et Al.

O restante do texto está organizado da seguinte forma. O capítulo 2 apresenta os principais trabalhos relacionados com esta dissertação. O capítulo 3 detalha a abordagem proposta para utilização da detecção de versões na atribuição de escores de reputação às páginas *Web*. Os experimentos realizados para medir o impacto no *ranking* apresentado pelos motores de busca, quando o mesmo é ordenado pelos escores propostos, são apresentados no capítulo 4. No capítulo 5 são apresentadas as conclusões do trabalho e também os possíveis trabalhos futuros.

2 TRABALHOS RELACIONADOS

Este capítulo apresenta os principais trabalhos relacionados ao tema da dissertação. Inicialmente, é apresentada a definição do termo *PageRank*, a definição do algoritmo de *PageRank* e algumas variações propostas para o algoritmo de *PageRank*. Em seguida, são apresentados os trabalhos diretamente relacionados ao tema da dissertação, ou seja, trabalhos que também propõem melhorias ao processo de *ranking*. Seguindo outra linha, são apresentadas as técnicas de detecção de quase duplicatas (*near-duplicate*). O tema detecção de duplicatas é importante no contexto desta dissertação porque será utilizado um algoritmo de cálculo de similaridade, já utilizado na detecção de quase duplicadas, adaptado nesse trabalho para a detecção de versões.

2.1 PageRank e suas variantes

Os primeiros algoritmos para ordenação dos resultados das consultas submetidas a um motor de busca eram dependentes da consulta e realizavam uma análise sobre o conteúdo das páginas para montar um *ranking*. Basicamente era verificado se as palavras-chave da busca estavam no título da página, ou estavam nos metadados de descrição da mesma e quantas ocorrências existiam das palavras-chave no conteúdo da página. De posse desses valores, era atribuído um escore a cada página que satisfazia os critérios da consulta e se montava a ordenação com base nos escores obtidos.

Em 1996 surgem os primeiros algoritmos que consideravam a "popularidade" das páginas na computação do escore. Esse escore era computado através de um modelo de análise de *links*. Esse modelo faz uma análise do grafo, cujos vértices são as páginas e os *links* são as arestas. Considerando que um *link* da página A para B é um "voto de popularidade" para a página B, quanto maior for o número de *links* (vindo de páginas também "populares") que apontam para certa página, maiores as chances de esta página ter um escore alto. Dentre esses algoritmos, o que mais teve destaque foi o *PageRank* (PAGE et al., 1998).

PageRank por definição é um número que mede a reputação de cada página *Web*. Isto significa que quanto maior o número, maior será a reputação da página. O algoritmo de PageRank é independente de consulta, dessa forma, é possível ordenar todas, ou quase todas, as páginas da Internet pelo seu grau de reputação, ou ainda, comparar um conjunto de páginas *Web*.

O algoritmo *PageRank* tem como principal objetivo simular o comportamento de um usuário ao navegar na Internet. Em outras palavras, o número do *PageRank* de uma página diz respeito a quão fácil/difícil seria encontrar esta página partindo de uma URL randômica e, seguindo os *links*, se tentasse chegar a página desejada. Quanto maior o

valor resultante de *PageRank*, mais fácil seria encontrar a página procurada (BRIN e PAGE, 1998).

Na atribuição da reputação, o algoritmo ainda reproduz a probabilidade de um navegador aleatório parar de seguir os *links* entre as páginas e introduzir uma nova URL no navegador através de uma variável denominada *damping factor*. O valor do escore de PageRank é dado pela fórmula a seguir:

$$PR(p_i) = \frac{1-d}{N} + d \sum_{p_j \in M(p_i)} \frac{PR(p_j)}{L(p_j)}$$

onde, *d* é o damping factor, *N* é o número total de páginas na coleção, *M(pi)* é o conjunto de todas as páginas que tem *links* para a página *pi*, *pj* é a página da iteração corrente. *L(pj)* representa quantos links de saída a página *pj* tem.

Outro algoritmo de análise de *links* bastante citado na literatura é o HITS, o qual não atribui um único escore às páginas e sim dois valores, um para *links* que saem da página e outro para *links* apontam para a página (*hub* e *authority*) e é processado em tempo de consulta e não em tempo de indexação (KLEINBERG, 1999).

Após a publicação do artigo que propôs o *PageRank* surgiram vários outros trabalhos propondo técnicas para melhorar a lógica e o desempenho do algoritmo de *PageRank*. Um exemplo foi a proposta de variante do algoritmo de *PageRank* que leva em conta os atributos das páginas *Web* (BAEZA-YATES e DAVIS, 2004). Em outro trabalho foi proposto um conjunto de algoritmos baseados em *links* (*link-based*), os quais propagam a importância das páginas através dos *links* (BAEZA-YATES et al., 2006). Também existe uma proposta de uma combinação dos algoritmos de *PageRank* e HITS para a criação de um *framework* unificado para análise de *links* (DING et al., 2002).

Nessa dissertação o escore *PageRank* será utilizado como base para comparativos visto que os escores propostos nesse trabalho utilizam, ou o próprio escore *PageRank*, ou a lógica do algoritmo de *PageRank*, na atribuição de reputação às páginas *Web*.

2.2 Trabalhos Relacionados

Berlt et al. (2007) propõem uma adaptação do algoritmo de *PageRank* chamada *HiperPageRank*. O *HiperPageRank* é computado utilizando um hipergrafo ao invés de utilizar o *WebgGaph* utilizado pelo *PageRank*. Neste hipergrafo, as páginas são agrupadas de acordo com algum critério de particionamento – páginas, domínios ou *hosts*, por exemplo – a fim de eliminar *links* "internos" desse particionamento para o cálculo da reputação de páginas *Web*. Nos experimentos realizados (BERLT et al., 2007) com uma base com mais de 12 milhões de páginas *Web*, o *HiperPageRank* obteve melhores resultados para consultas navegacionais do que o *PageRank*.

O modelo de hipergrafos proposto por Berlt et al. é similar ao modelo de *VersionGraph* proposto nesse trabalho. A diferença dos dois modelos é que o *VersionGraph* considera, na análise de *links*, os agrupamentos que representam todos os elementos do cluster como um único vértice, seja nos *links* que apontam para um vértice, seja para *links* apontados pelo vértice em questão. Devido a essa similaridade, os escores propostos por Berlt et al. foram comparados aos escores propostos nesse trabalho.

No trabalho proposto por Carvalho et al. (2006), ao invés de propor uma alteração no algoritmo *PageRank*, é proposta uma série de medidas visando a eliminação de *links* considerados ruidosos entre as páginas. Além disso, os autores propõem uma adaptação do algoritmo de *PageRank*, baseada na independência das páginas que apontam para uma página *Web* (CARVALHO et al., 2006). Esta série de medidas visa a melhoria do cálculo da reputação das páginas *Web*, através da detecção e tratamento de informações ruidosas no *WebGraph* de um repositório de páginas de um motor de busca. Os experimentos comprovam que a detecção e o tratamentos das informações ruidosas trazem uma melhoria em termos de MRR[1] e MAP[2] para as consultas navegacionais populares e para as consultas informacionais populares, respectivamente, em relação ao *PageRank*. As abordagens realizadas por Carvalho et al. para eliminação de *links* ruidosos não incluem a eliminação de *links* entre versões de um mesmo documento *Web*, o que é realizado no modelo de *VersionGraph* proposto neste trabalho.

Analisando a reprodução de conteúdo entre as páginas *Web*, Baeza-Yates et al. (2008) realizaram um estudo introduzindo o conceito de árvores genealógicas na *Web* para explicar a composição de conteúdos das páginas *Web* baseadas em textos já existentes em páginas mais antigas. Este estudo analisa, entre outras coisas, o impacto dessa composição na reputação das páginas *Web*. Este estudo revelou, entre outras coisas, que a ordem dos resultados apresentados pelos motores de busca, contribuem para elaboração de novos conteúdos na *Web*. Essa contribuição deriva da premissa que os resultados mais próximos do topo do *ranking* são os mais clicados pelos usuários e seus conteúdos são copiados para a criação de novas páginas. O estudo estima que 23,7% das páginas que surgem na *Web* no período de um ano, têm conteúdo de páginas já publicadas. O estudo realizado por Baeza-Yates et al. analisa os relacionamentos das páginas através de seus conteúdos, mas não propõe nenhuma melhoria no processo de *ranking* como resultado desse estudo. Nesta dissertação são propostos novos escores de reputação baseado na análise de similaridade do conteúdo das páginas *Web*.

Uma alternativa ao *PageRank* foi proposta recentemente por Liu et al. (2008) que, ao invés de utilizar o *WebGraph* para computar reputação das páginas *Web*, utiliza o grafo formado pela navegação do usuário na *Web* para atribuir reputação as páginas. O escore obtido através da análise do grafo formado pelo histórico de navegação dos usuários, denominado *BrowseRank*, mostrou-se mais eficiente que o *PageRank* nos casos em que existem distorções na análise de *links*. Por exemplo, pode-se considerar o uso de *spams* para melhorar os escores de certas páginas Web.

Apesar de existirem várias propostas de algoritmos de análise de *links* que se baseiam no modelo do algoritmo de *PageRank*, um problema em aberto é considerar as versões de um documento *Web* (páginas *Web*) no momento de atribuir reputação as páginas *Web*, visando uma distribuição mais homogênea da reputação para as diferentes versões do documento *Web*.

---

[1] *Mean Reciprocal Rank* é uma métrica utilizada para consultas navegacionais, pois quanto mais próximo do topo o resultado correto estiver, melhor o resultado obtido.

[2] *Mean Average Precision* é uma métrica utilizada para consultas informacionais e unifica os valores de precisão e revocação em uma única métrica.

2.3 Detecção de quase duplicatas

"Quase duplicatas" (do inglês *near-duplicates*) são páginas *Web* que diferem muito pouco em seus conteúdos, diferença essa quase sempre atribuída ao conteúdo não relevante, ou seja, cabeçalhos, rodapés, informações de versão, etc.

Algoritmos de cálculo de similaridade têm sido utilizados para realizar a detecção de quase duplicatas para melhorar um ou mais processos do motor de busca (*webcrawler*, armazenamento de páginas, apresentação de resultados das buscas). Dois dos principais algoritmos de detecção de quase duplicatas da literatura são baseados em *shingles* que por sua vez definem um *fingerprint* para cada documento da coleção (BRODER et al. 1997; CHARIKAR, 2002). A forma como os *fingerprints* são definidos é o que difere os dois trabalhos. Monica Henzinger (2006) realizou um comparativo entre os algoritmos de Broder et al. (1997) e Charikar (2002) e concluiu que o algoritmo de Charikar é melhor para a tarefa de detecção de quase duplicatas.

A técnica de *fingerprint* proposta por Charikar (2002) se baseia no mapeamento de vetores multidimensionais que representam as características de uma página *Web* em *fingerprints* com tamanho reduzido. Esses *fingerprints*, baseados nas características de uma página *Web*, geram um *hash* para cada página da coleção. A grande diferença do *hash* gerado pelo algoritmo de Charikar é que para páginas com características similares – características nesse caso são as palavras do conteúdo textual das páginas – são gerados *hashs* similares, ou seja, para páginas similares são gerados *hashs* que diferem poucos *bits* um dos outros. Essa característica do algoritmo de Charikar difere de outros algoritmos tais como MD5 e SHA-1 que geram *hashs* totalmente diferentes para páginas com conteúdo textual similar.

Manku et al. (2007) utiliza o algoritmo de *fingerprints* proposto por Charikar (CHARIKAR, 2002) para a detecção de quase duplicatas com o intuito de melhorar o desempenho do processo de *WebCrawler*.

Os experimentos apresentados no trabalho de Manku et al. (2007) demonstram que o algoritmo de Charikar (2007) pode ser utilizado eficientemente mesmo em repositórios contendo bilhões de páginas *Web*. O algoritmo de Charikar foi o algoritmo utilizado nesta dissertação por ter comprovado sua eficiência na detecção de quase duplicatas, e também porque o custo computacional do algoritmo não inviabiliza a detecção de versões em grandes coleções.

A detecção de réplicas também pode ajudar o estudo da evolução do conteúdo da *Web*. O algoritmo de Cho et al. (2000) propõe um método eficiente para identificar páginas *Web* replicadas em uma coleção. O estudo de Baeza-Yates et al. (2008) utiliza uma adaptação do algoritmo de Cho et al. (2000) para detectar quais páginas compõem a árvore genealógica de composição de cada página na Web.

3 ABORDAGEM PROPOSTA

Neste capítulo é detalhada a abordagem proposta para a utilização da detecção de versões para melhorar a qualidade da atribuição de reputação às páginas *Web*. O capítulo inicia apresentando uma visão geral do trabalho através da seção 3.1. A seção 3.2 detalha o mecanismo de detecção de versões e a sua parametrização através de experimentos. Em seguida, a seção 3.3 explica o modelo denominado *VersionGraph*. Por fim, a seção 3.4 apresenta os escores propostos baseados na detecção de versões.

3.1 Visão Geral

O principal objetivo deste trabalho é melhorar o posicionamento das diferentes versões de um mesmo documento *Web* no *ranking* de resultados dos motores de busca. A abordagem proposta é baseada na reputação de todas as versões do documento em questão. Como contribuição, são especificados e validados novos escores de reputação baseados na detecção de versões de páginas *Web*.

