UNIVERSIDADE FEDERAL DE ALFENAS
INSTITUTO DE CIÊNCIAS EXATAS
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Daniel Fernandes Rey

Análise do impacto das atualizações de fórmulas matemáticas em bibliotecas matemáticas digitais para a SearchOnMath.

Alfenas, 07 de Julho de 2015.

UNIVERSIDADE FEDERAL DE ALFENAS
INSTITUTO DE CIÊNCIAS EXATAS
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Análise do impacto das atualizações de fórmulas matemáticas em bibliotecas matemáticas digitais para a SearchOnMath.

Daniel Fernandes Rey

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação da Universidade Federal de Alfenas como requisito parcial para obtenção do Título de Bacharel em Ciência da Computação.

Orientador: Prof. Flavio Barbieri Gonzaga

Alfenas, 07 de Julho de 2015.

Daniel Fernandes Rey

Análise do impacto das atualizações de fórmulas matemáticas em bibliotecas matemáticas digitais para a SearchOnMath.

A Banca examinadora abaixo-assinada aprova a monografia apresentada como parte dos requisitos para obtenção do título de Bacharel em Ciência da Computação pela Universidade Federal de Alfenas.

Prof. Evandro Monteiro

Universidade Federal de Alfenas

Profa. Mariane Moreira

Universidade Federal de Alfenas

Prof. Flavio Barbieri Gonzaga (Orientador)

Universidade Federal de Alfenas

Alfenas, 07 de Julho de 2015.

Dedico este trabalho a meus pais Paulo e Luiza, ao meu irmão Thiago e à todos os grandes amigos que fiz em Alfenas..

AGRADECIMENTO

Agradeço primeiramente a Deus e aos meus amados pais, Paulo Marcelo Rey e Luiza Maria Fernandes Rey e irmão Thiago Fernandes Rey, por me apoiarem muito nestes anos todos em Alfenas, além de serem excelentes exemplos de como pais e irmãos devem ser.

Às grandes amizades conquistadas em Alfenas, especialmente Alberto Carilo, Gabriela Moreira, Reuel Ramos Ribeiro, Victor Carvalho e Wilson Sasaki Junior, o meu primeiro grande amigo na cidade! Agradeço a cada um de vocês por toda a ajuda com trabalhos e provas nestes anos todos. Um viva a todos os finais de semanas e madrugadas na universidade que, com certeza, jamais serão esquecidos. Talysson, Eugênio, Gabriel Amaral, Adenir, Vinícius Ferreira, Caio, Délio, Lucas, Tabarin, Leandro e Karine e todos os que fizeram estes anos inesquecíveis, muito obrigado. Aos grandes amigos que já saíram da universidade rumo ao sucesso, Jaffer Veronezi e João Antonio da Silva, que compartilharam muitos momentos comigo. As amigas Regiane Cristina e Isabella Cavalcante, pela companhia no tempo em que moraram em Alfenas. A todos os meus amigos, nunca mais comer lanche de madrugada será a mesma coisa. E um agradecimento especial à Gisele Novaes Franco, pela mais do que agradável surpresa e extremo apoio.

A minha família de Alfenas, que desde o dia em que cheguei me acolheu com muito amor, Leni Viçoso, Pâmela Viçoso Santos e Poenna Viçoso Veiga. Agradeço, em especial minha amiga Poenna, que ao longo desses anos todos se tornou mais do que uma amiga, uma pessoa extremamente especial e que, com certeza, sentirei muitas saudades.

A todos os professores do curso de Bacharelado em Ciência da Computação e ao grande professor do curso de Licenciatura em Matemática, Anderson Oliveira, pela dedicação e todo conhecimento passado. Ao coordenador Humberto Brandão, que não só resolveu o meu problema de falta de calculadora, mas como resolveu também a falta de vaga em matérias.

Em especial, agradeço ao meu orientador Flavio Barbieri Gonzaga, por ter me aceitado como integrante no LaReS para uma Iniciação Científica e continuado a trabalhar comigo até hoje. Agradeço pela dedicação e paciência em ensinar, confiança e amizade nesses anos todos. Ao excelente professor e orientador, muito obrigado por tudo.

"If you can't explain it simply, you don't understand it well enough."

Albert Einsten

RESUMO

Bibliotecas digitais como DLMF (*Digital Library of Mathematical Functions*) e MathWorld representam portais com uma grande base de dados de conteúdo matemático e que são disponibilizados livremente. Tais portais são dinâmicos, ou seja, ao longo de sua história recebem contribuições por parte de usuários e administradores. Assim, páginas (e consequentemente as fórmulas matemáticas contidas nas mesmas) podem ser excluídas, modificadas ou acrescentadas. Considerando o fato de que uma ferramenta de busca possui o conteúdo das páginas armazenado localmente (por exemplo, o Google possui uma cópia de todas as páginas que ele retorna na busca), saber com que frequência uma determinada página é modificada é um problema clássico na área de Busca e Recuperação de Informação. Isso porque se uma determinada página é modificada, e a ferramenta de busca não atualiza a sua cópia local, um usuário que busque por uma informação nova não a encontrará na ferramenta de busca, dado que o texto que a mesma possui ainda está desatualizado. O presente trabalho propõe uma análise inicial desse problema, da taxa de mudança de conteúdo, mas aplicado em fórmulas matemáticas. A motivação para o estudo é em função da ferramenta de busca por fórmulas matemáticas SearchOnMath (em constante desenvolvimento no LaReS). Para a ferramenta, mesmo que o texto de uma página sofra alguma modificação, como o foco da busca está na fórmula em si, se uma fórmula não tiver sido modificada, a ferramenta ainda será capaz de encontrá-la. Espera-se que a taxa de mudança de fórmulas não seja tão elevada quanto a taxa de mudança textual. Com isso, deseja-se testar o impacto de não atualização de uma base de dados composta por fórmulas, e a consequência dessa desatualização na qualidade da busca.

Palavras-Chave: SearchOnMath, Atualização de fórmulas, WebCrawler, DLMF, MathWorld

ABSTRACT

Digital libraries, such as DLMF (Digital Library of Mathematical Functions) and MathWorld, represent portals with large databases of mathematical content that are freely available. Such web portals are dynamic, wich means throughout their history they receive contributions from users and administrators. Therefore, pages (and consequently the mathematical formulas they contain) might be excluded, modified, or augmented. Considering the fact that search engines store the content of web pages locally (Google, for example, stores a copy of all the pages that are returned in a search), knowing the frequency with which a determined page is modified, is a classical problem in the Data Search and Recover Area, for if a certain page is modified, and the search engine does not update its stored copy, users will not be able to find the modified page using said engine, given the fact that its stored copy is not up to date. This paper proposes an initial analysis of the rate of content change applied to mathematical formulas. The motivation for this study is the mathematical formula search engine SearchOnMath (in constant development at LaReS). For this engine, however, the text changing of web pages contais. Hence, inasmuch as the formulas are not changed, SearchOnMath will still be able to find the page. The rate of change of mathematical formulas if expected not to be as high as the text changing rate. That being said, the aim is to analyse the impact of not updating a database of mathematical formulas in the quality of the results returned by a mathematical search engine

Keywords: SearchOnMath, Formulas updating, WebCrawler, DLMF, MathWorld

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1
Introdução

Com o contínuo crescimento da Internet, a busca pelas informações nela presente cresceu muito e ferramentas de busca como Google, Yahoo! e Bing surgiram para facilitar o acesso a este conteúdo. Entretanto, com o passar do tempo surgiu a necessidade de buscar informações específicas, seja por empresas ou por usuários interessados em apenas um tipo de informação. Devido a este fato, ferramentas de busca específicas como Google Scholar[1] cuja função é buscar apenas por artigos científicos, Wolfram Alpha[2], que é capaz de responder a perguntas ao invés de mostrar uma lista de sites com conteúdo sobre o assunto buscado, Tangent[3], uma ferramenta de busca por conteúdo matemático no portal Wikipedia[4] e a SearchOnMath, uma ferramenta de busca por conteúdo matemático em diversos portais.

Existem várias bibliotecas digitais de conteúdo matemático, como DLMF[1], MathWorld[2], Planetmath[5] e MathOverflow[6]. Para este trabalho, foram escolhidas DLMF e MathWorld como foco de estudo, pois, além de representarem uma parte da base de dados da ferramenta SearchOnMath[3], a primeira foi criada e é mantida pelo NIST (National Institute of Standards and Tecnology) e a segunda por ser um dos maiores portais de conteúdo matemático da internet.

