

## Um Mecanismo de Consulta Temporal por Palavras-Chave em Páginas Web

Aline Frozi<sup>1</sup>, Edimar Manica<sup>1,3</sup>, Renata Galante<sup>1</sup>, Carina F. Dorneles<sup>2</sup>

<sup>1</sup>Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)  
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brasil

<sup>2</sup>Depto. de Informática e Estatística – Universidade Federal de Santa Catarina (UFSC)  
Campus Universitário Trindade – 88.049-900 – Florianópolis – SC – Brasil

<sup>3</sup>Instituto Federal do Rio Grande do Sul (IFRS) – Campus Avançado Ibirubá  
Rua Nelsi Ribas Fritsch, nº 1111, Bairro Esperança – 98.200-000 – Ibirubá – RS –  
Brasil

{arfrozi, emanica, galante}@inf.ufrgs.br, dorneles@inf.ufsc.br

**Abstract.** *This paper presents a mechanism for querying text using keywords and time constraints. Throughout this document we show the system architecture and a prototype that works on Web pages: the Chronos search engine. Using this search engine, documents with time information related to the query terms receive higher scores, which is the main difference of the tool in comparison to the current search engines.*

**Resumo.** *Este artigo apresenta um mecanismo que permite a realização de consultas em textos utilizando palavras-chave e restrições de tempo. Ao longo deste documento é apresentada a arquitetura do sistema e um protótipo desenvolvido que atua sobre páginas Web: o motor de busca Chronos. Com ele, são valorizadas páginas que possuem a informação temporal dentro do mesmo contexto dos termos pesquisados – sendo este o principal diferencial da ferramenta em relação aos mecanismos de busca atuais.*

### 1. Introdução

Uma página Web possui diversas informações de tempo relacionadas a ela: a data de sua publicação, a data em que foi coletada pelo motor de busca e as datas e outros termos indicando tempo que aparecem no seu conteúdo textual. Motores de busca comerciais têm explorado o valor da informação temporal na busca Web. Eles fornecem maneiras do usuário realizar uma busca na Web baseada no tempo. Por exemplo, o Google permite definir o período de interesse, bem como a opção de visualizar os resultados da busca em uma linha de tempo. No entanto, estes motores de busca apenas consideram a data de coleta das páginas [Jin 2008].

Dessa forma, quando a página contém informações relacionadas ao passado ou ao futuro – por exemplo, uma página que foi publicada em 2011 falando sobre a Copa do Mundo de Futebol de 2002 – o tempo associado à página pelo motor de busca difere do tempo abordado no conteúdo da página, o que torna imprecisa a busca baseada no tempo oferecida pelos motores de busca atuais. Há trabalhos que propõem motores de busca temporal, como [Jin 2008], [Vicente-Díez e Martínez 2009] e [Pasca 2008].

Porém, não são adequados para páginas que descrevem várias informações temporais, conforme é discutido na Seção 5.

Para contornar esse problema, foi desenvolvida a ferramenta Chronos. Esta, ao indexar as páginas, mapeia cada fragmento de texto ao intervalo de tempo a que se refere. Por exemplo, o trecho *em 1958, o Brasil sagrava-se pela primeira vez Campeão Mundial de Futebol* é associado ao intervalo de tempo de 01/01/1958 a 31/12/1958. Dessa forma, o mecanismo de busca pode dar mais valor a páginas que têm a informação temporal desejada próxima aos termos que o usuário submeteu para pesquisa.

Este trabalho foi desenvolvido como uma extensão do trabalho [Manica 2010], e compôs o trabalho<sup>1</sup> de conclusão de curso [Frozi 2010a]. Manica et. al [Manica 2010] propõem uma classificação denominada TKC (Temporal Keyword Classification) que especifica os requisitos temporais que um mecanismo de consulta temporal por palavras-chave deve considerar. Com base nessa classificação, é proposta uma abordagem denominada TPI que suporta consultas temporais por palavras-chave em motores de busca XML, tratando os requisitos temporais definidos em TKC.