Os escores propostos visam atribuir às versões de um dado documento *Web* um valor de reputação que represente o documento e não cada uma de suas versões isoladamente (ou seja, páginas *Web* que são versões de um mesmo documento). Especificamente, são propostos quatro novos escores, classificados em duas categorias: os que realizam mudanças estruturais no *WebGraph* (denominados, respectivamente, *VersionRank* e *VersionPageRank*) e os que realizam operações aritméticas sobre os escores obtidos pelo algoritmo de *PageRank* (denominados, respectivamente, *VersionAverageRank* e *VersionSumRank*).

Os escores propostos utilizam um índice para determinar quais páginas *Web* compõem um documento. Esse índice é o resultado do processo de detecção de versões, cujo único objetivo é fornecer ao processo de atribuição de escores uma fonte de fácil acesso para determinar quais páginas *Web* são versões de um mesmo documento.

A Figura 3.1 apresenta uma visão geral da abordagem proposta. A partir do repositório de páginas *Web* é montada a estrutura denominada *WebGraph*, que mantém em uma estrutura de grafo direcionado a estrutura de *links* do repositório de páginas. No *WebGraph*, os vértices representam as páginas e seus *links* as arestas direcionadas entre as páginas. O detector de versões, que também tem como base o repositório de páginas *Web*, gera um índice de versões que serve para determinar quais páginas *Web* são versões de outras páginas. O índice de versões e o *WebGraph* são utilizados para montar o *VersionGraph*, onde as versões de um mesmo documento são representadas como um único vértice no grafo direcionado formado pelas páginas *Web* e seus *links*. A partir do *VersionGraph* é possível, então, realizar a atribuição do escore *VersionRank*. Os escores

VersionSumRank e *VersionAverageRank* são determinados com base no escore de *PageRank* e no índice de versões.

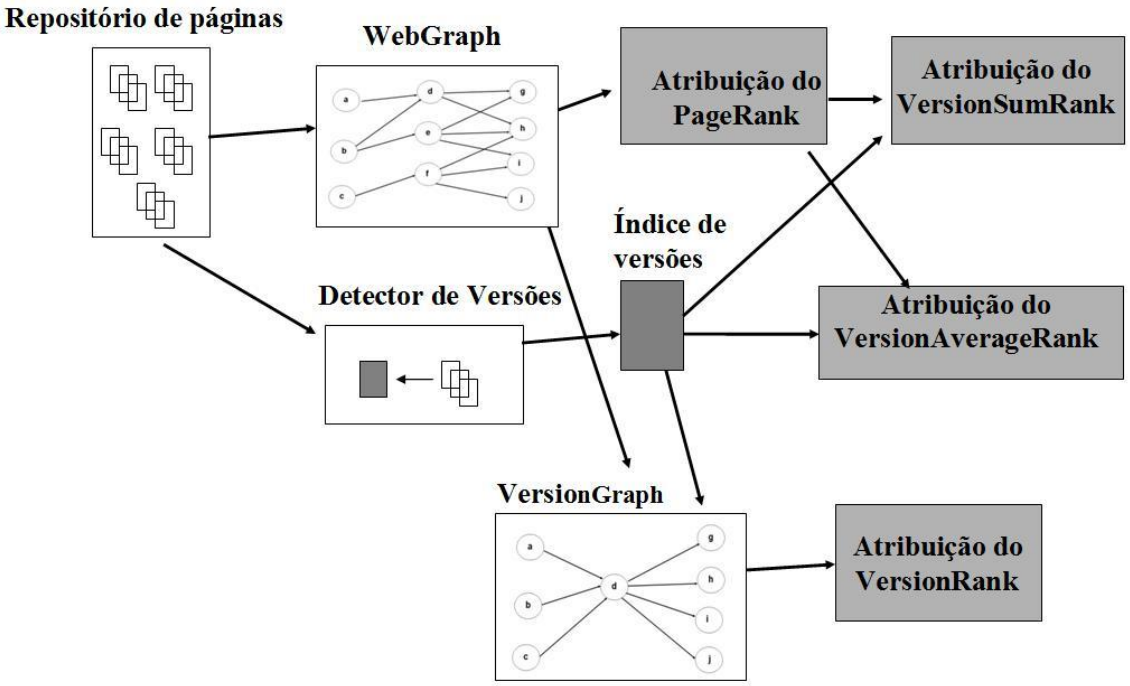

Figura 3.1: Atribuição de escores de reputação e suas dependências

As próximas subseções descrevem detalhadamente cada um dos componentes da abordagem proposta.

3.2 Detecção de Versões

A detecção de versões tem como objetivo gerar um índice de fácil acesso no qual seja possível determinar rapidamente quais são as páginas *Web* que compõem um documento. A detecção de versões indica como as versões das páginas *Web* estão agrupadas, determinando a existência de um documento lógico. Ao longo dessa seção, são detalhados o processo proposto para a detecção de versões, o algoritmo escolhido, sendo ainda descritos os experimentos para a calibragem dos parâmetros do algoritmo escolhido para detecção de versões.

O processo de detecção de versões deve, a partir de um repositório de páginas, gerar um índice de versões, conforme ilustrado na Figura 3.2. O detector de versões deve determinar, através do índice de versões, a qual documento lógico uma página *Web* pertence e quais páginas *Web* compõem um documento lógico. Na Figura 3.2 é possível observar que através do índice de versões é possível determinar, através de uma lista ordenada de todos os *ids* das páginas, a qual documento lógico cada uma pertence. Além disso, a partir de uma lista ordenada dos *ids* dos documentos lógicos e uma lista encadeada, é possível determinar quais páginas *Web* compõem um documento lógico.

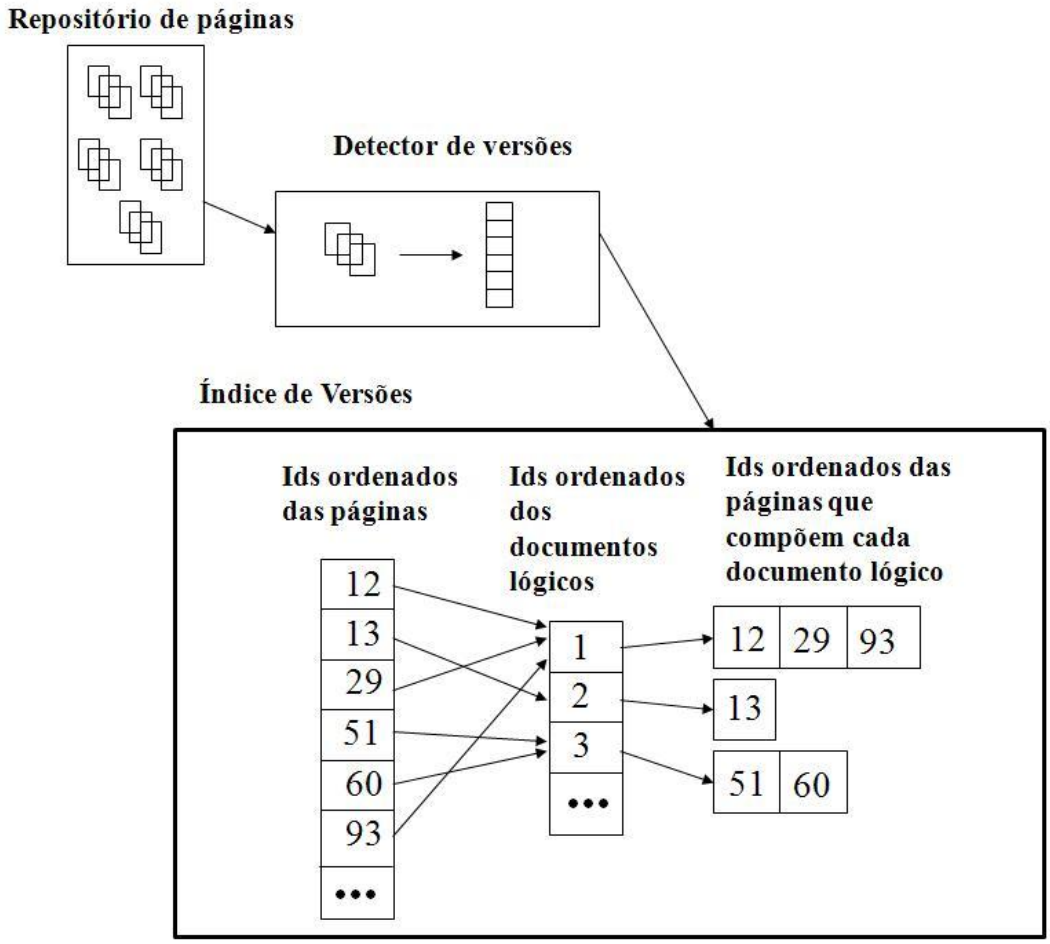

Figura 3.2: Detalhamento do Índice de Versões

Para realizar a tarefa de detecção de versões, foi escolhido o algoritmo de *fingerprints* proposto por Charikar (2002).

Os *shingles*, que determinam o valor do *hash* de cada página no algoritmo de Charikar (2002), podem ser derivados de muitas características das páginas *Web*. No caso desse trabalho, o uso do próprio conteúdo textual das páginas para geração dos *shingles* se mostra como a escolha mais lógica devido à natureza da detecção de versões.

De posse dos *hashs* de todas as páginas de uma coleção, cria-se uma lista dos identificadores (*ids*) das páginas ordenadas pelo *hashs*. O passo seguinte é determinar qual o limiar *k* que representa a diferença máxima de *bits* entre dois *hashs* de páginas, para que as mesmas sejam consideradas versões de um mesmo documento. Esse limiar *k* é comumente referido na literatura como Distância de *Hamming*. Para quase duplicatas, Manku et al. (2007) fixaram o valor de k=3.

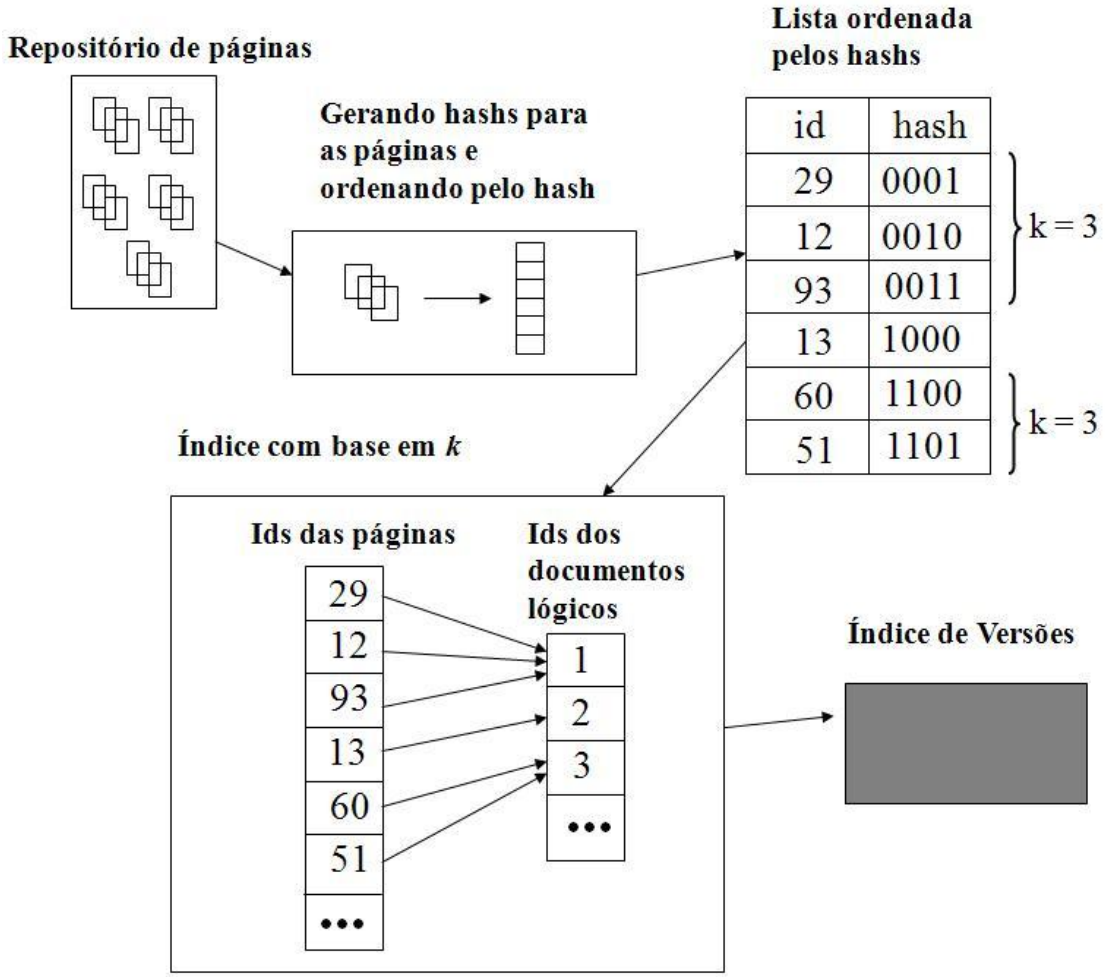

Figura 3.3: Gerando um índice de versões a partir do repositório de páginas

Com a definição de um determinado limiar, por exemplo 3, pode-se percorrer a lista de *hashs* ordenados, determinando quais páginas são versões de um mesmo documento lógico. Com essa informação, é construído o índice que indica, para cada página *Web*, quais são todas as suas versões dentro da coleção.