Cada ferramenta de busca utiliza uma ferramenta chamada WebCrawler para baixar o conteúdo de sites e armazená-los em seus bancos de dados. Entretanto, os sites sofrem atualizações, alguns mais constantes e outros menos. Cada atualização nova gera uma inconsistência no que a página exibe para o usuário e no que a ferramenta de busca pode exibir ao usuário. Devido a este fato, as ferramentas de busca tem que atualizar constantemente seus bancos de dados.

xxiiixxiii———————————

[1] https://scholar.google.com.br/

[2] http://www.wolframalpha.com/

[3] http://saskatoon.cs.rit.edu/tangent/

[4] https://en.wikipedia.org/

[5] http://planetmath.org/

[6] http://mathoverflow.net/

Considerando este fato, vários trabalhos foram realizados com a ideia de se identificar a melhor taxa de atualização para as ferramentas de busca. Entretanto, ferramentas como Google, Yahoo! e Bing são textuais e de conteúdo geral, isto é, elas têm como propósito exibir qualquer tipo de informação que o usuário desejar. Por isso, o principal foco delas está nos textos das páginas.
Isso faz com que as atualizações das páginas sejam mais constantes, o que, considerando páginas de notícias e páginas que se concentram em texto, no geral, é uma abordagem que dificilmente será evitada. Porém, páginas de conteúdo matemático se concentram em fórmulas e equações, o que leva a considerar se as políticas de atualizações aplicadas em outras páginas podem e devem ser aplicadas a este tipo de conteúdo. Com base nisso, este trabalho foi desenvolvido para verificar se as atualizações ocorridas em páginas de conteúdo matemático são suficientemente significativas para justificar a atualização das bases de dados de domínios matemáticos presentes em ferramentas de busca.

1.1 Justificativa e Motivação

Ferramentas de buscas específicas possuem como objetivo a recuperação de informação referente apenas ao assunto em que ela se propõe a cobrir. Neste trabalho, o assunto coberto é a Matemática.

Portais de conteúdo geral, como UOL[7], G1[8], são atualizados constantemente e, portanto, ferramentas de busca atualizam suas bases de dados referentes a estes domínios constantemente. Entretanto, portais matemáticos não possuem esta taxa frequente de atualização, sendo que alguns demoram meses ou até mesmo mais de um ano para serem atualizados. Porém, mesmo depois que eles são atualizados, não se sabe se esta atualização justificaria que uma ferramenta de busca atualizasse sua base de dados referente a algum dos portais matemáticos. Tendo este problema como base, o trabalho foi desenvolvido a fim de se determinar se mudanças sofridas por portais matemáticos sempre justificam uma atualização na base de dados de uma ferramenta de busca, tomando como base os portais DLMF e MathWorld.

xxivxxiv

[7] http://www.uol.com.br/

[8] http://g1.globo.com/index.html

1.2 Objetivos

1.2.1 Gerais

O objetivo deste trabalho é determinar se as atualizações sofridas por portais matemáticos sempre justificam que ferramentas de buscas atualizem suas bases de dados referentes aos portais.

1.2.2 Específicos

A fim de se atingir o objetivo proposto, alguns objetivos específicos precisam ser completados.

- Obtenção das diferentes bases de dados dos dois domínios estudados;
- Configuração de uma versão específica da ferramenta de busca SearchOnMath contendo apenas as bases de dados estudadas neste trabalho;
- Configuração de um servidor para rodar uma versão específica da ferramenta de busca SearchOnMath e de um WebService para permitir buscas na ferramenta de uma forma automatizada;
- Criação de um algoritmo para busca automatizada na SearchOnMath através do WebService.

2
Revisão Bibliográfica

Este capítulo apresenta uma revisão bibliográfica sobre o tema abordado. Alguns trabalhos já foram realizados para determinar a melhor taxa de atualização para ferramentas de buscas, porém, estes trabalham concentraram-se em ferramentas de busca de conteúdo geral.

Em Cho e Garcia-Molina [4], utilizando uma base de dados com 720.000 páginas de 270 sites diferentes e com base nas mudanças ocorridas nas mesmas, um modelo matemático foi construído para representar as mudanças realizadas nas páginas.

Neste caso, o modelo foi construído através do processo de Poisson[5], pois o mesmo é geralmente utilizado para modelar uma sequência de eventos randômicos que acontecem independentemente com uma taxa fixa durante o tempo. Como por exemplo, ocorrências de acidentes fatais na estrada, a chegada de clientes a um centro de serviços, número de ligações originárias de uma determinada região, geralmente são modelas através do processo de Poisson. O trabalho consistiu em separar as páginas em grupos chamados de janelas e, para cada página baixada, foi verificado o tempo que em que ela era acessível dentro da janela e, este número, foi utilizado com o período de vida da página.

Com base nisso, foi possível construir o modelo de Poisson abaixo.

Teorema 1 *Se T é o tempo para a ocorrência do próximo evento em um processo de Poisson com taxa λ, a função de densidade de probabilidade para T é*

$$fT(t) = \begin{cases} \lambda e^{-\lambda t} \; para \; t > 0 \\ 0 \; para \; t \leq 0 \end{cases}$$

Este teorema pode ser utilizado para verificar se as mudanças de uma página seguem um processo de Poisson. Isto é, se uma mudança de uma página segue um processo de Poisson com taxa λ, o intervalo de suas mudanças deve seguir a distribuição $\lambda e^{-\lambda t}$. Para comparar esta previsão com os dados presentes, os autores assumiram que cada página p_i na internet possui uma taxa média de variação λ_i, onde λ_i pode diferenciar de página para página. Então, um intervalo médio de mudança qualquer, como, por exemplo, 10 dias, pode ser escolhido e, apenas as páginas que seguem este intervalo, são selecionadas e, um gráfico contendo a distribuição de seus intervalos de mudança é gerado. Caso as páginas sigam, de fato, um processo de Poisson, o gráfico deve exibir uma distribuição exponencial. A figura 1 exibe dois gráficos gerados com essas configurações.

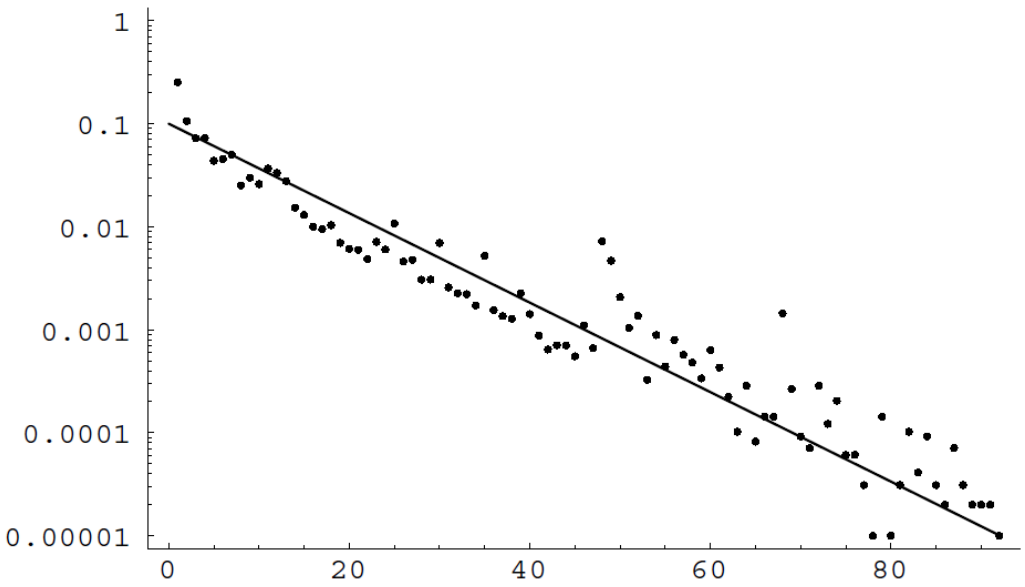

Figura 1 Representação páginas com um intervalo de mudança de 10 dias. Extraído de Cho e Garcia-Molina [1].