O principal diferencial do trabalho [Frozi 2010a] é tratar os requisitos temporais em motores de busca Web. Dois aspectos diferenciam os motores de busca Web dos motores de busca XML: a estrutura dos documentos consultados (não estruturados); e a existência de um mecanismo de *ranking*. A principal contribuição deste trabalho é, portanto, definir como incluir nas medidas de *ranking* a dimensão temporal. A ferramenta Chronos foi apresentada na Sessão de Demos do Simpósio Brasileiro de Banco de Dados de 2010 [Frozi 2010b]. Demais contribuições deste trabalho são: (i) o suporte a um conjunto de consultas temporais; (ii) o tratamento da informação temporal presente no conteúdo das páginas Web, identificando quais termos se referem a cada expressão temporal – permitindo tratar páginas que contêm diversas informações temporais; e (iii) o método de indexação temporal.

Chronos foi desenvolvido em Java e fez uso da biblioteca Lucene [Lucene 2010], que disponibiliza métodos para indexação e consulta de textos. Inicialmente, foi construído um motor de busca convencional. Após, foi desenvolvida uma extensão, de modo a adicionar a funcionalidade de busca temporal. A existência dos dois sistemas tornou possível a comparação da qualidade dos resultados obtidos pelos dois tipos de motores de busca.

O artigo está organizado como segue. Na Seção 2 é descrita a arquitetura do mecanismo de busca. A Seção 3 apresenta a ferramenta Chronos. Na Seção 4 são discutidos os experimentos realizados e resultados obtidos. Os trabalhos relacionados são apresentados na Seção 5. Finalmente, na Seção 6, são descritas as considerações finais e as direções futuras.

## 2. Arquitetura do Sistema

A ferramenta Chronos foi desenvolvida para suportar consultas temporais por palavras-chave sobre páginas Web. Com ela, o usuário pode restringir os resultados da consulta a um período de tempo de seu interesse.

---

<sup>1</sup> Esta monografia foi orientada por Renata Galante (UFRGS), co-orientada por Edimar Manica (UFRGS, IFRS) e teve a colaboração de Carina F. Dorneles (UFSC).

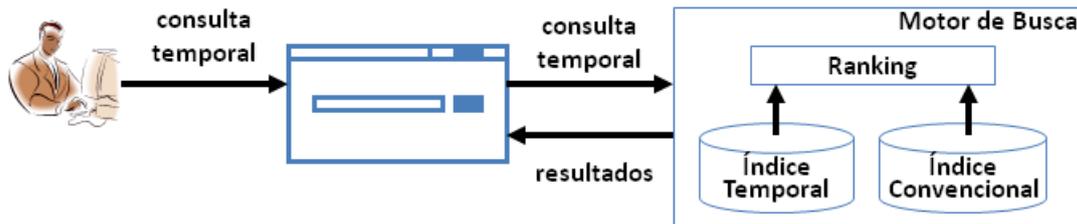


Figura 1. Visão geral da ferramenta Chronos

O funcionamento da ferramenta pode ser dividido em duas fases: a de indexação e a de pesquisa. A Figura 1 apresenta a visão geral da ferramenta. O usuário interage com uma interface fornecendo uma consulta temporal. Esta consulta temporal é constituída de palavras-chave convencionais e de expressões temporais. Por exemplo, seja considerada a consulta *senator Brazil intersect 2005*. Esta consulta é enviada a um motor de busca. O motor de busca, através de seu módulo de *ranking*, processa a consulta, acessa um índice invertido convencional e um índice temporal, e retorna os resultados ordenados de acordo a sua importância (score), que é obtida por uma média ponderada entre a relevância do texto e a relevância do tempo.

## 2.1. Especificação das Consultas

As consultas temporais seguem a forma definida no trabalho [Manica 2010]. A sintaxe é mostrada em *Backus–Naur Form* na Figura 2 a seguir. Colchetes [] indicam segmento opcional, chaves {} indicam segmento repetitivo (uma ou mais vezes), parênteses indicam um conjunto de opções, | indica “ou”, e palavras em negrito são usadas como termos pré-definidos.

```
tempQuery ::= {kws} tempPredicate
kws ::= string
tempPredicate ::= tempOp op
tempOp ::= ( After | Before | Contains | During | Equals | FinishedBy
           | Finishes | Intersect | Meets | MetBy | OverlappedBy
           | Overlaps | StartedBy | Starts )
op ::= ( instant | interval | compInterval )
instant ::= [ number / ] [ number / ] number
interval ::= [ instant , [ instant ] ] | [ , instant ]
compInterval ::= word [value] granularity
word ::= ( current | last | next )
granularity ::=
  ( Day(s) | Week(s) | Month(s) | Year(s) | Decade(s) | Century(ies) )
value ::= integer
```