A Figura 3.3 simula a execução do processo de geração do índice de versões. A simulação apresenta, detalhadamente, uma lista ordenada de *hashs* que é fruto de um processamento efetuado sobre um repositório de páginas fictício. A partir da lista ordenada de *hashs*, é montado um índice baseado nessa lista e no limiar k=3. Pode-se observar no índice, com base em *k*, que a atribuição de uma página a um documento lógico é determinada pela distância *k* máxima entre as páginas *Web*. A partir do índice baseado em *k*, é montado o índice de versões definitivo. Esse índice tem a mesma estrutura que aquela determinada no índice de versões da Figura 3.2

Para tornar a comparação entres os *hashs* de cada página da coleção viável, os mesmos foram ordenados e comparados com os valores próximos da lista. Esta medida que visa melhorar a desempenho do método, acaba por comprometer a revocação, visto que *hashs* com que diferem poucos bits, porém bits significativos, não serão detectados como versões.

O custo computacional dessa detecção é muito inferior aos tradicionais métodos de detecção de versões, pois os mesmos precisam comparar todos os documentos entre si, o que leva a uma complexidade quadrática. Para o método de detecção de versões utilizado, é importante informar que a complexidade computacional média para a montagem do índice de versões e os respectivos modelos provenientes do mesmo é da ordem de *O(nlogn)*, visto que o maior custo envolvido é o da ordenação das páginas pelos *hashs* gerados.

3.2.1 Validação Experimental para a Detecção de Versões

Os experimentos dessa seção visam determinar os parâmetros adequados para que o algoritmo de *fingerprints* possa realizar a detecção de versão de forma satisfatória. Os experimentos foram desenvolvidos com o intuito de determinar a variável Distância de *Hamming* (*k*) e o tamanho da frase utilizada para compor cada *shingle* (*m*), a serem utilizadas na detecção de versão utilizando o algoritmo proposto por Charikar (2002).

Para realizar os experimentos foi utilizada a base da Wikipédia[3]. Um requisito para conseguir medir a qualidade da detecção de versões é que os experimentos sejam realizados em uma coleção na qual as versões sejam identificáveis. A Wikimedia disponibiliza o conteúdo textual dos artigos da Wikipédia em intervalos de tempos regulares o que possibilita verificar a evolução do conteúdo textual dos artigos ao longo do tempo na Wikipédia.

Para simular o efeito de versionamento, foram utilizadas três coleções da Wikipédia em diferentes datas: 2 de março de 2008, 20 de junho de 2008 e 07 de novembro de 2008. Dessa forma é possível considerar que, para cada artigo da primeira coleta, serão encontradas respectivas versões ou réplicas nas coletas subsequentes. Considera-se, portanto, que as três coleções são na verdade uma só e que para cada artigo da primeira coleta deve ser detectado as suas duas versões nas outras duas coletas.

Os dados quantitativos mais significantes sobre as coleções dizem respeito a primeira coleção (março de 2008), pois é a partir desta coleção que serão realizados os experimentos. A coleção de março de 2008 apresenta 1.379.985 artigos, sendo que destes 210.799 apresentam tamanho superior a 1.500 *bytes*.

Considerando as três coleções como sendo uma única coleção, foi aplicado o detector de versões à coleção. As métricas utilizadas foram a precisão e a revocação, da seguinte forma: para cada documento pertencente originalmente a primeira coleta, deveriam ser detectados como versões somente os mesmos documentos da coleta 2 e 3 (ou seja, documentos com os mesmos *ids*), e, somente nesse caso, os valores de precisão e revocação deveriam ser 1. Os dados apresentados nesta seção representam a média da precisão e da revocação para todos os documentos da primeira coleta.

O gráfico da Figura 3.4 apresenta a variação da precisão obtida para cada tamanho de frase (*shingle, m*) em função do limiar de detecção (distância de *Hamming, k*). Observa-se no gráfico o comportamento detectado por Manku et al. (2007), que quanto menor o tamanho da frase, mais documentos não similares parecerão similares, ou documentos que não são versões pareceram versões nesse caso. Ainda no gráfico da Figura 3.4 é possível observar uma leve queda na precisão para frases menores que 5 quando é usado o limiar de detecção 10 e que se acentua no limiar de detecção 15.

[3] http://download.wikimedia.org/ptwiki/

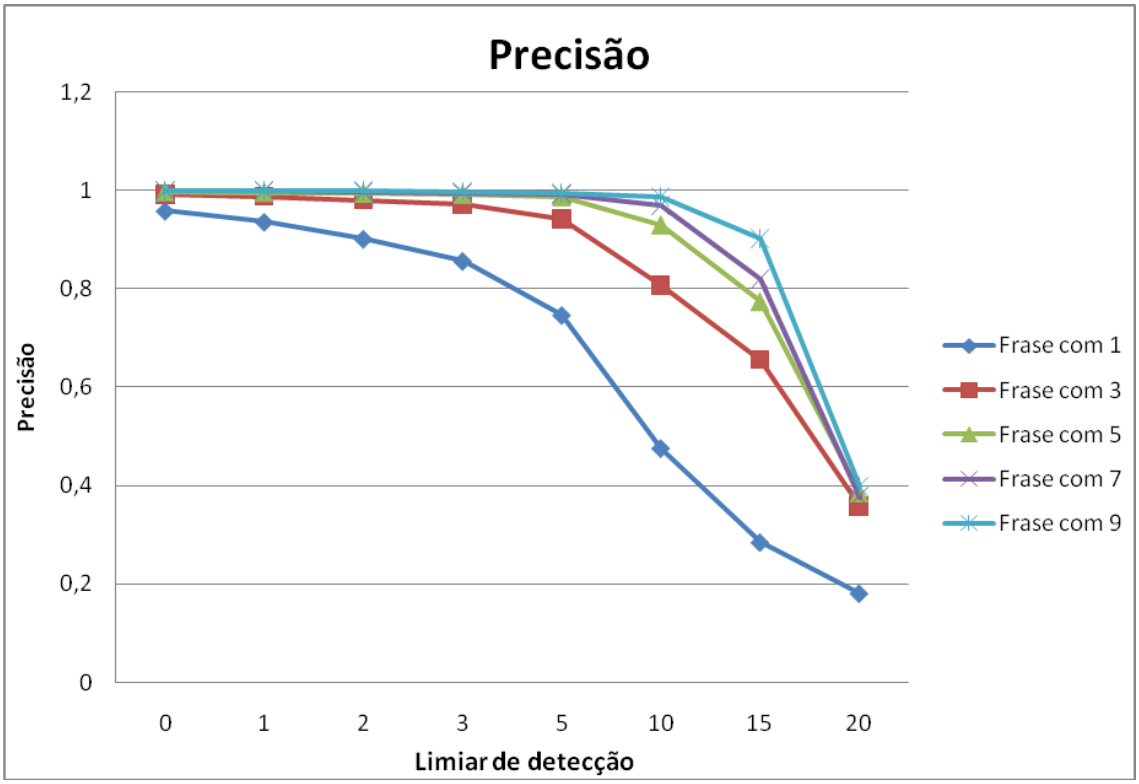

Figura 3.4: Precisão da detecção de versões na base Wikipédia.

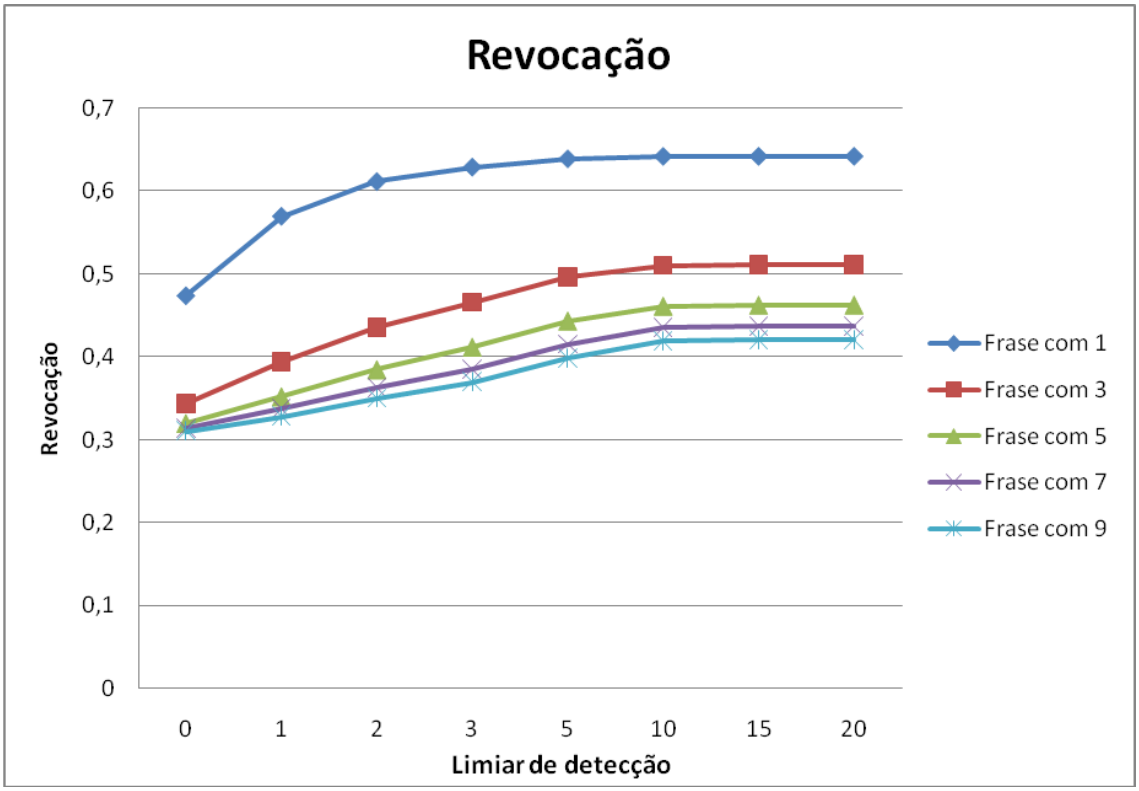

Figura 3.5: Revocação da detecção de versões na base Wikipédia.

No gráfico da Figura 3.5 é apresentada a variação da revocação obtida para cada tamanho de frase em função do limiar de detecção. Nota-se no gráfico que, ao contrário da precisão, têm-se melhores valores de revocação com o aumento do limiar de

detecção. Esse comportamento é o natural, pois, à medida que o limiar de detecção de versões é aumentado, mais documentos são detectados como versões, porém com uma menor precisão. Observando esse comportamento foi escolhido como tamanho de frase o valor de 5, definindo então m=5.

Já o limiar de detecção, ou Distância de *Hamming* (k), pode ser fixado em 10, já que com esse limiar temos a precisão de 0,93 e a revocação em 0,46. Porém, esse limiar foi deixado variável, visto que a qualidade das páginas da coleção Wikipédia é muito superior às páginas de uma coleção da *Web*, que possui conteúdo de sites das mais diversas origens. Essa maior qualidade se dá pelo conteúdo ser regrado e a inexistências de cabeçalhos, rodapés, propagandas, etc., que podem impactar na qualidade dos textos das páginas *Web*.

Cabe ressaltar que definições adequadas de valores de limiares são discutidas na literatura e não fazem parte do escopo do trabalho apresentado neste trabalho (Silva et al., 2007).

Apesar da detecção de versões ser uma parte importante para a qualidade dos resultados obtidos, a proposta é flexível para a adoção de outros algoritmos de detecção de versões de páginas *Web* sem nenhuma alteração da mesma.

3.3 VersionGraph

O modelo que melhor representa as páginas *Web* de uma coleção e os relacionamentos entre as mesmas através de seus *links* é denominado *WebGraph*. O *WebGraph* é um grafo direcionado cujos vértices representam as páginas e as arestas os *links* entre as mesmas. A Figura 3.6 ilustra um *WebGraph* no qual as letras representam as páginas e os números o escore *PageRank* a elas atribuídas.