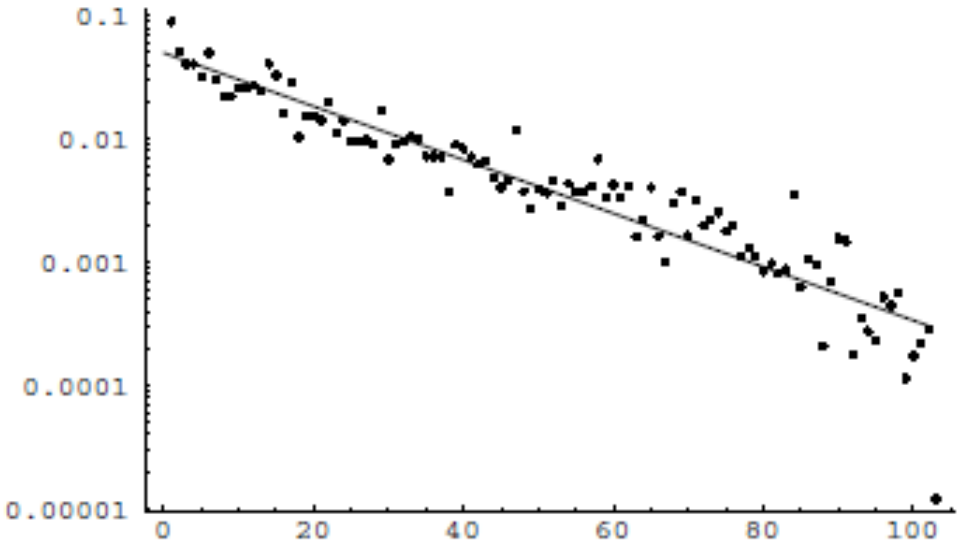

Figura 2 Representação páginas com um intervalo de mudança de 20 dias. Extraído de Cho e Garcia-Molina [4].

A Figura 1 representa o gráfico das páginas com 10 dias de intervalo entre as atualizações. Semelhantemente, a Figura 2 representa o gráfico para as páginas com 20 dias de intervalo entre as atualizações. O eixo horizontal representa o intervalo entre as sucessivas modificações nas páginas e o eixo vertical exibe a fração de modificações naquele intervalo. As linhas nos gráficos são as previsões feitas pelo processo de Poisson. Como pode ser visto, este processo é uma boa escolha para a previsão de quando as páginas vão ser atualizadas.

Cho e Garcia-Molina [5] investigaram políticas eficientes de atualização para WebCrawlers. Os autores estudaram como as cópias locais criadas pelos Crawlers podem ser atualizadas de formas mais eficiente. Para isso, foram estudados quatro pontos importantes.

- Frequência de Sincronização;
- Alocação de Recursos;
- Ordem de Sincronização;
- Horário da Sincronização.

2.1 Frequência de Sincronização

A primeira decisão que deve ser tomada é em relação a frequência de sincronização entre um domínio e a base de dados mantida pela ferramenta. Obviamente, quanto mais frequente for esta sincronização, mais atual será a base de dados. Na análise realizada pelos autores, N elementos (páginas da web) foram sincronizados com uma taxa I de unidade de tempo. Variando I, é possível ajustar a frequência de atualização da base de dados.

2.2 Alocação de Recursos

Mesmo que a quantidade de elementos que serão sincronizados por unidade de tempo já tenha sido decidida, ainda é necessário decidir com que frequência cada elemento, individualmente, será sincronizado. Para isto, foram propostas duas políticas pelos autores. São elas:

- Política de Alocação Uniforme – Onde todos os elementos são atualizados com a mesma frequência, independente das próprias frequências de cada elemento.
- Política de Alocação Não Uniforme – Onde todos os elementos são sincronizados utilizando frequências diferentes.

2.3 Ordem de Sincronização

É necessário escolher a ordem em que cada elemento será atualizado, uma vez que isso pode ter algum impacto no resultado visualizado por um usuário de uma ferramenta de busca. Foram propostas três políticas para a ordem de atualização. São elas:

- Ordem Fixa – Todos os elementos serão sempre sincronizados na mesma ordem. Portanto, neste caso, a política de alocação uniforme estaria sendo aplicada.
- Ordem Aleatória – Todos os elementos serão sempre sincronizados, porém, a ordem de sincronização pode variar a cada iteração da sincronização. Uma permutação aleatória dos elementos em cada iteração é selecionada e os elementos são sincronizados na ordem permutada.
- Ordem Puramente Randômica – A cada passo da sincronização um elemento é selecionado e sincronizado. Portanto, um elemento é sincronizado em intervalos de tamanhos arbitrários. Entretanto, os autores deixam claro que esta política para a ordem de sincronização é puramente hipotética, porém, devido ao fato de ela apresentar a maior variedade entre os intervalos, ela apresenta um bom ponto de comparação em relação as outras políticas para a ordem de sincronização

2.4 Horários de Sincronização

Em alguns casos, escolher o melhor horário para realizar a sincronização entre um web site e a base de dados é extremamente importante. Por exemplo, se um web site é extremamente acessado na parte da manhã, pode ser melhor rodar o Crawler para esse web site na parte da noite, com o objetivo de evitar um aumento no número de acessos, pois sempre há a possibilidade que o web site não esteja preparado para este aumento no número de acessos.

É importante ressaltar que para a confecção do trabalho, os autores usaram uma política de sincronização uniforme e sincronizaram as bases de dados sempre no mesmo horário, pois como o trabalho citado investigou vários web sites de várias partes do mundo, identificar o fuso horário em que eles se encontram, então a tarefa de se determinar o melhor horário para sincronização, nesse caso, se torna muito complexa.

Assumindo que todos os elementos na base de dados são modificados com a mesma frequência e após estudarem todas as ordens de sincronização, os autores chegaram a conclusão de que a política de ordem fixa funciona melhor para *Web Crawler*s em geral.

Entretanto, o foco do trabalho citado são páginas consideradas "populares", ou seja, páginas que possuem seu conteúdo focado em texto. Portanto, talvez não seja possível afirmar que as mesmas políticas e ordem de sincronização sejam aplicadas a portais de conteúdo específico, como, por exemplo, Matemática ou Física.

Vários trabalhos desenvolvidos na área consideram, também, o processo de Poisson como um modelo para previsão das atualizações de páginas, como, por exemplo, em Fetterly et al.[7], um trabalho que expandiu o de Cho e Henry-Garcia tanto em termos de quantidade de páginas cobertas quanto em sensitividade da mudança. Para a obtenção dos dados, os autores utilizaram um famoso *Crawler* conhecido como *Mercator*[8] para realizar o download de 150.836.209 páginas. Este *Crawler* foi executado uma vez por semana durante 11 semanas.

Um ponto interessante deste trabalho é que os autores descobriram que muitas páginas costumam sofrer atualizações triviais ou em apenas em seu conteúdo HTML, isto é, apenas mudanças em tags HTML. Este tipo de mudança pode impactar negativamente o processo executado pelo Crawler de uma ferramenta de busca, pois dependendo de como a ferramenta verifica mudanças nas páginas existentes em suas bases de dados, esse tipo de mudança apenas em seu conteúdo HTML pode fazer com que o Crawler da ferramenta obtenha a "nova" página e a ferramenta a indexe em sua base de dados. Porém, como a mudança não foi significativa, todo este processo terá sido em vão, desperdiçando, assim, recursos do servidor. Portanto, analisar corretamente a significância da atualização de um web site antes de atualizar o mesmo na base de dados é de extrema importância para economizar recursos para uma empresa que possua uma ferramenta de busca.

Focando, ainda, na análise das taxas de atualização ideais, Ford D., Grimes C.,e Tassone E.[9] investigaram quais os riscos envolvidos nas atualizações das bases de dados realizadas pelos Crawlers, como permitir que as base de dados fiquem mais tempo sem atualizar, gerando um possível risco de tais páginas se tornarem obsoletas. Investigaram várias taxas de atualização a fim de se chegar no que seria considerado um intervalo ótimo. Modelando, também, um processo de Poisson para auxiliar nesta tarefa.

Ferramentas de busca de conteúdo geral tratam a informação buscada pelo usuário na forma textual, o que dificulta, por exemplo, a busca por fórmulas matemáticas. Tal problema não ocorre nas ferramentas Tangent e SearchOnMath. As duas possuem o mesmo propósito, com a diferença de que a primeira é usada para buscar, exclusivamente, na Wikipedia, e a segunda realiza buscas em diversos portais.

Gonzaga F.B.[10], desenvolveu a SerchOnMath com o objetivo de permitir a busca por fórmulas matemáticas de uma forma não textual e, sim, uma busca por fórmulas matemáticas. Tal ferramenta foi desenvolvida no LaReS, o que permitiu acesso ao fator de similaridade calculado por ela e utilizado para o desenvolvimento do mesmo.

O acesso a ferramenta permitiu que fosse possível configurar um servidor onde uma versão específica, da SearchOnMath, contendo apenas algumas bases de dados selecionadas como objeto de estudo, foi hospedada.