Figura 2. Formato de consulta temporal

Uma consulta temporal (*tempQuery*) é composta por um conjunto de palavras-chave sem informação temporal (*kws*) e um predicado temporal (*tempPredicate*). O

predicado temporal é uma restrição temporal dividida em duas partes: operador temporal (tempOp) e operando (op). O operando é um instante ou um intervalo, que deve seguir o operador temporal. O operador temporal define a relação temporal entre o conjunto de palavras-chave convencional e o operando. É importante notar que o operador temporal cobre todas as relações básicas entre intervalos temporais definidas em [Allen 1983] e, adicionalmente, o operador INTERSECT.

Há três possibilidades para usar o operando: instante (instant), intervalo (interval) ou intervalo composto (compInterval). O instante é um ponto no tempo. Por exemplo, *2008*. O intervalo é definido por um período temporal composto por dois instantes que representam, respectivamente, o instante inicial e final do período. Estes dois instantes devem estar entre colchetes e separados por vírgula. Por exemplo, *[01/01/2008, 30/06/2008]*. Um intervalo composto define um deslocamento a partir da data atual. Por exemplo, *last 10 months* (últimos 10 meses). Alguns exemplos de consultas são: (i) *senator Brazil after 2008*; (ii) *senator Brazil intersect [01/01/2002, 30/06/2005]*; e (iii) *senator Brazil before last 4 years*.

## 2.2. Indexação

A indexação é pré-requisito para que se possam fazer consultas. Deve ser executada de forma antecipada para formar as estruturas (índices) que são posteriormente acessadas pelo motor de busca para definir a importância de cada página para uma dada consulta.

A ferramenta Chronos utiliza dois índices criados a partir das páginas Web coletadas por um *crawler*. Ambos os índices são criados *off-line*, após a coleta dos documentos. De tempos em tempos, após cada coleta, os índices são atualizados. Durante a fase de consultas, estes índices são acessados pelo mecanismo de busca.

O primeiro é um índice invertido convencional construído de acordo com técnicas tradicionais de RI [Baeza-Yates e Ribeiro-Neto 1999]. Este índice é criado através da adição de documentos, armazenando basicamente três informações de cada arquivo: caminho do arquivo, data de modificação e conteúdo.

O segundo é um índice “temporal”, que se diferencia do índice convencional principalmente no seguinte aspecto: neste índice, ao invés de serem adicionados documentos, são adicionados *nuggets* temporais. Um *nugget* é um fragmento de uma sentença que informa fatos de domínio aberto associados com alguma entidade [Pasca 2008]. Por exemplo, o fragmento “O Brasil ganhou a Copa do Mundo em 1994” é um *nugget* temporal, pois informa um fato (ganhou a Copa do Mundo) da entidade Brasil. Este fato ainda está associado a sua data de ocorrência (1994). Cada *nugget* temporal é adicionado ao índice temporal, e o intervalo temporal existente no *nugget* é armazenado em um banco de dados. O processo de construção do índice temporal possui seis etapas, conforme é mostrado na Figura 3.

A etapa **A. Processamento Linguístico** rotula as palavras do texto de cada página Web com suas funções morfosintáticas, como verbo e substantivo. Esta etapa é realizada pela ferramenta TreeTagger [TreeTagger 2010]. A etapa **B. Reconhecimento de Expressões Temporais** anota as expressões temporais presentes no texto. A etapa **C. Normalizador** normaliza as expressões temporais para um formato único. As etapas **B** e **C** são executadas pela ferramenta GUTime [GUTime 2010], que detecta e normaliza

expressões temporais em textos por meio de um autômato finito, considerando as funções morfossintáticas das palavras identificadas na etapa A.