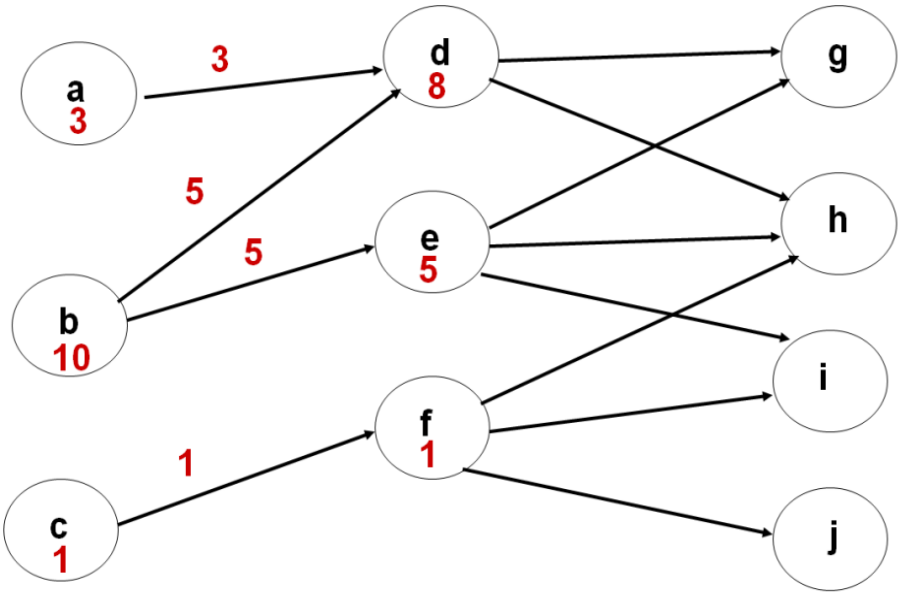

Figura 3.6 *WebGraph* e suas respectivas atribuições de escore de reputação às páginas.

Observa-se na Figura 3.6 que as páginas *d, e* e *f* têm respectivamente 8,5 e 1 como escores de *PageRank*, baseado na reputação atribuída pelas páginas que apontam para as mesmas, conforme o algoritmo de *PageRank* detalhado na seção 2.1.

A partir do índice de versões definido na seção 3.2, as páginas *Web* podem ser agrupadas em *clusters* de modo que todas as páginas de cada *cluster* são versões de um mesmo documento lógico e que cada *cluster* representa um documento *Web*.

De posse de quais páginas *Web* são versões de um mesmo documento lógico é construído um *WebGraph* no qual cada vértice representa um documento lógico e não uma página *Web*. Esse novo *WebGraph* é denominado *VersionGraph*.

A Figura 3.7 ilustra a lógica da transformação de um *WebGraph* (Figura 3.7.A) em um *VersionGraph* (Figura 3.7.C) a partir da detecção de versões (Figura 3.7.B). As páginas *e* e *f* são detectadas e representadas como versões da página *d*.

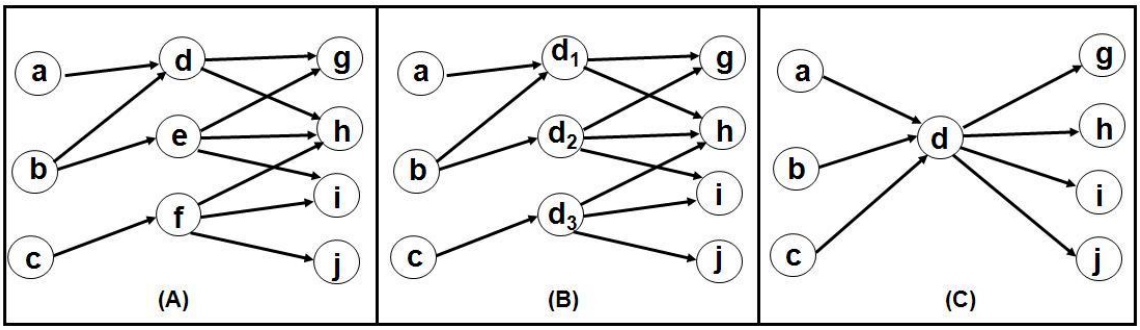

Figura 3.7: *WebGraph* baseado em páginas (A), no qual é aplicada a detecção de versões (B), resultando no *VersionGraph* (C).

3.4 Escores de Reputação baseados na detecção de versões

Esta seção descreve os quatro novos escores propostos para melhorar o posicionamento de versões de um mesmo documento no *ranking* de resultados dos motores de buscas. A atribuição desses novos escores é efetuada levando em consideração o escore de todas as versões do documento, sendo que essas versões podem, muitas vezes, estarem localizadas em diferentes sites e/ou domínios na Internet.

Considerando uma página *d* bem ranqueada, podem surgir novas versões dessa mesma página em diferentes localidades da Internet: *d1, d2 ... dn*. Essas novas versões de *d* não são detectadas a *priori* como versões de *d*, e, por isso, as mesmas não terão o mesmo escore de *d*, mesmo tratando-se de versões de um mesmo documento na *Web*. De outra forma, a distribuição dos *links* entre as versões de *d* pode ainda trazer prejuízos ao escore de reputação de *d*, pois os *links* que atribuem reputação a *d* estão espalhados pelas suas versões.

Considere, agora, que as páginas *d, e* e *f*, representadas no WebGraph da Figura 3.6, são versões de um mesmo documento. Neste caso, diversas alternativas podem ser utilizadas para atribuir uma reputação mais homogênea para as páginas, visto que se tratam de versões de um mesmo documento.

A atribuição de uma reputação mais homogênea às páginas passa pela definição de escores que utilizem a informação do índice de versões para atribuir reputação às páginas com base em todas as suas versões. Todos os escores propostos neste trabalho têm a finalidade de tornar mais homogêneos os escores de páginas que são versões de

um mesmo documento. Isso é feito atribuindo um peso único de reputação a todas as versões do documento.

Foram definidos quatro escores que utilizam a informação do índice de versões com três abordagens diferentes. O primeiro escore, denominado *VersionRank,* promove mudanças estruturais no *WebGraph* a fim de representar as versões de um documento lógico como um único vértice no grafo. O segundo escore, denominado *VersionPageRank,* utiliza o escore *VersionRank* somente para páginas que são versões e mantém o escore *PageRank* para páginas únicas na coleção. O terceiro e o quarto escores utilizam operações aritméticas para atribuir uma reputação homogênea para páginas que são versões de um mesmo documento. Essa abordagem é utilizada nos escores *VersionSumRank* e *VersionAverageRank.*

Os escores de reputação propostos nesse trabalho podem ser utilizados em qualquer coleção de páginas *Web* e tem seu uso indicado quando nessas coleções houver a incidência de páginas que podem ser agrupadas como versões de um único documento lógico ou mesmo quando existe a incidência de quase duplicatas. A maioria das coleções Web reais têm a incidência de versões de páginas *Web* que na sua maioria são consideradas como documentos únicos na coleção, o que está semanticamente inadequado.

No seguimento desta seção, o funcionamento de cada um dos escores é detalhadamente descrito.

3.4.1 VersionRank

O objetivo do *VersionRank* é a atribuição de escores de reputação a documentos lógicos na *Web* e não a páginas *Web*. A premissa para a atribuição do escore *VersionRank* é a construção do *VersionGraph* (vide seção 3.3).

Com base no modelo de *VersionGraph* é possível realizar a atribuição de escores do algoritmo de *PageRank* a partir de um grafo que represente somente documentos do mundo real e não várias versões de um mesmo documento. A partir dessa atribuição de escores, baseada no *VersionGraph* (documentos) e não físico (páginas), pode-se atribuir um valor de *PageRank* a um documento lógico e não a um conjunto de páginas que na verdade são versões de um mesmo documento lógico.

A fórmula para definição do escore *VersionRank* é igual à fórmula de definição do *PageRank* (seção 2.1), porém ao invés de considerar as páginas do *WebGraph* na computação do escore, a fórmula considera os documentos do *VersionGraph* para o cálculo.

Esse valor de *PageRank* atribuído a um documento lógico é denominado *VersionRank.* O *VersionRank* é um valor que atribui um escore a documentos da *Web*, independente de quantas versões ou réplicas esses documentos tenham em diferentes sites. Ao ordenar os resultados de uma busca considerando o valor de *VersionRank*, garante-se que todas as versões de um mesmo documento estejam em posições próximas no *ranking*.

Por exemplo, considere o grafo apresentado na Figura 3.6. Observa-se que o voto de reputação dado pela página web *c* para a página *f* é de pouco peso, e que os votos de reputação das páginas *a* e *b* tem um peso significativo. Nesse cenário, utilizando o escore *PageRank*, o posicionamento no *ranking* de resultados da página *f* - em um

universo de outras páginas com conteúdo semelhante e reputação - estaria bem distante das páginas *d* e *e*.

Entretanto, ao considerar que as páginas *d*, *e* e *f* são versões de uma mesma página e representá-las no modelo de *VersionGraph* (Figura 3.8) garante-se que estas páginas estarão em posições próximas no *ranking* já que as mesmas terão o mesmo valor de *VersionRank*.

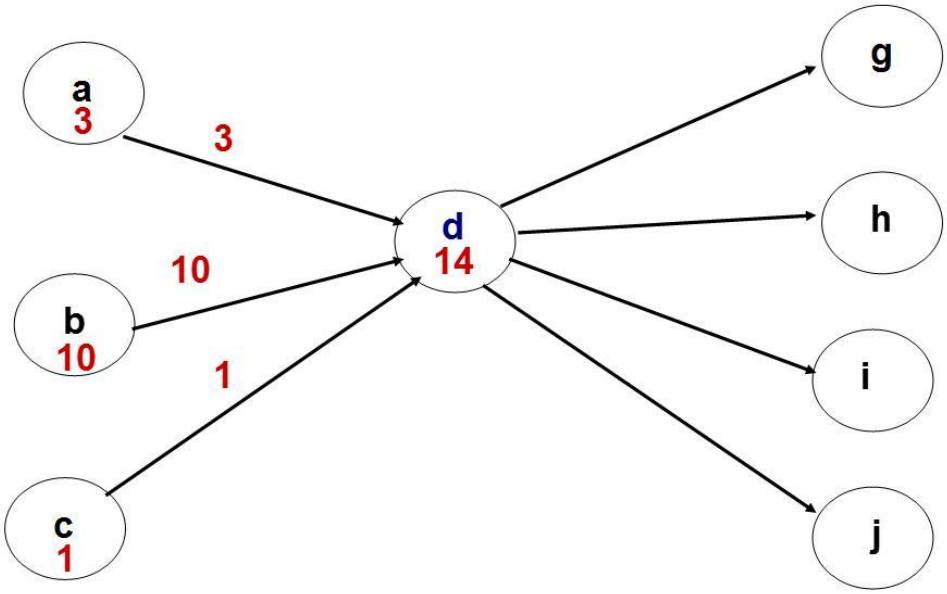

Figura 3.8: VersionGraph, e a atribuição do escore VersionRank.

Cabe ressaltar que o valor de *VersionRank* de uma página que não tenha versões não é igual ao valor de *PageRank* da mesma. Isso ocorre devido a mudança substancial ocorrida ao transformar o *WebGraph* em um *VersionGraph* (Figura 3.7). Como os grafos podem se tornar bem diferentes, a computação dos escores também terá diferenças.

Da mesma forma, vale acrescentar que o exemplo apresentado é meramente didático, visto que as páginas *a*, *b* e *c* também teriam seus escores de *VersionRank* alterados, pois o *VersionGraph* da coleção que as mesmas pertencem viria a atribuir para as páginas valores de reputação diferentes.

O *VersionRank* visa atribuir uma reputação mais correta aos documentos lógicos da *Web* porque faz sua atribuição baseada no modelo *VersionGraph*, porém tende a favorecer páginas que tenham versões.

Além disso, o modelo de *VersionGraph* elimina *links* ruidosos entre páginas, visto que só são representados no modelo *links* entre os documentos lógicos e não *links* entre versões de um mesmo documento.

3.4.2 VersionPageRank

O escore *VersionPageRank* tem como objetivo atribuir um escore de reputação às páginas *Web* visando aproveitar o desempenho do *VersionRank* para páginas que tenham versões e mantendo o escore *PageRank* para páginas que são únicas na coleção.

A atribuição do escore *VersionPageRank* segue os seguintes passos:

1. se a página é única na coleção, ou seja, não existem diferentes versões da mesma, é atribuída para a mesma o escore *PageRank*;

2. se a página tem outras versões dentro da coleção analisada, é atribuída a ela o escore *VersionRank*.

Dessa forma, a lista de resultados do motor de busca é ordenada de acordo com os dois escores: *VersionRank* e *PageRank*.

Este tipo de discriminação – atribuir dois escores com pesos diferentes para páginas de uma mesma coleção – irá atribuir reputações com significado diferente aos dois conjuntos de páginas, visto que os escores são baseados em *WebGraphs* que têm uma semântica diferente. Na prática, o *VersionPageRank* irá tornar maior o escore atribuído as páginas que têm versões e manter o escore *PageRank* para páginas sem versões. Esse pode ser um comportamento indesejado e será analisado, juntamente com outros comportamentos, durante os experimentos apresentados no próximo capítulo.

3.4.3 VersionSumRank

É a soma do *PageRank* das páginas que compõem um mesmo documento. A atribuição desse escore considera que todas as páginas que são versões devem ter o mesmo escore. A soma é utilizada para aumentar o valor do escore de todas as versões de um mesmo documento. O objetivo é alavancar a reputação de todas as versões de um documento, atribuindo um valor elevado a todas as versões do documento *Web*. *VersionSumRank* (VSR) é definido como:

$$VSR(pi) = \sum_{pj \in V(pi)} PR(pj)$$

onde, *VersionSumRank* (VSR) da página *pi* é definido como a soma de todos os *PageRank* (PR) das páginas que são versões de *pi* e *V(pi)* representa esse conjunto de versões. De forma geral, pode-se dizer que o VSR de uma página *pi* é a soma de todos os PRs das páginas que são detectadas como versões de *pi*, inclusive o próprio PR da página *pi*.