3
Referencial Teórico

3.1 Web Crawler

Web Crawler, também conhecidos como *Robots*, *Spiders* ou apenas *Crawlers*, é um nome dado a softwares desenvolvidos para obtenção de conteúdo de web sites, seja ele uma cópia completa do site ou uma parte dele.
Crawlers são utilizados por todas as ferramentas de busca para obter cópias completas de web sites e armazená-los em seus bancos de dados, com objetivo de oferecer melhores resultados aos usuários das mesmas.
O *Web Crawler* desenvolvido para obtenção do conteúdo da DLMF, foi desenvolvido como parte de um projeto de Iniciação Científica no LaReS por Rey, D.F.[11].
Geralmente, tais ferramentas possuem o funcionamento exibido pela Figura 3:

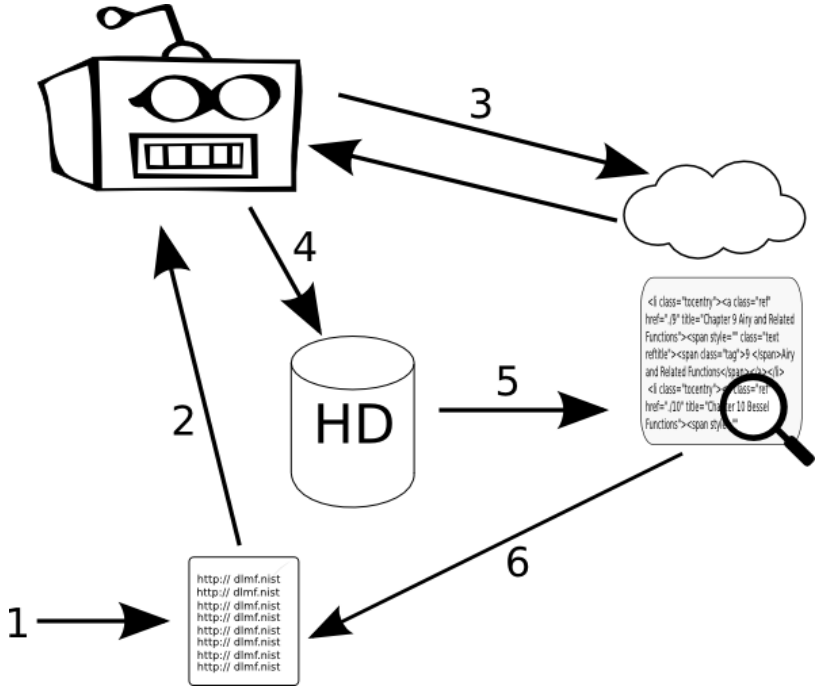

Figura 3 representação o funcionamento de um WebCrawler.

1 – Uma lista contendo vários links é enviada ao Crawler;
2 – O primeiro o link da lista é selecionado e é acessado pelo Crawler;
3 – O Crawler "recebe" o conteúdo da página acessada;
4 – O conteúdo da página é armazenado no em um banco de dados;
5 – O Crawler extrai todos os links que existem dentro do conteúdo da página recentemente acessada e os armazena em sua lista de links;
6 – O processo é repetido até que todos os links tenham sido acessados.

3.2 Python

Python foi a linguagem de programação escolhida para o desenvolvimento do WebCrawler utilizado neste trabalho. A linguagem foi escolhida pois a mesma é orientada a objetos de propósito geral, isto é, pode-se criar qualquer coisa com ela, desde simples scripts até web sites completos. Sua sintaxe simples e vasta coleção de bibliotecas permitem um desenvolvimento rápido, ao mesmo tempo em que o código é desenvolvido com uma bela endentação.

3.3 Scrapy

Scrapy[9] é um *framework* gratuito e *open source* desenvolvido para facilitar a criação de *WebCrawlers* utilizando a linguagem Python. Através do seu uso, o desenvolvimento do *WebCrawler* foi facilitado pois um framework, por definição, oferece o esqueleto do software já pronto, deixando a cargo do desenvolvedor apenas customiza-lo de acordo com suas necessidades.

O Scrapy é utilizado por várias companhias como Parserly, Lyst, ScraperWiki, entre outros. Em [12] pode ser obtida uma lista completa de companhias que utilizam este framework.

xxxviixxxvii

[9] http://scrapy.org/

3.4 WebService

Um WebService é um software sendo executado, geralmente em um servidor, e acessível pela internet. É sua responsabilidade fornecer uma ligação entre um software sendo executado por uma máquina cliente em qualquer lugar do mundo e uma outra aplicação rodando, também, no servidor.
Seu uso é extremamente comum quando deseja-se fornecer um modo fácil de se obter dados de uma aplicação sendo executada em um servidor remoto. No contexto deste trabalho, para que fosse possível criar e automatizar a busca de equações na SearchOnMath, um WebService foi implementado no servidor configurado especialmente para este trabalho. O processo funciona da seguinte forma:

- O algoritmo acessa a URL relativa a fórmula buscada;
- O WebService, percebendo o acesso, separa a fórmula desejada e a envia a SearchOnMath;
- A SearchOnMath realiza a busca, gera os resultados em arquivo no JSON e envia ao WebService;
- O WebService envia este resultado ao algoritmo;
- O algoritmo separa apenas o primeiro resultado e armazena sua similaridade em um arquivo texto. Caso nenhum resultado seja encontrado, é armazenado o símbolo "----" no arquivo texto;
- O processo é repetido até que todas as fórmulas sejam buscadas.

É necessário que a equação buscada seja escrita em TeX. Porém, esta equação deve passar por um processo de codificação HTML para que a pesquisa seja realizada com sucesso. Uma vez que o algoritmo receba, do WebService, o JSON contendo os resultados, este pode, então, ser processado. Esse arquivo de resultado contém os seguintes campos:

- currentPage: Página atual onde os resultados são exibidos. Por exemplo: Primeira página de resultados.
- errorCode: Caso, no acesso a URL, algum erro tenha acontecido, este campo contém um número referente ao erro. Caso contrário, o número 0 é utilizado.
- result: Campo que contém todos os resultados encontrados

Dentro do campo result, porém, existem os seguintes campos:

- abst: Campo que possui um resumo do texto presente na página referente ao resultado encontrado;
- equation: A equação encontrada;
- sch: Nome do schema do banco de dados onde o resultado foi encontrado;
- similarity: Grau de similaridade entre a fórmula buscada e o resultado encontrado;
- title: Título da página referente ao resultado encontrado.
- url: A URL utilizada para acessar a página referente ao resultado encontrado.

Sobre o campo similaridade, é importante informar que ela é uma medida proprietária da SearchOnMath, definida no intervalo (-∞, 1]. O valor 1,0 é alcançado para expressões exatamente iguais tanto do ponto de vista literal quanto em tamanho. Por exemplo, se um usuário buscar por x, e ocorrer em uma página x, o valor de similaridade será igual a 1 nesse caso. Caso não ocorra x, mas ocorra y, o valor já é reduzido um pouco, indicando que é uma expressão semelhante, mas não exata. Quanto maior a distância estrutural e literal entre a fórmula buscada, e uma determinada fórmula extraída do banco de dados, menor será o valor de similaridade. A ordenação dos resultados da SearchOnMath se baseia principalmente nessa medida. Observa-se de maneira empírica que geralmente, valores de similaridade ≥ 0.6 já são satisfatórios.

3.4.1 Exemplo de pesquisa realizada através do WebService

Equação buscada: x = y + 2
Equação codificada para formato HTML: x%3Dy%2B2
URL utilizada para a busca:
http://173.254.246.22/webservice/?equation= x%3Dy%2B2
É possível também adicionar o parâmetro &page=I na url utilizada para busca, onde I indica qual página de resultados é desejada.
Exemplo: http://173.254.246.22/webservice/?equation= x%3Dy%2B2 &page=1

As Figuras 4 e 5 mostram a forma que os resultados são exibidos quando a busca é feita diretamente na SearchOnMath pelo navegador e quando ela é realizada pelo WebService. Os campos presentes no arquivo JSON, “abst”, “equation”, “title”, “url”, “similarity” e “sch”, estão destacados na Figura 4. Já na Figura 5, apenas os campos “title”, “equation”, “abst” e “sch”, pois o campo “url” é visível ao passar o mouse por cima do título da página e o campo “similarity” só é visível através da consulta pelo WebService.