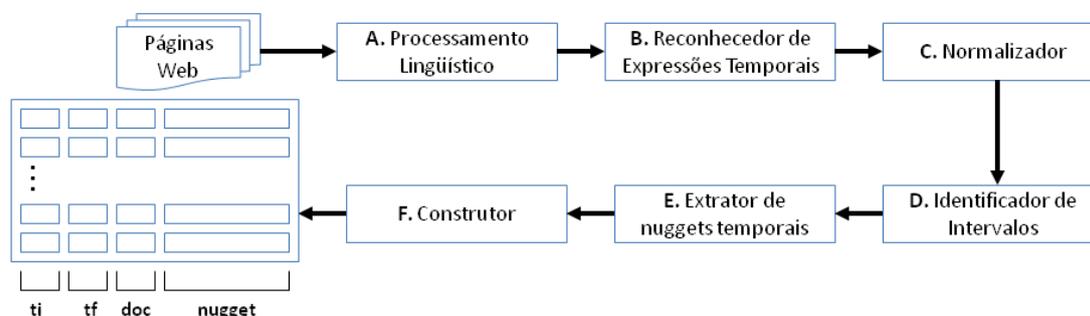


Figura 3. Detalhamento da Indexação Temporal

A etapa **D. Identificador de Intervalos** identifica quando duas expressões temporais representam um intervalo. Esta etapa é realizada a partir de alguns padrões léxico-sintáticos, tais como: *from DATE to DATE* e *between DATE and DATE*. A etapa **E. Extrator de Nuggets Temporais** extrai *nuggets* do texto que estão associados com expressões temporais. Esta etapa é baseada nos padrões léxico-sintáticos definidos em [Pasca 2008] e considera as funções morfossintáticas das palavras identificadas na etapa A. Um exemplo de padrão utilizado é *[StartOfSent] Nugget [in|on] Date [EndOfSent]*.

A etapa **F. Construtor** é responsável por criar o índice. Cada *nugget* temporal é considerado como um pseudo-documento que é indexado. Alguns campos adicionais são associados a cada *nugget*: (i) o tempo inicial (ti) do intervalo representado pela expressão temporal associada ao *nugget*; (ii) o tempo final (tf) do intervalo; e (iii) uma referência ao documento que contém o *nugget*. Quando a expressão temporal associada ao *nugget* não constitui um intervalo e sim um ponto no tempo (... *elegeu-se em 1994*, por exemplo), atribui-se seu valor tanto para o tempo inicial quanto para o final.

### 2.3. Ranking das Páginas Web

O módulo de *ranking* das páginas Web recebe como entrada uma consulta temporal e retorna os documentos correspondentes. Estes resultados são classificados de acordo com uma média ponderada entre a relevância do texto e a relevância do tempo.

Dada uma consulta temporal que é submetida no Chronos, são executados dois tipos de busca: convencional e temporal. O objetivo da busca convencional é obter o escore de cada documento que um motor de busca convencional daria para a consulta sem considerá-la temporal. O objetivo da busca temporal é dar mais valor a documentos que possuem frases que contêm, ao mesmo tempo, ambas as partes da consulta: os termos buscados e o intervalo temporal desejado.

O *ranking* convencional é obtido por funções de consulta existentes na biblioteca Lucene, que utiliza uma combinação de modelo Vetorial e modelo Booleano para determinar o escore de um documento.

A forma de obter o *ranking* temporal diferencia-se da anterior por não se basear nos documentos, mas sim nos seus *nuggets* temporais. Para computar o valor de cada documento, primeiro calcula-se o escore segundo o critério de relevância da Lucene para a consulta sem a parte temporal sobre o índice temporal. Com isso, cada *nugget*

temporal indexado recebe um *score*. Em seguida, cada *nugget* temporal cujo intervalo temporal associado não satisfaz o predicado temporal da consulta é eliminado. Finalmente, o escore da página será o escore de seu *nugget* temporal com maior escore.

O escore final de relevância de uma página é computado de acordo com a fórmula

$$\text{rank}(P) = \alpha * ksim + \beta * tsim, \quad (1)$$

onde *ksim* representa o escore de relevância do texto, dado pelo *ranking* convencional, e *tsim* representa o escore de relevância do tempo, dado pelo *ranking* temporal. Os valores de  $\alpha$  e  $\beta$  representam os pesos das variáveis. A fórmula (1) é baseada na que é apresentada em [Jin 2008], para o motor de busca TISE. As diferenças são: (i) no trabalho de [Jin 2008], não existe o cálculo de *tsim*, uma vez que não são utilizados os *nuggets* temporais; e (ii) no trabalho de [Jin 2008], é considerada na soma do escore também uma parcela referente à importância da página dada pelo algoritmo PageRank [Page 1998].