Considere, por exemplo, o grafo da Figura 3.6. As páginas *d*, *e* e *f* são versões do mesmo documento. Nesse caso o *VersionSumRank* das páginas *d*, *e* e *f* é 14, tornando homogêneo o escore de todas as versões, fazendo que os mesmos estejam em posições próximas no *ranking* e também em posições mais próximas ao topo do *ranking*, visto que o escore *VersionSumRank* das versões será maior que de possíveis páginas com conteúdo também relevante porém com uma única versão na coleção.

O *VersionSumRank* atribui um escore de reputação mais elevado as páginas *Web* que tenham versões na coleção. Páginas relevantes que tenham versões terão sua reputação aumentada. Entretanto, esse comportamento pode prejudicar páginas relevantes que sejam únicas na coleção e favorecer páginas não relevantes que tenham versões.

3.4.4 VersionAverageRank

É a média do *PageRank* das páginas que são versões de um mesmo documento. O escore *VersionAverageRank* atribui uma mesma reputação para todas as versões de um

documento, resultando em uma distribuição mais homogênea do escore de *PageRank*. A seguinte fórmula define o *VersionAverageRank* (VAR):

$$VAR(pi) = \frac{VSR(pi)}{TV(pi)}$$

onde, *VersionAverageRank* (*VAR*) da página pi é definido como o quociente da divisão entre *VersionSumRank* (*VSR*) de *pi* e o número total de versões de *pi*, representado na fórmula por *TV(pi)*. De forma geral, o VAR de uma página *pi* é a média de todos os PRs das páginas que são versões de *pi*, nessa média entrando também o valor do PR de *pi*.

Considere, por exemplo, o grafo da Figura 3.6. As páginas *d*, *e* e *f* são versões do mesmo documento. O *VersionAverageRank* das páginas *d*, *e* e *f* é 4,66, e da mesma forma que o VSR, o VAR torna homogêneo o valor atribuído a todas as versões das páginas e garante a proximidade das versões no ranking de resultado, porém não alavanca o valor do escore baseado em quantas versões a página tem como faz o VSR.

4 VALIDAÇÃO EXPERIMENTAL

Este capítulo apresenta os experimentos para validação dos escores propostos baseados na detecção de versão. Foram realizados experimentos sobre duas coleções que representam dados de um motor de busca real: WBR99 e WBR03. Os experimentos realizados em ambas as coleções têm o objetivo de verificar a eficiência dos escores propostos em comparação com o principal escore de reputação da literatura: o *PageRank*. Para medir o impacto da ordenação dos resultados dos motores de busca utilizando os escores propostos, foram utilizadas as métricas da área de recuperação de informação para a *Web*.

Este capítulo está organizado da seguinte forma. A seção 4.1 apresenta o projeto experimental composto pelas métricas de avaliação, a descrição das coleções e a metodologia utilizadas. A seção 4.2 analisa os experimentos de comparação entre os escores propostos e o *PageRank*, sobre a WBR99. Já a seção 4.3 realiza os mesmos experimentos sobre a coleção WBR03, comparando os resultados obtidos com o desempenho do *PageRank* e dos escores propostos por Berlt et al. (2007). O capítulo é finalizado na seção 4.4 com uma análise geral dos resultados obtidos nos experimentos.

4.1 Projeto Experimental

Esta seção apresenta as métricas utilizadas para medição dos resultados obtidos pelos escores propostos, os dados gerais dos dois ambientes sobre os quais foram executados os experimentos, e a metodologia utilizada na execução dos experimentos.

4.1.1 Métricas de avaliação

Antes de apresentar as métricas de avaliação para motores de busca é necessário definir os tipos de consultas, pois, para cada tipo de consulta, são utilizadas métricas diferentes. Broder (2002) propõem a divisão das consultas em três tipos: consultas navegacionais, consultas informacionais e consultas transacionais. Neste trabalho só foram utilizadas as duas primeiras. Consultas navegacionais são consultas que o usuário deseja ir a algum site e um resultado correto já satisfaz a consulta. Consultas informacionais têm como objetivo obter informações sobre um tópico específico, portanto mais de um resultado podem ser relevantes para o usuário. Apresentados os dois tipos de consultas utilizados nos experimentos, são definidas as métricas utilizadas no seguimento do trabalho.

A precisão é a métrica que mede qual o percentual de documentos relevantes no universo de documentos retornados pelo motor de busca. A precisão é dada pela seguinte fórmula:

$$precisão = \frac{DocumentosRelevantes \cap DocumentosRecuperados}{DocumentosRecuperados}$$

A precisão considera todos os documentos retornados pelo motor de busca. Entretanto, uma adaptação bastante utilizada para motores de busca para Web é considerar um ponto de corte nos documentos retornados, devido ao número elevado de documentos retornados. Desse corte, surgem as métricas denominadas *P@N (precision at N)*, onde *N*, é o ponto de corte. Por exemplo, a métrica *P@10* considera, para cálculo da precisão, somente os dez primeiros documentos retornados pelo motor de busca. Nesse caso, se entre esse 10 documentos, encontram-se 3 relevantes, a P@10 é 3/10, ou seja, 0,3. A métrica *P@10* foi utilizada nos experimentos que envolvem comparações entre os escores propostos nesse trabalho e alguns já existentes na literatura.

A revocação mede qual o percentual de documentos relevantes recuperados por uma pesquisa perante o número total de documentos relevantes na coleção. A revocação é calculada pela seguinte fórmula:

$$revocação = \frac{DocumentosRelevantes \cap DocumentosRecuperados}{DocumentosRelevantes}$$

A revocação é utilizada em conjunto com a precisão para calcular a MAP (*Mean Average Precision*). A métrica *Mean Average Precision* unifica em um mesmo escore os valores clássicos da recuperação de informação: precisão e revocação. A métrica MAP é utilizada para avaliação de consultas informacionais. A MAP é dada pela média das precisões médias (*average precision*) de cada consulta entre as consultas que estão sendo avaliadas. Por sua vez, a precisão média de uma consulta é a média das precisões calculadas após cada documento relevante recuperado.

$$MAP = \frac{\sum_{q \in Q} AP(q)}{|Q|}$$

Na fórmula, Q é o numero de consultas avaliadas e *AP(q)* é a precisão média da consulta *q*.

A métrica *Mean Reciprocal Rank* (MRR) é utilizada quando são submetidas consultas navegacionais aos motores de busca. A métrica valoriza a posição no ranking do resultado considerado correto, quanto mais próximo ao topo, melhor o valor obtido por essa métrica. A MRR é calculada através da seguinte fórmula:

$$\mathrm{MRR} = \frac{1}{|Q|} \sum_{i=1}^{Q} \frac{1}{\mathrm{rank}_i}.$$

Na fórmula, Q é o número que consultas avaliadas e *rank* é a posição no *ranking* do resultado considerado correto para cada consulta. Por exemplo, dispondo de 3 consultas para avaliar um motor de busca e sabendo que o resultado relevante da primeira consulta está na posição 4, o da segunda consulta está na posição 5 e o da terceira está na posição 1, chegar-se ao seguinte cálculo: (1/4 + 1/5 + 1/1)/3 = 0,483, ou seja, 0,483 é o valor de MRR para as três consultas avaliadas sobre o motor de busca avaliado

Para medir a qualidade dos resultados obtidos quando a lista de páginas (contendo páginas relevantes e não relevantes) é ordenada seguindo os diferentes escores

propostos, foram utilizadas as diferentes métricas conforme o tipo de consulta. A métrica MRR valoriza a posição no *ranking* do resultado considerado correto, quanto mais próximo ao topo, melhor o escore obtido e por isso ela é utilizada em consultas navegacionais.

Para consultas informacionais foram utilizadas as métricas *Mean Average Precision* (MAP) e *Precision at 10* (P@10). Serão comparados os resultados obtidos pelas consultas quando forem utilizados os diferentes escores propostos nesse trabalho.

4.1.2 Descrição das Coleções

As duas coleções utilizadas nos experimentos (WBR99 e a WBR03) têm origem nas bases de dados do extinto motor de busca real denominado TodoBR[4].

A WBR99 representa uma coleta de páginas realizada na *Web* brasileira em novembro de 1999. Ao todo são 5.939.061 páginas na coleção, 2.669.965 termos e 40.871.504 links entre os documentos.

Tabela 4.1: Dados gerais da WBR99.

Número de páginas	5.939.061	Número de consultas disponíveis	33.154
Número de termos	2.669.965	Número de consultas avaliadas	50
Número de *links*	40.871.504	Número de páginas avaliadas	4.117

A Tabela 4.1 apresenta um apanhado geral dos dados da WBR99. O número de consultas avaliadas é um conjunto de consultas selecionado dentre as consultas mais frequentemente submetidas ao motor de busca no mês de novembro de 1999. Para cada consulta, foi gerada uma lista de páginas de acordo com os algoritmos descritos em Calado et al. (2003). Para todas as páginas da lista foi avaliada a relevância por um grupo de 29 pessoas familiarizadas com buscas na *Web*. Devido à natureza das consultas da WBR99, elas foram classificadas como informacionais.

A WBR03 é uma coleção baseada no mesmo motor de busca da WBR99, porém coletada no ano de 2003. O número de páginas da WBR03 é mais que o dobro que a WBR99, são 12.020.513 páginas. Também se observa um aumento considerável no número de *links*, são 130.717.004. O tamanho médio do texto das páginas da WBR03 é de 5kb.

Tabela 4.2: Dados gerais da WBR03.

Número de páginas	12.020.513
Número de *links*	130.717.004
Tamanho médio do texto das páginas	5kb

A Tabela 4.2 apresenta os dados gerais da WBR03. As consultas sobre a WBR03 também foram extraídas do *log* de consultas do motor de busca e classificadas em dois grupos: consultas navegacionais e consultas informacionais. As consultas navegacionais

[4] TodoBR é uma marca registrada de Akwan Information Technologies, que foi adquirida pelo Google em Julho de 2005.

contam com um conjunto de 74 consultas avaliadas. As consultas informacionais foram subdivididas em mais dois subgrupos: um contendo consultas populares e outro contendo consultas aleatórias. A relevância ou não do conjunto de páginas retornadas para cada consulta foi avaliado por um conjunto de 15 pessoas. Ao final, para a WBR03 chegaram-se aos seguintes números para cada tipo de consulta: 62 consultas informacionais populares, 62 consultas informacionais aleatórias e 74 consultas navegacionais.

Os experimentos realizados sobre a coleção WBR99 têm como objetivo medir a eficiência dos escores propostos em comparação com o escore PageRank, para consulta informacionais, utilizando as métricas MAP e P@10. Já os experimentos realizados sobre a coleção WBR03 têm como objetivo medir a eficiência dos escores deste trabalho perante o escore *PageRank* e também aos escores propostos por Berlt et al. Sobre a WBR03 foi possível realizar a comparação com os escores de Berlt et al. porque os dados dos experimentos realizados sobre esta coleção foram publicados. Além disso, a coleção WBR03 também disponibiliza uma série de consultas navegacionais, o que possibilitou a comparação para este tipo de consulta, utilizando a métrica MRR.

4.1.3 Metodologia

Para a análise do desempenho dos escores propostos foi seguida a seguinte metodologia: para cada consulta, a lista de documentos, na qual cada página foi considerada como relevantes ou não relevantes, foi ordenada tendo como critério de ordenação o escore a ser avaliado no momento. A partir dessa lista ordenada, foram calculas as métricas *Average Precision* e P@10 para cada consulta informacional e a métrica *Reciprocal Rank* para cada consulta navegacional. A média das métricas para todas as consultas de cada tipo (informacionais aleatórias, informacionais populares e navegacionais) foi o valor (nesse caso MAP, média do escore P@10 e MRR) levado em consideração para efetuar os comparativos entre as métricas.

4.2 Experimentos com a coleção WBR99

Nesta seção são apresentados os resultados obtidos através dos experimentos na WBR99. Os experimentos tiveram o objetivo de medir a eficiência dos escores propostos frente ao escore *PageRank*, para consulta informacionais.

Sobre a WBR99 foram executadas 50 consultas informacionais para medir em termos de MAP e P@10, o desempenho dos escores propostos. O gráfico da Figura 4.1 ilustra o desempenho dos escores propostos em função do limiar de detecção de versões, utilizando a métrica P@10 na WBR99.