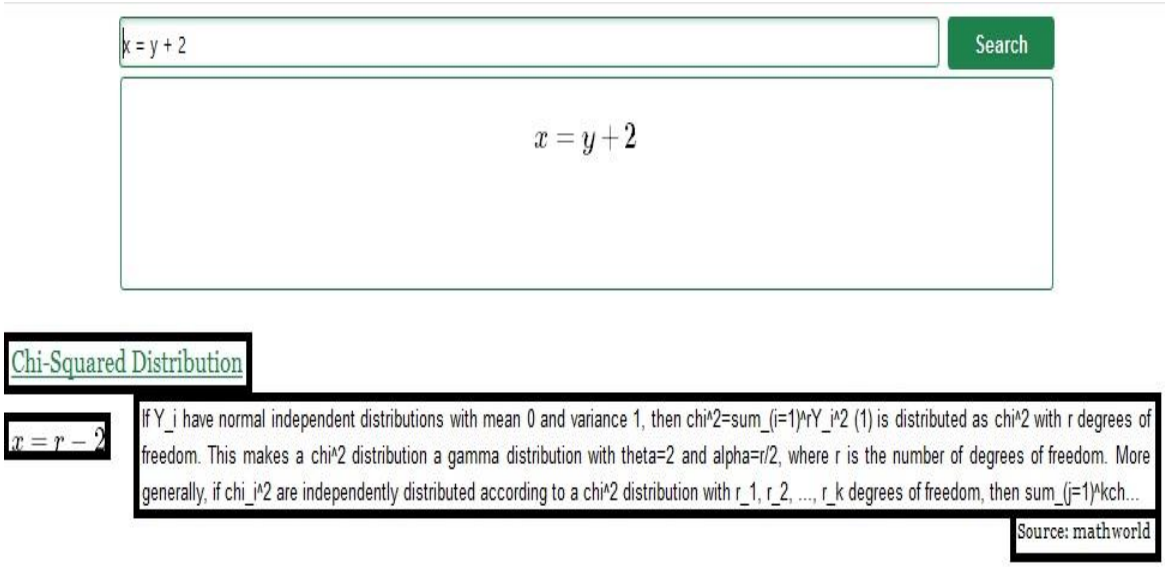

Figura 4 Fórmula buscada na SearchOnMath.

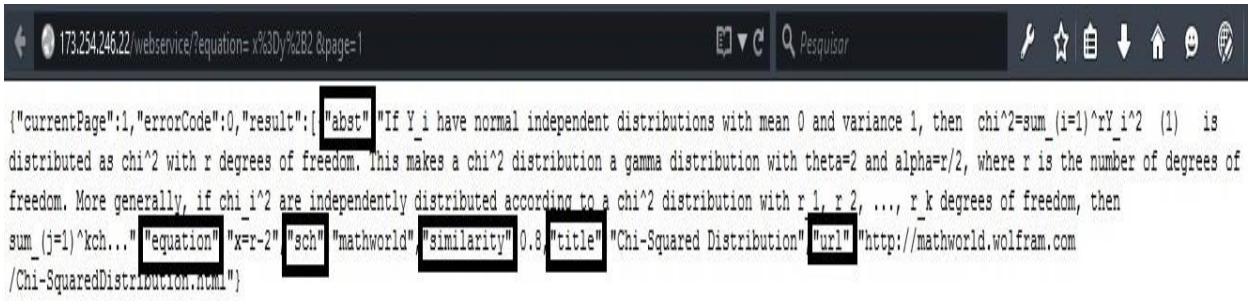

Figura 5 Fórmula buscada através do WebService.

3.5 JSON

"JSON é um formato de texto que facilita a troca de dados estruturados entre todas as linguagens de programação." (The JSON Data Interchange Format, 2013).

O JavaScript Object Notation – JSON – é um formato de arquivo de texto extremamente utilizado em aplicações no mundo todo justamente devido ao fato de possuir uma sintaxe simples e ser extremamente leve, o que permite sua transmissão pela internet de forma simples.

4
Metodologia

Para este trabalho, foi verificado se a mudança em páginas de bibliotecas digitais matemáticas como a DLMF e o MathWorld são significativas a ponto de justificar a atualização da base de dados da SearchOnMath. Também foi verificado se as fórmulas que deixaram de existir entre uma base de dados e outras impactaram de forma negativa a busca.

Para isso, foram selecionadas duas versões diferentes da base de dados da DLMF e três versões diferentes da base de dados do MathWorld. A DLMF realiza mudanças mais esporádicas, sendo que algumas vezes ela foi atualizada apenas depois de mais de 1 ano. Devido a este fato, foram escolhidas as bases de dados de 25/05/2014 e de 29/08/2014 para a DLMF, sendo a base de dados de 29/08/2014 a que, hoje, está on-line no site da DLMF.

Em relação ao MathWorld, foram escolhidas 3 bases de dados diferentes, uma de 05/2014, outra de 09/2014 e outra de 05/2015. Essas bases de dados receberam estes nomes, específicos com data completa, pois as mesmas podem ser baixadas em menos de um dia**,** enquanto as do MathWorld precisam de vários dias para serem obtidas**.**

Foram considerados os seguintes cenários para a DLMF na realização do trabalho:

- **Cenário 1**: Fórmulas existentes apenas na base de dados de 29/08/2014 foram buscadas numa versão da SearchOnMath utilizando a base de dados de 25/05/2014;
- **Cenário 2**: Fórmulas existentes apenas na base de dados de 25/05/2014 foram buscadas numa versão da SearchOnMath utilizando a base de dados de 29/08/2014.

No **Cenário 1**, a ideia é verificar se a SearchOnMath consegue encontrar páginas que satisfaçam o usuário mesmo utilizando uma versão mais antiga da base de dados. Os resultados obtidos podem indicar se uma ferramenta de busca deve, ou não, atualizar sua base de dados.

No **Cenário 2,** a ideia é verificar se a SearchOnMath consegue encontrar páginas que sejam satisfatórias ao usuário, mesmo que sejam buscadas fórmulas que deixaram de existir quando a DLMF atualizou sua base de dados. Os resultados podem indicar se os resultados da busca de uma ferramenta de busca seriam impactados, ou não, negativamente.

Os seguintes cenários foram considerados para o MathWorld na realização do trabalho:

- **Cenário 1:** Fórmulas existentes apenas nas bases de dados de 09/2014 e de 05/2015 foram buscadas numa versão da SearchOnMath utilizando a base base de dados de 04/2014.
- **Cenário 2:** Fórmulas existentes apenas nas bases de dados de 05/2014 e 05/2015 foram buscadas numa versão da SearchOnMath utilizando a base de dados de 09/2014.
- **Cenário 3:** Fórmulas existentes apenas nas bases de dados de 04/2014 e 09/2014 foram buscadas numa versão da SearchOnMath utilizando a base de dados de 05/2015.

No **Cenário 1,** a ideia é de verificar se a SearchOnMath consegue encontrar páginas com resultados que satisfaçam ao usuário quando forem buscadas fórmulas que existem nas duas versões mais novas da base de dados do MathWorld. Os resultados podem indicar se uma ferramenta de busca deve atualizar, ou não, sua base de dados referente a este portal.

No **Cenário 2,** a ideia é verificar se a SearchOnMath é capaz de encontrar páginas com resultados satisfatórios para o usuário quando forem buscadas fórmulas que deixaram de existir quando o MathWorld atualizou sua base de dados de 05/2014 para 09/014 e fórmulas que só passaram a existir quando o portal atualizou sua base de dados de 09/2014 para 05/2015. Os resultados podem indicar o quão impactante, para ferramentas de busca, quando fórmulas deixaram de existir na atualização de 05/2014 para 09/2014 e, considerando as buscas relativas apenas as fórmulas incluídas na atualização entre 09/2014 e 05/2015, se as ferramentas de busca devem, ou não, atualizar suas bases de dados referentes ao MathWorld.

No **Cenário 3,** a ideia é verificar se a SearchOnMath é capaz de encontrar resultados satisfatórios ao usuário quando são buscadas fórmulas que existem apenas nas bases de dados de 05/2014 e 09/2014. Os resultados podem indicar se ferramentas de busca seriam impactadas negativamente ao usarem a versão mais nova da base dados.

Considerando os cenários descritos para a DLMF e o MathWorld, o trabalho foi executado da seguinte forma:

- Comparar as bases de dados do mesmo domínio e gerar um arquivo contento as fórmulas existentes em uma base que não existem na outra;
- Buscar cada fórmula na ferramenta SearchOnMath e montar um arquivo contento a similaridade entre a fórmula buscada e a primeira fórmula encontrada. Foi considerada apenas a similaridade entre a fórmula buscada e o primeiro resultado encontrado, pois, caso o primeiro resultado já não seja satisfatório, os demais também não serão. Portanto, é desejável medir se apenas um resultado já é satisfatório ao usuário.
- Verificar quantas similaridades normalizadas estão entre [< 0,0 – 0,0), [0,0 – 0,2), [0,2 – 0,4) , [0,4 – 0,6) , [0,6 – 0,8) e [0,8 – 1,0];
- Gerar os gráficos contendo os resultados

.