### 3. Apresentação da Ferramenta

A interface da ferramenta Chronos é Web. Assim como a maioria dos motores de busca, a ferramenta Chronos possui uma interface simples para a realização das consultas. A Figura 4 apresenta a página de busca do Chronos, e a Figura 5 apresenta um exemplo de página de resultados. A base de dados usada para teste continha páginas sobre personalidades estadunidenses. Ressalta-se que, como resumo de cada página nos resultados, é apresentado o *nugget* temporal com maior valor de escore; assim, o usuário pode, muitas vezes, encontrar a informação que necessita já nos resumos apresentados na página de resultados.

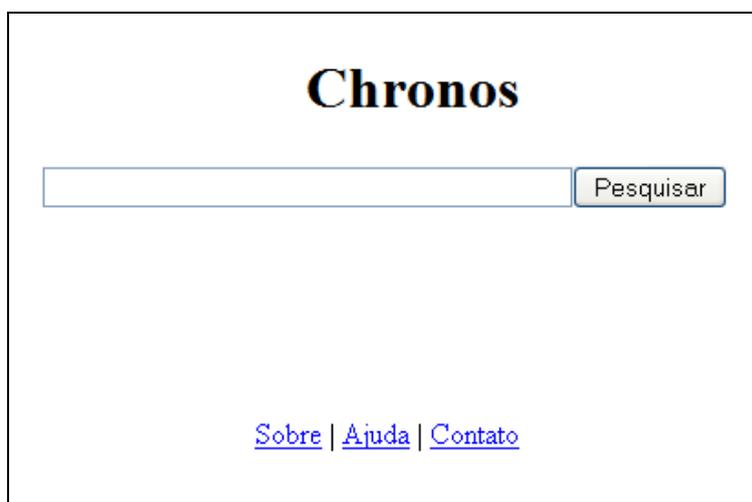


Figura 4. Página de busca

### 4. Experimentos

O objetivo dos experimentos realizados foi validar a ferramenta Chronos no quesito consultas temporais por palavras-chave. Foi medida a qualidade dos resultados retornados por ela mediante algumas consultas. Seus resultados foram comparados com

os resultados retornados por um motor de busca convencional, ambos desenvolvidos com o apoio da biblioteca Lucene.

**Chronos**

(30 resultados para a consulta "Obama intersects 2004". Mostrando 1-10).

Página	Resumo
1. <a href="#">Barack Obama - Wikipedia, the free encyclopedia</a>	In July 2004, Obama delivered the keynote address at the 2004 Democratic National
2. <a href="#">United States Senate career of Barack Obama</a>	The Making of a Senator: Barack Obama and the 2004 Illinois Senate Race
3. <a href="#">Wikipedia:Featured article candidates/Barack</a>	Ambi 01:23, 7 Aug 2004 (UTC). Support. It's fabulous that Wikipedia can be as thorough and as
4. <a href="#">Barack Obama - Wikipedia bahasa Indonesia</a>	Setelah kemenangan Maret 2004, Obama menyampaikan key notenya pada Konvensi
5. <a href="#">Obama's Wikipedia Page Distances President From Wright</a>	from February 2008, a theme of Obama's 2004 Democratic National Convention keynote ...
6. <a href="#">Obama's Wikipedia Page distances him from both Wright</a>	But according to an archived Wikipedia page for Obama from February 2008, a theme of Obama's 2004 D
7. <a href="#">Barack Obama: Biography from Answers.com</a>	Obama shot to national fame after delivering a stirring keynote speech in support of John Kerry at the 2004.
8. <a href="#">Barry Soetoro Obama - Wikipedia - Mitra Encyclopedia..</a>	Obama served three terms in the Illinois Senate from 1997 to 2004.
9. <a href="#">Transcript: Illinois Senate Candidate Barack Obama...</a>	Tuesday, July 27, 2004; 11:09 PM. Candidate for U.S. Senate in Illinois, Barack Obama...
10. <a href="#">CNN.com - Obama: Time to reclaim America's promise</a>	27 Jul 2004 ... Barack Obama, who is running for the US Senate from Illinois,

[Primeiro](#) [Anterior](#) [1](#) [2](#) [3](#) [Próximo](#) [Último](#)

[Sobre](#) | [Ajuda](#) | [Contato](#)

**Figura 5. Página de resultados**

Para os experimentos realizados, o conjunto de dados selecionado compreendeu quarenta e três páginas HTML coletadas da Wikipedia [Wikipedia 2010] com informações sobre os presidentes dos Estados Unidos. Cada página contém a biografia de um presidente. Previamente, foram construídos os índices convencional e temporal com base nessas páginas. O índice convencional foi criado a partir do conjunto completo, e o índice temporal a partir do conjunto processado.