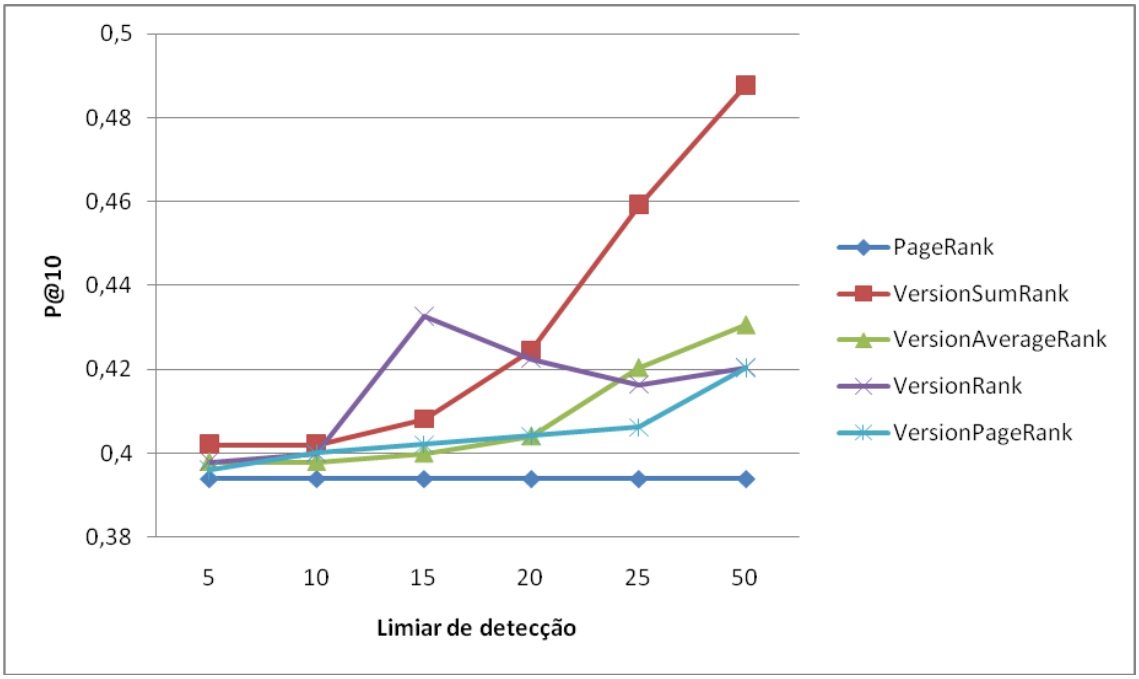

Figura 4.1: Desempenho dos escores propostos em função do limiar de detecção de versões, utilizando a métrica P@10 na WBR99.

Observa-se que o escore que tem uma semântica mais correta, pois representa documentos lógicos no *VersionGraph*, o *VersionRank*, obteve um melhor desempenho quando o limiar de detecção foi fixado em 15. Esse desempenho deve-se, provavelmente, porque nessa coleção, com o limiar de detecção fixado em 15, as versões foram agrupadas de forma mais satisfatória, montando um *VersionGraph* que representa melhor os documentos lógicos da *Web*. Já os escores que notadamente favorecem somente as páginas que tenham versões, *VersionSumRank* e *VersionPageRank*, obtiveram seus melhores resultados conforme o afrouxamento do limiar de detecção de versões, o que é uma característica esperada, visto que, com a flexibilização do limiar de detecção de versões, mais páginas são detectadas como versões, dentre elas as páginas que têm conteúdo semelhante e são relevantes.

O *VersionAverageRank* também acompanhou os limiares que favorecem as páginas que têm versões, e melhorou seu desempenho conforme a flexibilização do limiar de detecção de versões.

A Tabela 4.3 apresenta os valores de MAP e P@10 para consultas informacionais na WBR99, utilizando como limiar de detecção 15.

Tabela 4.3: MAP e P@10 para consultas informacionais na WBR99, utilizando k=15.

Escore Atribuído	*MAP*	*P@10*
PageRank	0,5550	0,3939
VersionSumRank	0,5547	0,4082
VersionAverageRank	0,5570	0,4000
VersionRank	0,5574	**0,4327**
VersionPageRank	0,5551	0,4020

Para consultas informacionais sobre a WBR99 e com o limiar *k*=15, o escore *VersionRank* foi o que apresentou melhor resultado em termos de P@10 obtendo um ganho de 9,84% sobre o *PageRank*. Para a métrica MAP, os resultados dos escores propostos foram bem similares ao resultado obtido pelo *PageRank* e não apresentaram ganho significante estatisticamente, medido através do *teste-t*

Já quando ajustamos o limiar de detecção para 25, o escore que tem uma melhor desempenho é o *VersionSumRank*, com um ganho de 16,58%, seguido do escore *VersionAverageRank* com um ganho de 6,74%, em termos de P@10. A Tabela 4.4 apresenta os resultados para MAP e P@10, quando é utilizado o limiar de detecção 25. Em termos da métrica MAP, a variação dos valores dos escores propostos é bem pequena, não apresentando significância estatística.

Tabela 4.4: MAP e P@10 para consultas informacionais na WBR99, utilizando *k*=25.

Escore Atribuído	*MAP*	*P@10*
PageRank	0,5550	0,3939
VersionSumRank	0,5703	**0,4592**
VersionAverageRank	0,5641	**0,4204**
VersionRank	0,5525	0,4163
VersionPageRank	0,5535	0,4061

4.3 Experimentos com a coleção WBR03

Os resultados apresentados nessa seção tiveram origem em experimentos realizados sobre a coleção WBR03. Nesta seção, além de comparar os escores propostos com o escore *PageRank*, os mesmos são comparados com os escores propostos por Berlt et al. (2007).

A eficiência dos escores propostos perante o *PageRank* e aos escores de Berlt et al. é medida para consultas navegacionais, assim como para consultas informacionais aleatórias e consulta informacionais populares.

4.3.1 Consultas Navegacionais

Sobre a WBR03, a primeira métrica analisada foi a MRR para consultas navegacionais. Para obter a MRR para cada escore proposto, foram utilizadas as 74 consultas disponíveis e os seus resultados, os quais tiveram a relevância previamente definida (seção 4.1.2). O gráfico da figura 4.2 apresenta os valores de MRR obtidos pelos escores propostos, em função do limiar de detecção de versões.

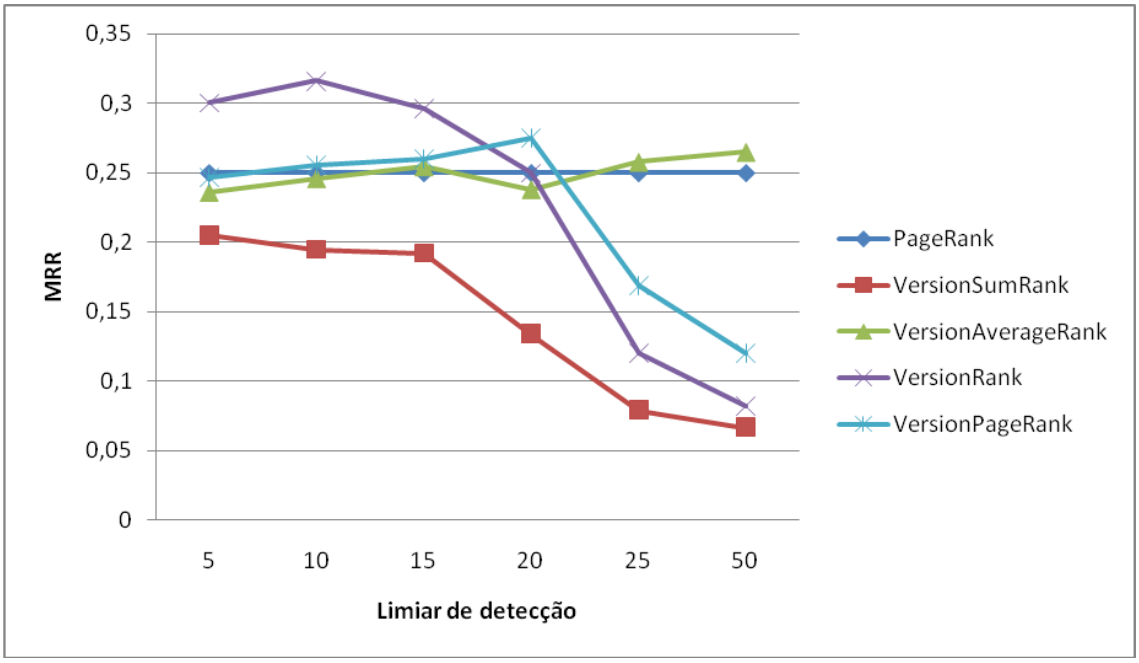

Figura 4.2: Desempenho dos escores propostos em função do limiar de detecção de versões utilizando a métrica MRR.

O escore baseado no *VersionGraph*, o *VersionRank*, apresentou uma melhora em termos de MRR para consultas navegacionais quando foram utilizados os limiares de detecção 5, 10 e 15. O ganho mais significativo em termos de MRR foi obtido quando se utilizou o escore *VersionRank* com um limiar de detecção k=10. Com o limiar de detecção de versões fixado em 5, 10 e 15 o *VersionGraph* consegue representar de forma satisfatória a estrutura de *links* entre os documentos lógicos. Para valores de k acima de 20, o escore *VersionRank* se mostrou pior que o *PageRank*, possivelmente porque o grafo de documentos *VersionGraph*, com esses limiares altos para detecção, representou páginas que não são versões de um único documento como um único vértice. Já os escores que atribuem um peso maior as páginas que possuem versões – *VersionSumRank* e *VersionPageRank* – não obtiveram resultados satisfatórios e seus resultados pioraram, em sua maioria, a medida que o limiar de detecção foi flexibilizado. Isso se deve ao fato desses dois escores tornarem maiores os valores de reputação de páginas que tem muitas versões, podendo vir a favorecer *spams* ou páginas geradas automaticamente, páginas essas que não consideradas relevantes para as consultas. O *VersionAverageRank* apresentou comportamento bem similar ao *PageRank* para os diferentes limiares apresentados.

A Tabela 4.5 apresenta a média dos valores obtidos para a métrica MRR quando são atribuídos diferentes escores para as páginas retornadas das 74 consultas e seus relativos ganhos ou perdas em relação ao *PageRank*, utilizando o limiar de detecção k=10. Para consultas navegacionais, observa-se que o melhor desempenho é obtido quando os resultados são ordenados via escore *VersionRank*, obtendo um ganho de 26,55% em relação ao *PageRank* em termos de MRR. O escore híbrido *VersionPageRank* obtém um ganho de 2,27% em relação ao *PageRank*.

Tabela 4.5: MRR para os escores atribuídos, com limiar de detecção *k*=10.

Escore Atribuído	*MRR*	*Ganho/perda em relação ao PageRank*
PageRank	0,2498	0,00%
VersionSumRank	0,1945	-22,13%
VersionAverageRank	0,2457	-1,64%
VersionRank	**0,3161**	**26,55%**
VersionPageRank	0,2554	2,27%

Os resultados superiores do escore *VersionRank* em relação ao *PageRank*, em termos de MRR, são provavelmente atribuídos a eliminação de *links* ruidosos que acontece quando representamos os documento e suas versões como um único vértice no *VersionGraph*. O *VersionGraph* por ser um grafo que representa melhor a semântica das relações entre páginas *Web*, acaba por favorecer os algoritmos de análise de *links* na atribuição de reputação.

A utilização de hipergrafos acaba por trazer benefícios às consultas navegacionais para um variado grupo de particionamentos como apresenta o gráfico da Figura 4.3, no qual são comparados os escores desse trabalho com os escores baseados em hipergrafos propostos por Berlt et al. Os percentuais apresentados no eixo Y representam o ganho dos escores perante o *PageRank*. O gráfico da Figura 4.3 apresenta um bom desempenho do *VersionRank* perante o *PageRank* para consultas navegacionais, porém os melhores resultados para a métrica MRR são obtidos pelo particionamento do hipergrafo via domínio (HiPrDom).

Convém ressaltar que os resultados superiores obtidos pelos escores propostos por Berlt et al. (2007) só são observados quando são utilizadas consultas navegacionais. Para consultas informacionais, como será apresentado a seguir, os escores propostos por este trabalho apresentam-se como melhores alternativas. Isso se deve ao fato que os escores propostos nesse trabalho se baseiam no conteúdo textual das páginas para a definição da reputação (beneficiando consultas informacionais), enquanto o trabalho de Berlt et al. define novos escores baseados em URLs (beneficiando consultas navegacionais).

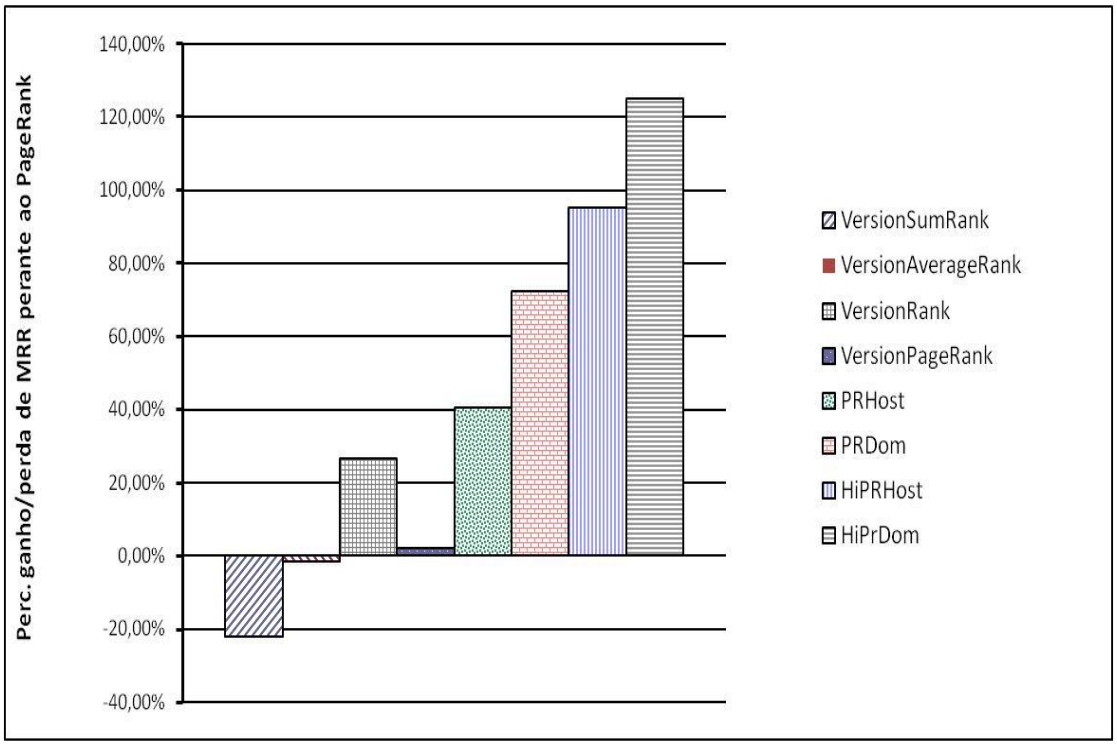

Figura 4.3: Percentual de ganho/perda para consultas navegacionais em termos de MRR perante o PageRank, com *k*=10.