4.1 Obtenção das diferentes bases de dados dos dois domínios estudados

Ao longo de um ano, diferentes bases de dados de cada um dos domínios foram obtidas. Em relação a DLMF, foi obtida uma versão em 25/05/2014 e uma outra quando ela foi atualizada em 29/08/2014. Apenas duas versões diferentes foram baixadas pois desde agosto de 2014 a DLMF não é atualizada. Já em relação ao MathWorld, foi feito o download de três versões, uma em maio de 14, outra em setembro de 2014 e uma última em maio de 2015. Cada uma destas versões corresponde a uma data de atualização desta biblioteca.

Para obtenção das bibliotecas foi utilizada uma ferramenta chamada *WebCrawler*. Porém, para cada uma das bibliotecas foi utilizado um *WebCrawler* diferente. Enquanto a obtenção do MathWorld foi realizada com o uso do HTTrack[10], um *WebCrawler open source* e disponibilizado gratuitamente na Internet, o download da DLMF foi realizado através de um *WebCrawler* desenvolvido pelo LaReS – Laboratório de Redes de Computadores e Sistemas Distribuídos – da UNIFAL-MG.

O desenvolvimento *WebCrawler* exclusivo para obtenção da DLMF se deve ao fato de que o HTTrack apresentava problemas para o download dessa biblioteca. Entretanto, devido a restrições impostas pelo MathWorld, que não permite que *WebCrawlers* baixem seu conteúdo de forma automatizada, o uso do HTTrack se fez necessário. Ao detectar algum *WebCrawler* tentando baixar conteúdo da página, o servidor onde o MathWorld está hospedado bloqueia o endereço IP utilizado pela ferramenta, dificultando, assim, sua obtenção. Porém, o HTTrack possui uma opção para controle de banda, tornando possível configurar um atraso entre um acesso a uma página e outra, de forma a fazer com que o servidor pense que este *WebCrawler*, é, na verdade, um usuário comum. Tal procedimento consegue garantir a obtenção completa da biblioteca.

xlvixlvi
[10] https://www.httrack.com/

4.2 Selecionando as fórmulas

Para selecionar as fórmulas, foi necessário comparar as bases de dados pertencentes ao mesmo portal matemático entre si. Portanto, foi utilizada uma simples consulta SQL, onde foram selecionadas apenas as fórmulas existentes em uma base de dados e não existentes na outra.
Exemplo:
Considerando o portal MathWorld e as bases de dados Maio/15 e Maio/14, onde, dentro do SGBD MySQL a primeira possui o nome de mathworld_maio_2015 e a segunda possui o nome de mathworld_maio_2014.

```
"SELECT equ_equation FROM mathworld_maio_2015.tb_equation where
equ_equation not in (
    SELECT equ_equation FROM mathworld_maio_2014.tb_equation ) "
```

Portanto, para ambos os portais estudados e para cada uma de sua base de dados, foi executada a consulta acima. Ao final de cada execução, os resultados foram exportados para um arquivo texto.

5
Resultados

5.1 DLMF

Primeiramente, uma comparação entre a base de dados da DLMF de 25/05/2014 e a base de dados de 29/08/2014 foi realizada. Neste primeiro caso existem 141 fórmulas na base de dados de maio que deixaram de existir na base de dados de agosto. Cada uma destas fórmulas foi buscada utilizando a ferramenta SearchOnMath e, para todas, a ferramenta conseguiu encontrar alguma fórmula semelhante. Os resultados podem ser vistos na figura 6.

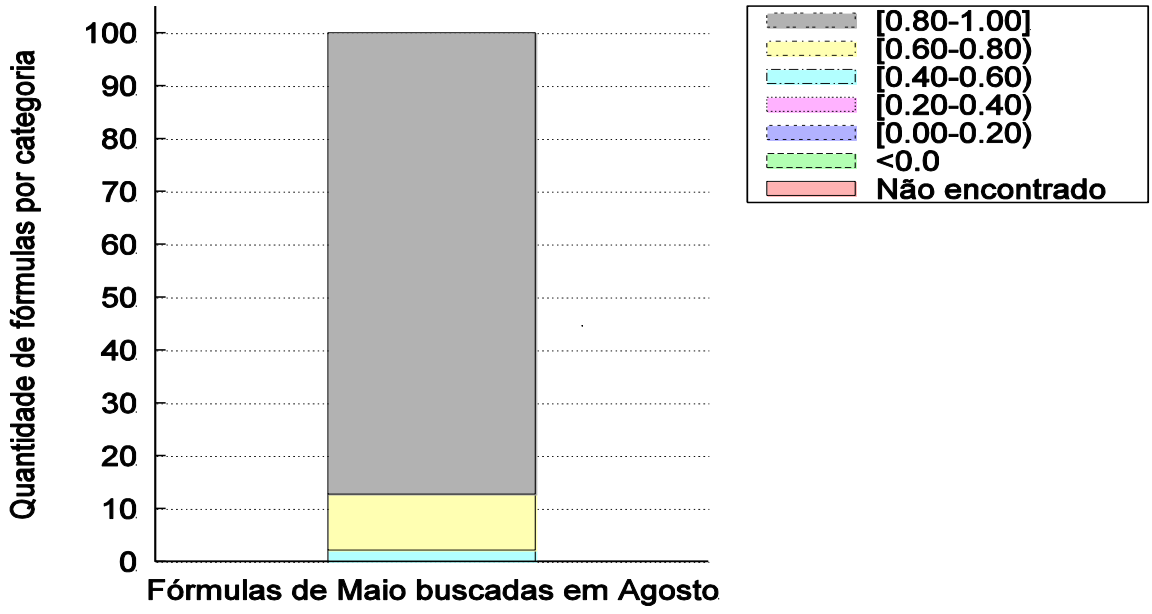

Figura 6 Similaridade entre as fórmulas buscadas na base de dados de 29/08/2014 e o primeiro resultado encontrado para cada uma.

Como é possível verificar, mesmo que na atualização da base de dados de maio para agosto 141 fórmulas deixaram de existir, caso uma ferramenta de busca utilizasse a base de dados mais nova, a busca não seria impactada, pois mais de 80% das buscas obtiveram similaridades entre 0.8 e 1.0.
Podemos concluir se a atualização da base de dados é necessária ou não ao compararmos a base de dados de 29/08/2014 com a de 25/05/2014 . Neste caso, existem 123 fórmulas na base de dados de agosto que não existem na base de dados de maio. E, para 122 das 123 fórmulas, a ferramenta conseguiu encontrar alguma fórmula semelhante. Sendo que em apenas uma fórmula nada foi encontrado pela ferramenta. A figura 7 apresenta os resultados.

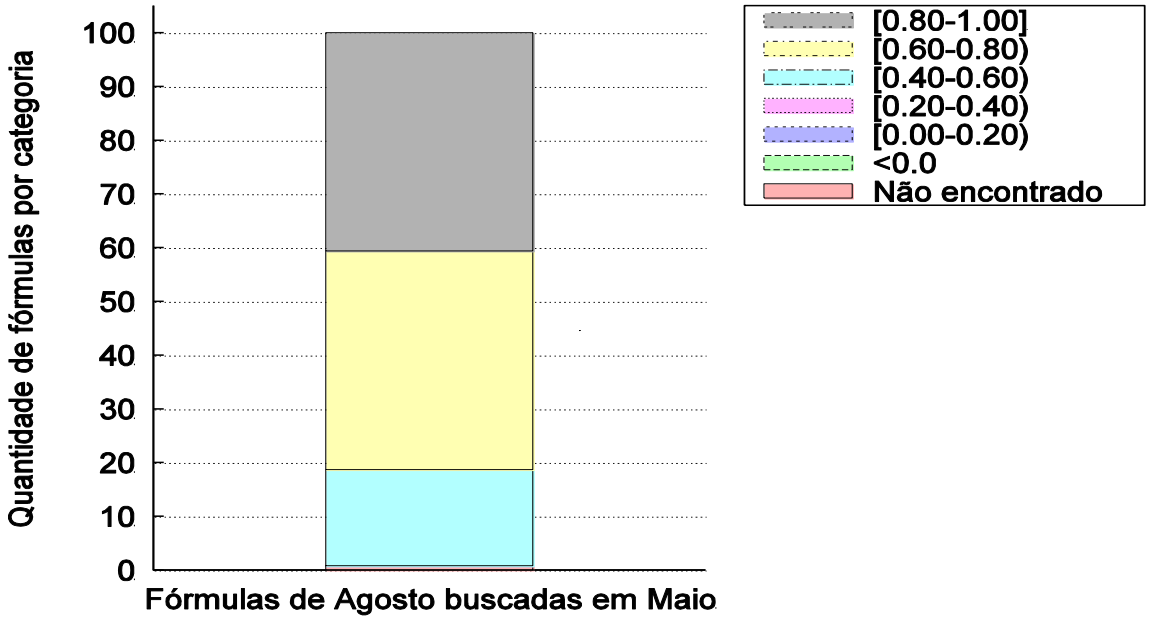

Figura 7 Similaridades entre as fórmulas buscadas na base de dados de 25/05/214 e o primeiro resultado encontrado para cada fórmula buscada.