Foram submetidas 15 consultas temporais aos dois motores de busca, considerando para análise apenas os 10 primeiros resultados. Para encontrar as páginas relevantes foram analisados manualmente os resultados de ambas as ferramentas.

A métrica para avaliação de desempenho utilizada nos experimentos foi *Precision at X*. Essa medida indica a precisão após X documentos recuperados. Foram utilizados  $X=10$ ,  $X=5$  e  $X=1$ .

A Figura 6 mostra o gráfico obtido pelo cálculo da precisão considerando os dez primeiros resultados. É possível notar que não houve um grande aumento na precisão dos resultados do Chronos em comparação com os do mecanismo convencional. Isso se deve ao fato de a medida da precisão considerar apenas a quantidade de documentos relevantes que foram obtidos e não a sua posição na lista de resultados. Os resultados relevantes com Chronos geralmente ficavam mais ao topo da lista, mas isso não é mostrado pela métrica da precisão.

A Figura 7 apresenta o cálculo da precisão para os cinco primeiros resultados. Conforme foi dito, a maioria dos resultados relevantes do Chronos ficou acima dos resultados relevantes do motor de busca convencional. Sendo assim, pode ser visto na Figura 7 que a precisão das consultas 4, 7 e 15 com Lucene diminuíram em relação ao

que foi mostrado na Figura 6, pois elas continham resultados relevantes abaixo da quinta posição do *ranking*.