4.3.2 Consultas Informacionais Aleatórias.

Sobre a coleção WBR03 ainda foram realizados experimentos envolvendo consultas informacionais e as métricas utilizadas para medição de desempenho foram a MAP e a P@10. Os primeiros dados dizem respeito à medição das métricas utilizando 62 consultas informacionais aleatórias.

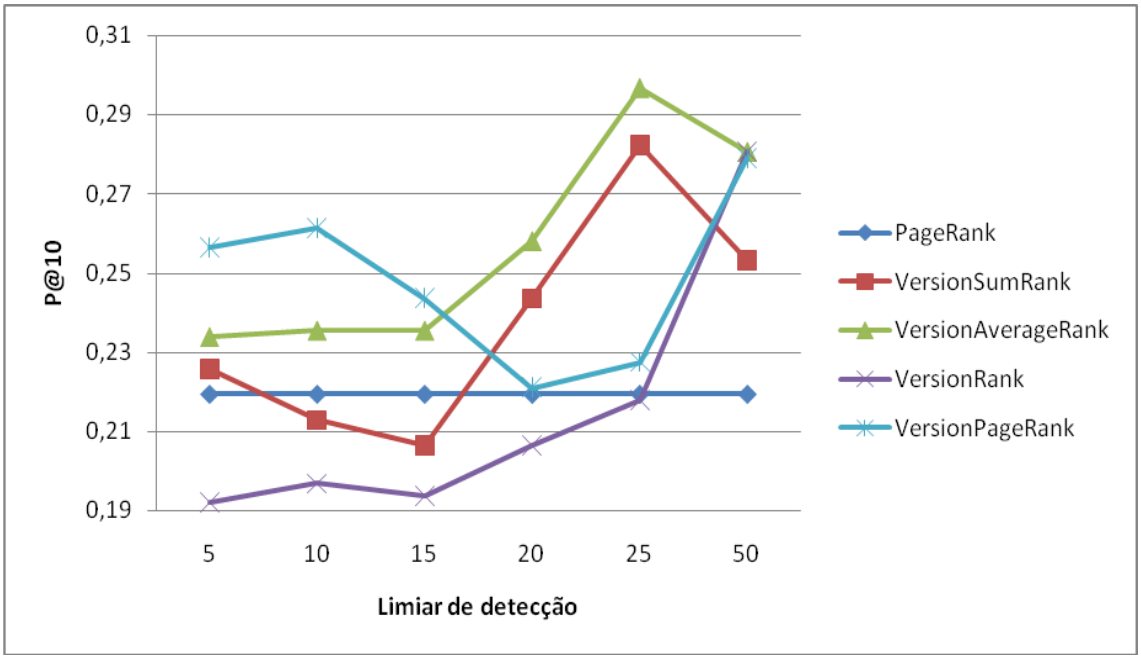

Figura 4.4: Desempenho dos escores propostos em função do limiar de detecção de versões utilizando a métrica P@10, para consultas informacionais aleatórias.

O gráfico da Figura 4.4 apresenta o desempenho dos escores propostos em função do limiar de detecção de versões utilizando a métrica P@10. A primeira constatação é que o escore *VersionRank*, que havia se comportado de uma forma muito satisfatória para consultas informacionais na coleção WBR99, não conseguiu repetir seu bom desempenho na coleção WBR03. A natureza da coleção WBR03 pode ter contribuído para o baixo desempenho do escore *VersionRank*, visto que a mesma contém muitas páginas *spams* e de conteúdo gerado de forma automática. Essas páginas não são relevantes para as consultas, mas podem ser agrupadas como um único vértice no *VersionGraph*, e por conseqüência, obter um alto valor de *VersionRank*. O escore *VersionRank* só consegue um desempenho superior ao *PageRank* quando o limiar de detecção é definido com 50, o que se considera um afrouxamento muito grande no limiar que pode acabar considerando versões, páginas que não são versões. Entretanto, deve-se observar a curva de desempenho obtida pelo escore VersionPageRank, que obteve resultados melhores que o PageRank para todos os limiares de detecção.

Ainda é possível observar no gráfico da Figura 4.4 o bom desempenho do escore *VersionAverageRank*, sempre acima do desempenho do *PageRank*, e especialmente quando o limiar de detecção escolhido é 25. O *VersionAverageRank* apresenta um crescimento no seu desempenho a medida que o limiar de detecção é flexibilizado, com exceção do limiar de detecção 50. No experimento com consultas informacionais sobre a WBR99 (seção 4.2) e no experimento que será apresentado na próxima seção, com consultas informacionais populares, este comportamento do escore *VersionAverageRank* também pode ser constatado.

A Tabela 4.6 apresenta a MAP e P@10 obtidas pelas consultas informacionais aleatórias sobre a WBR03, com um limiar de detecção k=10.

Tabela 4.6: MAP e P@10 para consultas informacionais aleatórias, com k=10.

Escore Atribuído	*MAP*	*P@10*
PageRank	0,3348	0,2194
VersionSumRank	0,3328	0,2129
VersionAverageRank	0,3379	0,2355
VersionRank	0,3234	0,1968
VersionPageRank	0,3425	**0,2613**

Com a definição do limiar de detecção de versões fixado em 10, os valores da MAP apresentaram uma variação muito pequena (não apresentando significância estatística, segundo o *teste-t*) e para os valores de P@10 o escore *VersionPageRank* apresenta um ganho de 19,12% em comparação com o escore PageRank.

Na Tabela 4.7, são apresentados também dados para consultas informacionais aleatórias sobre a WBR03, porém com um limiar de detecção definido como 25.

Tabela 4.7: MAP e P@10 para consultas informacionais aleatórias, com k=25.

Escore Atribuído	*MAP*	*P@10*

PageRank	0,3348	0,2194
VersionSumRank	0,3637	0,2823
VersionAverageRank	**0,3648**	**0,2968**
VersionRank	0,3467	0,2177
VersionPageRank	0,3498	0,2274

Com o limar de detecção fixado em 25, o escore *VersionAverageRank* apresentou um ganho de 8,96% em termos de MAP. Através do *teste-t* sobre as precisões médias, foi comprovado que o escore *VersionAverageRank* é significantemente superior ao *PageRank* para consultas informacionais aleatórias sobre a base WBR03. Já em termos de P@10, o *VersionAverageRank* apresentou um ganho de 35,29% em comparação com o escore *PageRank*, ainda utilizando o limiar de detecção 25

O gráfico da Figura 4.5 apresenta o ganho/perda percentual dos escores propostos nesse trabalho em comparação com os escores propostos por Berlt et al. (2007), tomando como base de comparação o escore *PageRank*. Importante notar que os ganhos/perdas dos escores propostos por Berlt et al, são baseados nos resultados obtidos sem a combinação de nenhum outro método de ordenação e são resultados baseados em consultas informacionais populares.

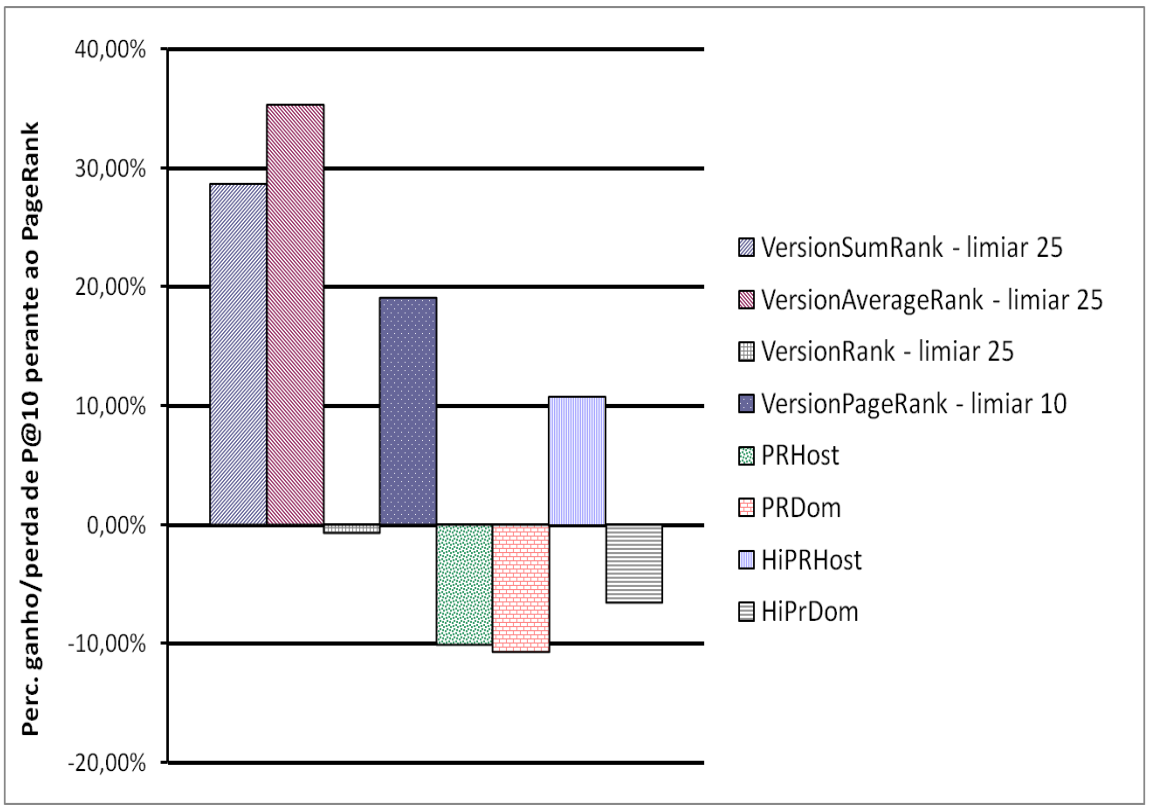

Figura 4.5: Percentual de ganho/perda em termos de P@10 perante o PageRank para consultas informacionais na coleção WBR03.

O fato dos escores *VersionAverageRank* e o *VersionPageRank* apresentarem um desempenho superior ao *PageRank* e aos escores propostos por Berlt et al. para consultas informacionais, na métrica P@10, deve-se a natureza dessas consultas.

Consulta informacionais visam obter informações sobre um tópico específico, sendo que várias páginas podem ser consideradas relevantes. Essa característica acaba por trazer benefícios aos escores *VersionAverageRank* e *VersionPageRank*, visto que as páginas que são versões de um mesmo documento acabam sendo favorecidas quando consideramos a métrica P@10 para a medição do desempenho. Partindo-se do pressuposto que se uma das versões é considerada relevante, as outras também serão (pois tem conteúdo similar), os ganhos de P@10 podem ser significativos para algumas consultas.

4.3.3 Consultas Informacionais Populares

Para as 62 consultas informacionais populares da WBR03, o resultado mais positivo foi o do escore *VersionAverageRank* com limiares de detecção de versões fixados em 25 e 50. A melhora de desempenho do escore *VersionAverageRank* conforme a flexibilização do limiar de detecção foi observada para todas as consultas informacionais avaliadas em termos de P@10 – nas duas coleções: WBR99 e WBR03.

O gráfico da Figura 4.6 apresenta o desempenho dos escores propostos em função dos limiares de detecção, para a métrica P@10. Além do melhor desempenho do escore *VersionAverageRank*, é possível observar o fraco desempenho do escore *VersionRank* e o desempenho bem próximo ao *PageRank* dos outros escores propostos para consultas informacionais populares na WBR03.