O gráfico acima deixa claro que 40% dos resultados encontrados possuem similaridade entre 0.8 e 1.0 e 40% possuem entre 0.6 e 0.8, o que classifica 80% dos resultados como satisfatórios. Levando, então, a conclusão de que mesmo com 1 fórmula não sendo encontrada, a atualização da base de dados não é justificada .

5.2 MathWorld

Para o MathWorld, as comparações foram feitas da seguinte forma:

- Fórmulas que existem na base de dados de Maio/14 e não existem nas bases de dados de Setembro/14 e Maio/15;
- Fórmulas que existem na base de dados de Setembro/14 e não existem nas bases de dados de Maio/14 e Maio/15;
- Fórmulas que existem na base de dados de Maio/15 e não existem nas bases de dados de Maio/14 Setembro/14

No primeiro caso, comparando Maio/14 e Setembro/14, foram encontradas 58 fórmulas que existem na primeira e não existem na segunda. Sendo que para cada uma delas foram encontradas fórmulas semelhantes pela SearchOnMath. Ao comparar Maio/14 com Maio/15 foram encontradas 132 fórmulas que existem em Maio/14 e não existem em Maio/15. Neste caso, também, para cada uma das 132 fórmulas foram encontradas fórmulas semelhantes pela SearchOnMath. A figura 8 apresenta os resultados.

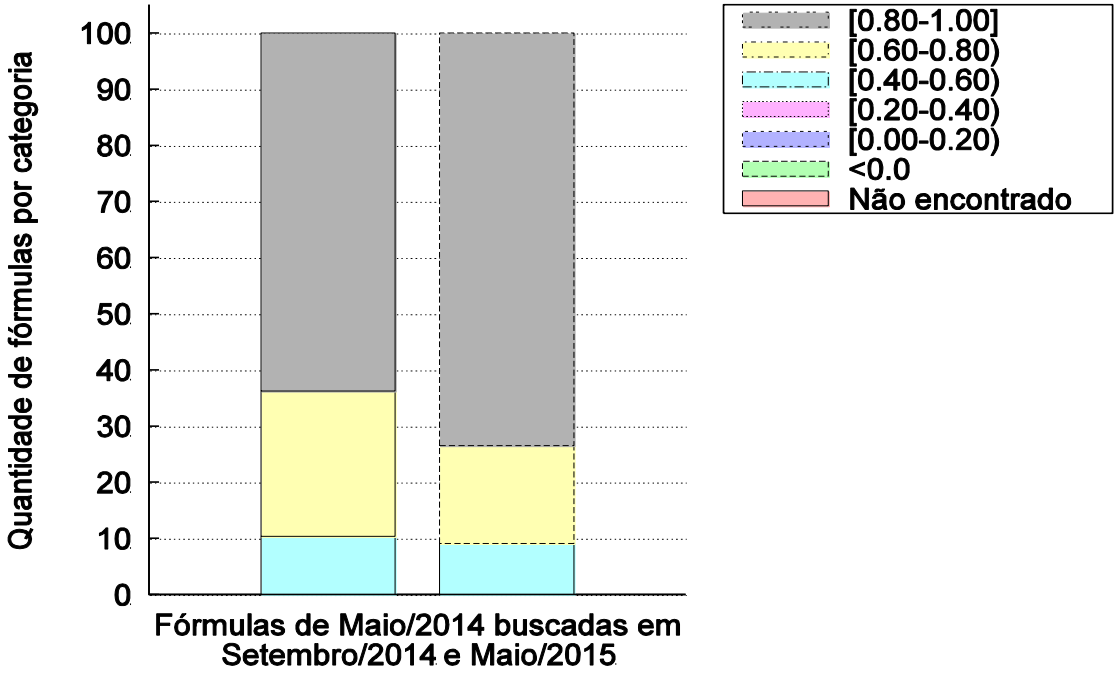

Figura 8 Similaridade entre as fórmulas buscadas nas bases de dados de Setembro/2014 e Maio/15 e o primeiro resultado encontrado para cada uma das fórmulas buscas.

Como é possível ver, ao comparar Maio/14 e Setembro/14, mais de 60% das similaridades encontradas estão entre 0.8 e 1.0 e mais de 20% estão entre 0.6 e 0.8. Semelhante resultado acontece ao comparar Maio/14 com Maio/15, onde mais 70% das fórmulas possuem similaridade entre 0.8 e 1.0 e mais de 10% possuem similaridade entre 0.6 e 0.8.

Nos dois casos, existem mais de 90% das fórmulas com similaridades aceitáveis, levando a conclusão de que mesmo as fórmulas que deixaram de existir em atualizações posteriores a Maio/14 e caso alguma ferramenta de busca utilizasse a base de dados do MathWorld de Setembro/14 ou Maio/15, o resultado das buscas não seria impactado de forma negativa.

Para o segundo caso estudado, ao comparar as bases de dados de Setembro/14 e Maio/14, foram encontradas 664 fórmulas que existem na primeira e não existem na segunda. Sendo que ao todo, foram encontradas 653 fórmulas semelhantes através da SearchOnMath. Ao compararmos Setembro/14 com Maio/15, temos 78 fórmulas que existem na primeira e não existem na segunda, sendo que foram encontrados resultados semelhantes para cada uma das 78 fórmulas. A figura 9 apresenta os resultados.

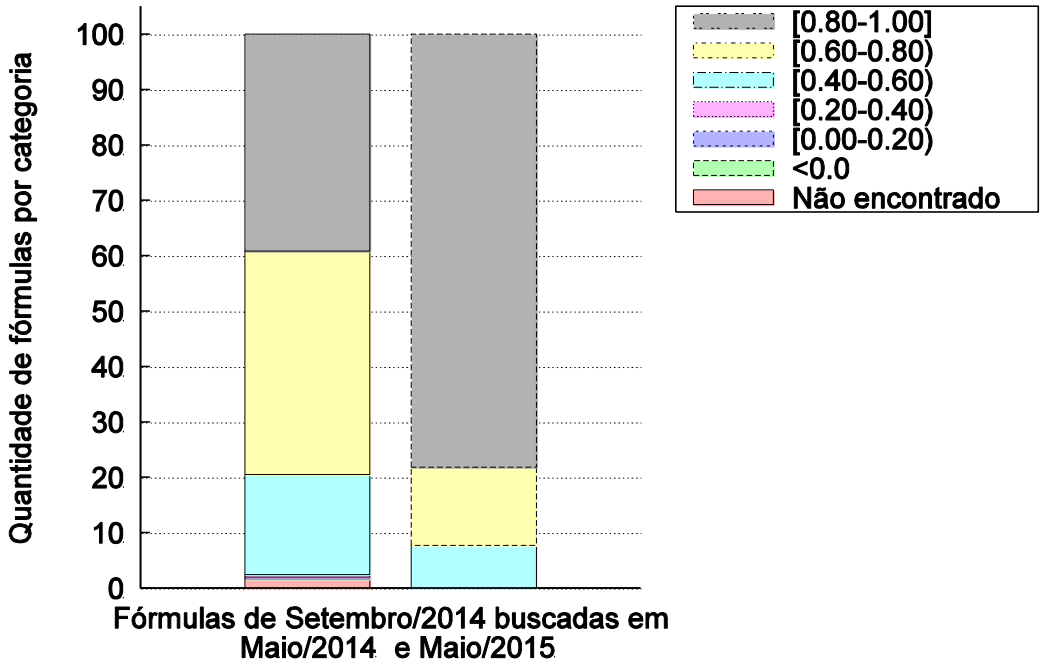

Figura 9 Similaridade entre as fórmulas buscadas nas bases de dados Maio/14 e Maio/15 e o primeiro resultado encontrado para cada uma das fórmulas buscadas.

Como é possível verificar, ao comparar Setembro/2014 com Maio/2014, temos que 80% dos resultados possuem similaridade aceitável, isto é, similaridades entre 0.6 e 1.0. Tal resultado não justifica a atualização da base de dados de Maio/2014 para Setembro/2014, mesmo que um número significativo de fórmulas tenha deixado de existir.