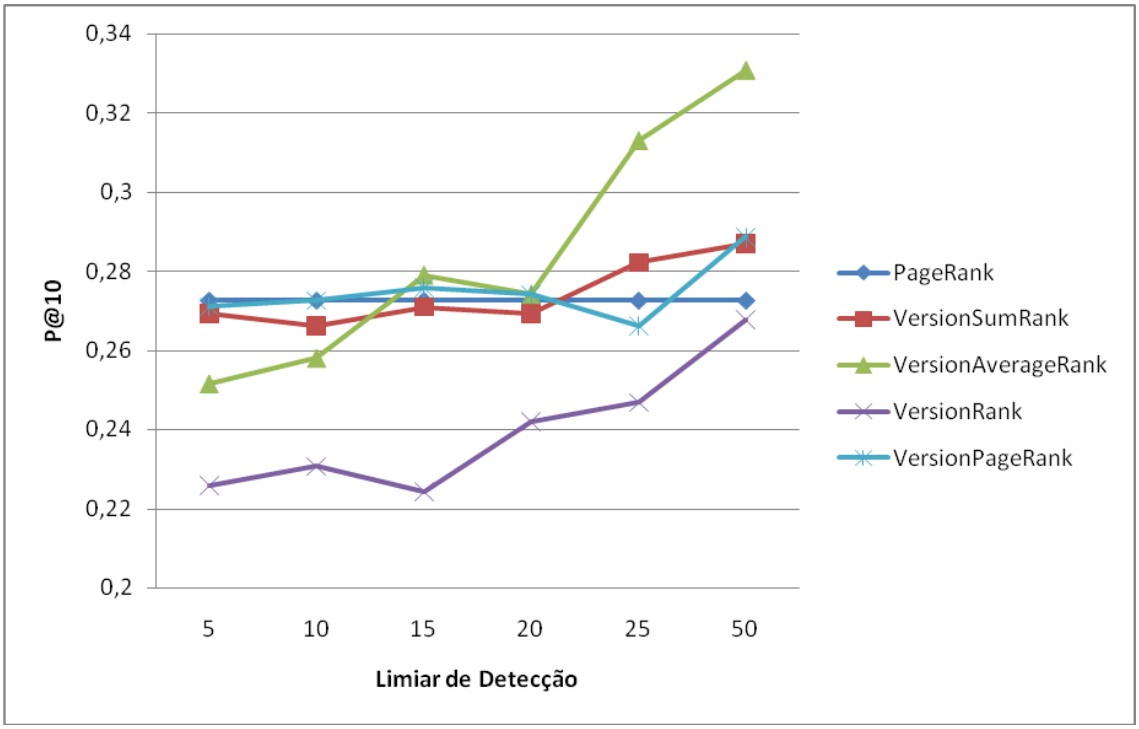

Figura 4.6: Desempenho dos escores propostos em função do limiar de detecção de versões utilizando a métrica P@10 para consultas informacionais populares.

A Tabela 4.8 apresenta os dados para consultas informacionais populares sobre a WBR03 com um limiar de detecção definido como 25. Os valores para a métrica MAP apresentaram valores bem similares para todos os escores propostos e através do *teste-t* não foi detectada diferença estaticamente significativa. Já para a métrica P@10, o escore *VersionAverageRank* conseguiu um desempenho 14,79% superior ao *PageRank*.

Tabela 4.8: MAP e P@10 para consultas informacionais populares, com *k*=25.

Escore Atribuído	*MAP*	*P@10*
PageRank	0,3455	0,2726
VersionSumRank	0,3418	0,2823
VersionAverageRank	0,3564	**0,3129**
VersionRank	0,3417	0,2468
VersionPageRank	0,3502	0,2661

O desempenho similar entre os escores *PageRank* e *VersionPageRank* e o baixo desempenho do escore *VersionRank* para consultas informacionais populares na WBR03 pode ser explicado pela natureza de uma boa parte das consultas populares avaliadas. As consultas populares por serem de muita abrangência tais como: borboleta, horóscopo, baladas, santos, classificados, acabam por trazer um efeito colateral do uso *VersionRank*, e por consequência para o *VersionPageRank*, porém em menor escala, à tona. Páginas que são *spams*, ou simplesmente lixos replicados, que contém os termos da consulta, por terem inúmeras versões e/ou réplicas acabam sendo favorecidas pelo *VersionRank*, e como não são relevantes trazem prejuízos ao escore para esses tipos de consulta.

4.4 Análise Geral dos Experimentos

Para consultas navegacionais, o escore *VersionRank* obteve melhores resultados que o *PageRank*. O *VersionRank*, para consultas navegacionais, obteve um ganho de 26,55% em termos de MRR perante o *PageRank*, com o limiar de detecção *k*=10. Esse ganho reflete o melhor significado semântico do *VersionGraph* em relação ao *WebGraph*, e também a eliminação de *links* ruidosos efetuada ao representar somente documentos lógicos com o *VersionGraph*. Esse resultado pode ainda ser combinado com outras propostas mais eficazes pra consultas navegacionais, como o trabalho de Berlt et al. (2007).

Já para consultas informacionais, o *VersionRank*, na coleção WBR99, obteve um ganho de 9,84% em termos de P@10 perante o *PageRank*, com o limiar de detecção de versões fixado em 15.

Sobre a coleção WBR03, o escore *VersionPageRank*, quando fixado o limiar de detecção *k*=10, obteve um ganho de 19,12% em termos de P@10, para consultas informacionais aleatórias. Para consultas informacionais populares, o *VersionPageRank* não obteve ganhos e nem perdas significativas, em termos de P@10, perante o PageRank.

Quando o limiar de detecção foi fixado em 25, o escore *VersionAverageRank* obteve os melhores resultados para consultas informacionais na WBR03 e também resultados satisfatórios na WBR99. Em termos de P@10, em comparação com o *PageRank*, o *VersionAverageRank* teve um ganho de 6,74% para consultas informacionais na WBR99. Na WBR03, o *VersionAverageRank* obteve um ganho de 35,29% para consulta informacionais aleatórias e de 14,79% para consultas informacionais populares, em termos de P@10. Sobre consultas informacionais aleatórias da WBR03,

ainda com o limiar de detecção fixado em 25, o *VersionAverageRank* mostrou-se significantemente melhor que o *PageRank*, constatação feita ao realizar o *teste-t* sobre as precisões médias.

De uma maneira geral, é possível afirmar que para consultas navegacionais o escore *VersionRank*, com o limiar de detecção fixado em 10, se constituí uma alternativa muito interessante ao *PageRank* por ter obtido ganho de 26,55% em termos de MRR sobre o mesmo. Já para consultas informacionais, é possível afirmar que o escore *VersionAverageRank* apresentou um desempenho muito bom e que cresceu com a flexibilização do limiar de detecção de versões. O *VersionAverageRank* apresentou resultados melhores, em termos de P@10, que o *PageRank* e que os escores propostos por Berlt et al. (2007), para consultas informacionais sobre a WBR03, avaliando as mesmas consultas que foram utilizadas por Berlt et al., e sem utilizar outros critérios de ordenação.

5 CONCLUSÕES E TRABALHOS FUTUROS

Esta dissertação apresentou uma abordagem para utilizar a detecção de versões para aprimorar a atribuição de reputação as páginas *Web*. Foram apresentados os trabalhos relacionados com o tema dessa dissertação, destacando as diferenças dos mesmos com a proposta aqui apresentada. Ao detalhar a abordagem proposta, foi escolhido e parametrizado o algoritmo de similaridade utilizado na detecção de versões. Utilizando o índice de versões produzido pelo processo de detecção de versões, foram propostos quatro escores de reputação, que visam atribuir uma reputação mais homogênea a todas as versões de um documento *Web*. Foram realizados experimentos para medir o impacto da ordenação dos resultados dos motores de busca utilizando os escores propostos.

A principal contribuição desse trabalho é propor escores baseados na detecção de versões de páginas *Web* como alternativa aos algoritmos de análise de *links* existentes.

Nos experimentos realizados, o escore *VersionAverageRank* apresentou melhores resultados em consultas informacionais perante o *PageRank* (nas coleções WBR99 e WBR03) e aos escores propostos por Berlt et al. (2007) (na coleção WBR03), nas coleções de páginas *Web* em que foram executados os experimentos. Já o escore *VersionRank* apresentou um melhor desempenho que o *PageRank* para consultas informacionais sobre a WBR99. Para consultas navegacionais, o escore *VersionRank* apresentou melhores resultados que o *PageRank* na base WBR03.

O principal objetivo dos novos escores propostos é considerar a reputação de todas as versões de uma página *Web* no momento de montar um *ranking* de resultados ao usuário de um motor de busca. Considerando todas as versões, facilita-se o acesso a novas versões de páginas relevantes e aumenta a reputação de todas as versões, mantendo todas as versões de um mesmo documento lógico em posições próximas no *ranking*. Como ponto negativo, os escores propostos acabam por atribuir um escore de reputação mais elevado para páginas que utilizam o plágio para compor seus conteúdos. Apesar desse fato, alguns dos escores definidos nesse trabalho obtiveram resultados percentualmente superiores a outros definidos na literatura nos experimentos realizados, o que vem justificar o seu uso.

Os escores propostos nesse trabalho podem ser utilizados, em conjunto com outras técnicas, para a montagem do *ranking* de resultados apresentados pelos motores de busca. A técnica de detecção de versões aqui apresentada pode ser usada para melhorar a apresentação dos resultados em um motor de busca.

No seguimento do trabalho, os escores de reputação desse trabalho podem ser combinados com outros escores da literatura (PAGE, 1999; BERLT et al., 2007) para a definição de novos escores que obtenham resultados melhores em consultas navegacionais e informacionais.

Como alternativa para trabalhos futuros, ao invés de propor agrupamentos baseados em versões para montar o *VersionGraph*, podem ser propostos a criação de clusters de documentos baseados na similaridade textual dos seus conteúdos, não sendo necessariamente versões, sendo uma oportunidade de criação de novos *WebGraphs* baseados em similaridade textual e, por consequência, novos escores de reputação.

REFERÊNCIAS

BAEZA-YATES, R.; DAVIS, E. Web page ranking using link attributes. **Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters**, New York, p. 328 – 329, 2004.

BAEZA-YATES, R.; BOLDI, P.; CASTILLO, C. Generalizing PageRank: damping functions for link-based ranking algorithms. **Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval**, Seattle, p. 308 – 315, 2006.

BAEZA-YATES, R.; PEREIRA, Á.; ZIVIANI, N. Genealogical trees on the web: a search engine user perspective. **Proceeding of the 17th international conference on World Wide Web**, Beijing, China, p. 367-376, 2008.

BERLT, K.; MOURA, E. S.; CARVALHO, A. L. C.; CRISTO, M.; ZIVIANI N.; COUTO, T. A hypergraph model for computing page reputation on web collections. **Anais do Simpósio Brasileiro de Banco de Dados**, João Pessoa, p. 35-49, 2007.

BRIN, S.; PAGE, L. The anatomy of a large-scale hypertextual Web search engine. **Computer Networks and ISDN Systems**. Amsterdam, p. 107-117, abr. 1998

BRODER, A. Z.; GLASSMAN, S. C.; MANASSE, M. S.; ZWEIG, G. Syntactic clustering of the Web. **Selected Papers From the Sixth international Conference on World Wide Web**, Santa Clara, California, EUA, p. 1157-1166, 1997

BRODER, A. A taxonomy of web search. **SIGIR Forum**, New York, v.36, n.2, p, 3-10, 2002.

CALADO, P.; RIBEIRO-NETO, B.; ZIVIANI, N.; MOURA, E.; SILVA, I. Local versus global link information in the Web. **ACM Transactions on Information Systems (TOIS)**, v.21, i.1 p 42-63, jan. 2003.

CARVALHO, A. L. C,; CHIRITA, P.; MOURA, E. S.; CALADO, P.; NEJDL, W. Site level noise removal for search engines. **Proceedings of the 15th international Conference on World Wide Web**, Edinburgh, Scotland, p.73-82, 2006.

CHARIKAR, M. S. Similarity estimation techniques from rounding algorithms. **Proceedings of the Thiry-Fourth Annual ACM Symposium on Theory of computing**, Montreal, Canada, p. 380-388, 2002.

CHO, J.; SHIVAKUMAR, N.; GARCIA-MOLINA, H. Finding replicated Web collections. **ACM SIGMOD Record**, New York, v. 29, p. 355-366, 2000.

DING, C.; HE, X.; HUSBANDS, P.; ZHA, H.; SIMON, H. D. PageRank, HITS and a unified framework for link analysis. **Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval**, Tampere, Finland, p. 353-354, 2002

HENZINGER, M. Finding near-duplicate web pages: a large-scale evaluation of algorithms. **Proceedings of the 29th Annual international ACM SIGIR Conference on Research and development in information retrieval**, Seattle, p. 284-291, 2006

KLEINBERG, J. M. Authoritative sources in a hyperlinked environment. **Journal of the ACM (JACM)**, New York, v.46, p. 604 – 632, 1999.

LIU, Y.; GAO, B.; LIU, T.; ZHANG, Y.; MA, Z.; HE, S.; LI, H. BrowseRank: letting web users vote for page importance. **Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval**. Singapore, p. 451-458, 2008.

MANKU, G. S.; JAIN, A.; DAS SARMA, A. Detecting near-duplicates for web crawling. **Proceedings of the 16th international Conference on World Wide Web**, Banff, Canada, p. 141-150, 2007.

PAGE, L.; BRIN, S.; MOTWANI, R.; WINOGRAD T. **The PageRank citation ranking: Bringing order to the Web**. Technical Report n. 1999-66, Stanford University, Stanford, 1999.

SILVA, R.; STASIU, R.; ORENGO, V. M.; HEUSER, C.A. Measuring quality of similarity functions in approximate data matching. **Journal of Informetrics,** v.1, i.1, p. 35-46, jan. 2007.