Em relação as fórmulas que deixaram de existir na atualização de Setembro/14 para Maio/15, mesmo que fosse utilizada a base de dados de Maio/15 e o usuário buscasse uma dessas fórmulas, o resultado não seria impactado negativamente pois neste caso um pouco mais de 90% das similaridades encontradas estão entre 0.6 e 1.0, o que representa um resultado aceitável.

Por fim, foram, primeiramente, comparadas as bases de dados de Maio/15 e Maio/14 e depois as bases de dados de Maio/15 e Setembro/14. No primeiro caso, temos cenário onde a base de dados usada é a de Maio/14 e foram buscadas fórmulas que existem apenas em Maio/15, totalizando 1091 fórmulas, sendo que apenas 20 delas não foram encontradas pela SearchOnMath.

Para o segundo caso, onde o cenário é de que a base de dados usada é a de Setembro/14 e foram buscadas nela apenas fórmulas que existem na base de dados

de Maio/15, temos 431 fórmulas que se encaixam nesse contexto. Apenas 9 fórmulas não foram encontradas pela ferramenta. A figura 10 apresenta os resultados.

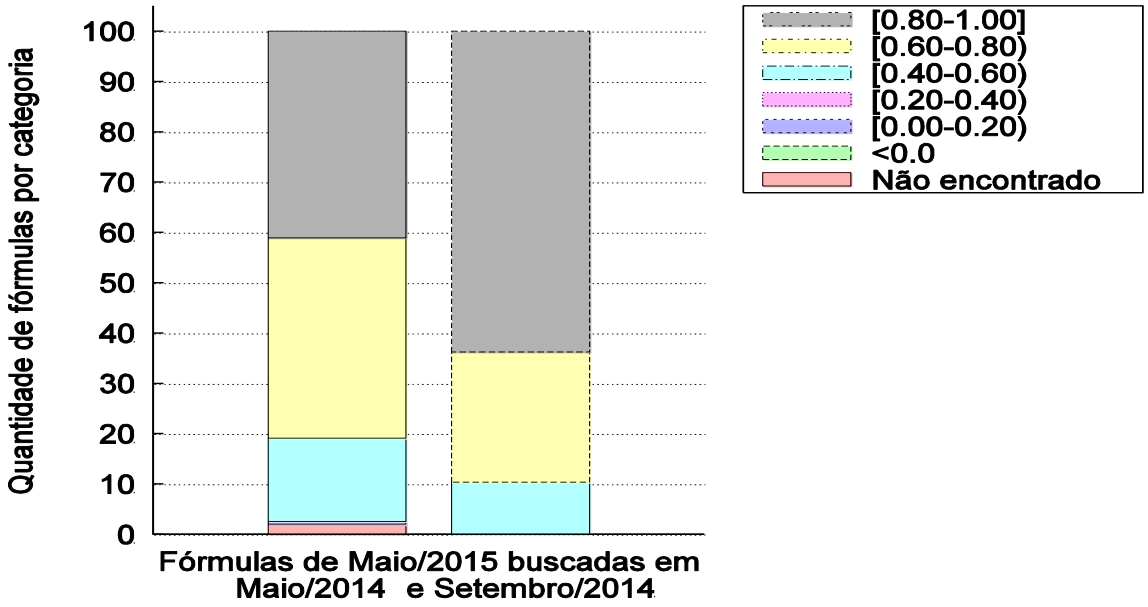

Figura 10 Similaridade entre as fórmulas buscadas na bases de dados de Maio/14 e Setembro/14 e o primeiro resultado encontrado para cada uma das fórmulas buscadas.

Como pode ser verificado, caso a base de dados usada por uma ferramenta de fosse a de Maio/14, as novas fórmulas que surgiram apenas em Maio/15 ao serem buscadas na ferramenta, teriam um resultado satisfatório, pois um pouco mais de 80% das possuem similaridades entre 0.6 e 1.0.

Caso a base de dados usada por uma ferramenta fosse a de Setembro/14 e fossem buscadas apenas as fórmulas que existem em Maio/15, não existiria um impacto negativo, pois 90% dos resultados obtidos possuem similaridade entre 0.6 e 1.0.

6 Conclusões

6.1 Considerações Finais

Os trabalhos estudados para a confecção deste foram focados em analisar o quanto as páginas mudavam considerando, em geral, o conteúdo textual das páginas. Entretanto, este trabalho concentrou-se em analisar não a frequência das taxas de atualização, mas sim o quão impactante elas seriam para ferramentas de buscas matemáticas.

Este trabalho focou-se em analisar duas das mais conhecidas bibliotecas digitais matemáticas, verificando se as mudanças nas fórmulas presentes nas mesmas eram significativas ao ponto de justificar a atualização da base de dados de qualquer ferramenta de busca.

Selecionando bases de dados de datas diferentes da mesma biblioteca, como 25/05/2014 e de 29/08/2014 da DLMF e Maio/14, Setembro/14 e Maio/15 do MathWorld, selecionando apenas as fórmulas que existem em uma base de dados e não existem na outra e utilizando a ferramenta SearchOnMath para buscar estas fórmulas na base de dados desejada, foi possível concluir que em nenhum dos casos a atualização das bases de dados seria necessária pois os resultados foram satisfatórios.

6.2 Recomendações para trabalhos futuros.

Ao estudar os trabalhos presentes na literatura, fica claro que o processo de Poisson é extremamente importante para a previsão de atualização de páginas da web.
Devido a este fato e ao fato de que com os resultados não é possível afirmar que todas as atualizações sofridas por portais matemáticos digitais serão sempre não significativas.
Portanto, uma sugestão seria que a partir de atualizações consideradas significantes de portais de conteúdo matemático, investigar a possibilidade de se modelar um processo de Poisson a fim de prever atualizações mais significantes e, assim, permitir que as ferramentas de buscas atualizem as bases de dados relacionadas a estes portais apenas quando for realmente necessário.
Outra sugestão, seria investigar se as políticas de sincronização e ordens de sincronização propostas por e CHO e GARCIA-MOLINA podem ser aplicadas a portais de conteúdo específico, como o matemático, a fim de obter-se uma sincronização de base de dados com melhor impacto para uma ferramenta de busca deste tipo de conteúdo.

7 Referências Bibliográficas

[1] DLMF, http://dlmf.nist.gov/, acesso em 29/06/2015.

[2] MathWorld, http://mathworld.com/, acesso em 29/06/2015.

[3] SearchOnMath, http://searchonmath.com/, acesso em 29/06/2015.

[4] Junghoo C. e Garcia-Molina H. The Evolution of the Web and Implications for an Incremental Crawler, 1999.

[5] Poisson S.D., Probabilité des jugements en matière criminelle et en matière civile, précédées des règles générales du calcul des probabilitiés, 1837, p. 206.

[6] Junghoo C. e Garcia-Molina H. Effective Page Refresh Policies for Web Crawlers, 2003.

[7] Fetterly D., Manasse M., Najork M. e Wiener J. A Large-Scale Study of the Evolution of Web Pages, 2003.

[8] Heydon A. e Najork M. Mercator: A Scalable, Extensible Web Crawler, 1999.

[9] Ford D., Grimes C. e Tassone E. Keeping a Search Engine Index Fresh: Risk and optimality in estimating refresh rates for web pages, 2008.

[10] Gonzaga F. B., 2013. Recuperação de Informação Orientada ao Domínio da Matemática

[11] Rey D. F., Desenvolvimento de um WebCrawler para obtenção de conteúdo matemático a partir de bibliotecas digitais, 2011.

[12] Companie that are using Scrapy, http://scrapy.org/companies/, acesso em 29/06/2015.

8 Apêndice

8.1 DLMF

Exemplos de fórmulas existentes na base de dados de 25/05/2014 e não existentes na base de dados de 29/08/2014 e a similaridade encontrada pela SearchOnMath quando a fórmula é buscada apenas na base de dados de 29/08/2014.

Tabela 1 - Exemplos de fórmulas buscas e a similaridade encontrada entre o primeiro resultado e a fórmula buscada.

Fórmula buscada	Similaridade
\sqrt{E/2}	0.96875
\displaystyle>e_{3}.	0.75
W^(1,p)(Omega)	0.88

8.2 MathWorld

Exemplos de fórmulas existentes na base de dados Maio/15 e não existentes na base de dados Maio/14 e a similaridade encontrada pela SearchOnMath quando a fórmula é buscada na base de dados de Maio/14.

Tabela 2 - Exemplos de fórmulas buscas e a similaridade encontrada entre o primeiro resultado e a fórmula buscada.

Fórmula buscada	Similaridade
Omega subset R^n	0.85
C_c^infty(Omega)	----
v in H	0.88

Apenas ressaltando que fórmulas não encontradas, apresentam o sinal "----" como similaridade.