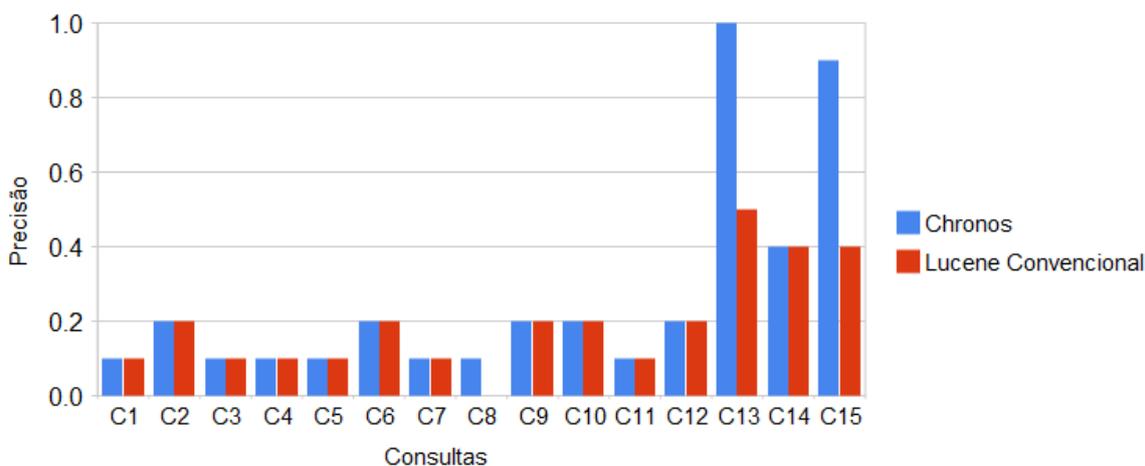


Figura 6. Precision at 10

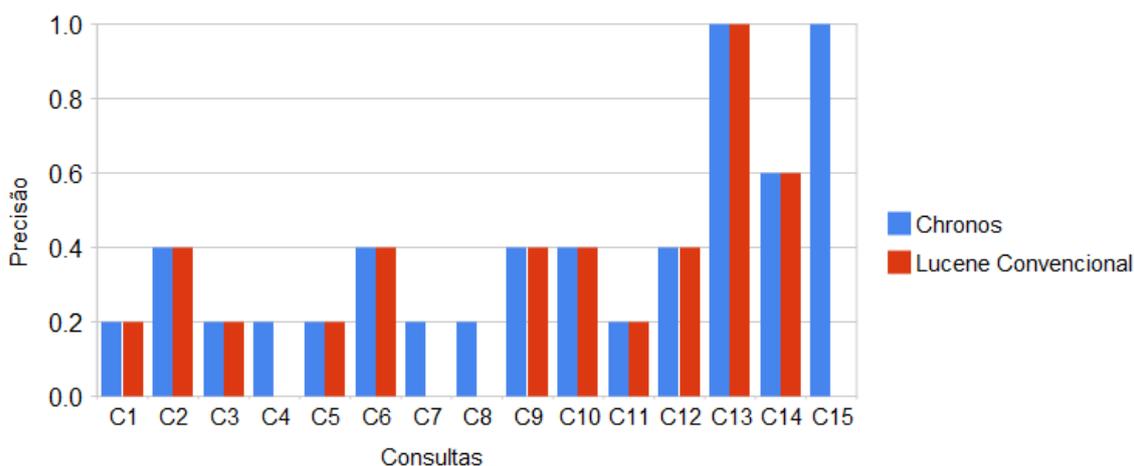
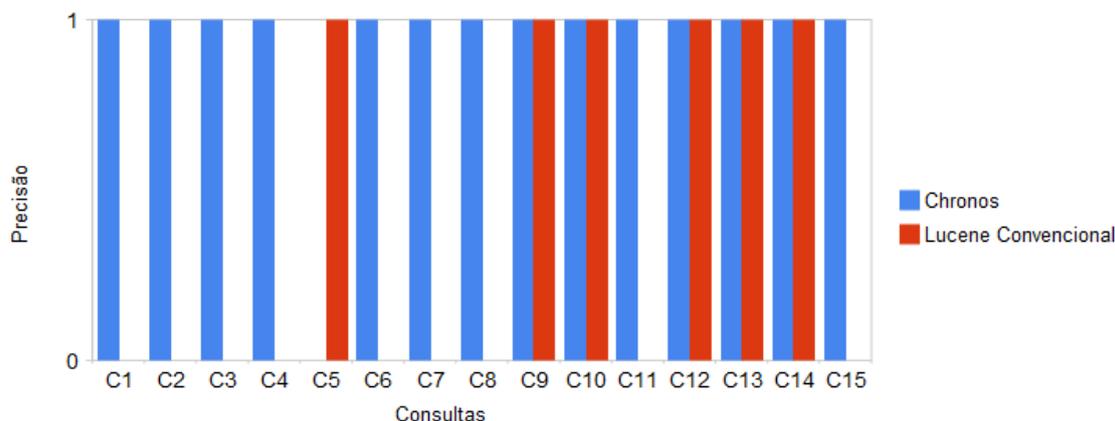


Figura 7. Precision at 5

A seguir é mostrada a Figura 8, que contém a precisão considerando apenas o primeiro resultado de cada ferramenta. Em nove das quinze consultas Chronos retornou um resultado relevante na primeira posição, enquanto que o motor convencional retornou zero. Em cinco houve empate e, em apenas um caso (C5), o mecanismo convencional retornou um relevante contra zero de Chronos.

Pode ser observado pelos resultados que a ferramenta Chronos obteve um aumento significativo na precisão quando são utilizados operadores temporais que não podem ser mapeados em uma consulta convencional (como *after* e *before*). Resultados bons também foram conseguidos quando o intervalo solicitado é maior do que um ano, já que um mecanismo de busca convencional consulta o termo correspondente ao ano que foi informado, enquanto que Chronos consulta todo o intervalo correspondente.



**Figura 8. Precision at 1**

Também é possível notar que Chronos apresentou os resultados relevantes mais ao topo. Isso ocorreu porque a ferramenta Chronos normaliza as expressões temporais presentes no conteúdo das páginas e associa a cada expressão temporal apenas os termos relacionados com ela.

No entanto, existem alguns casos em que a ferramenta Chronos precisa ser melhorada. Por exemplo, para a consulta *was assassinated intersect 1881*, havia um documento que continha *was mortally wounded in 1881* que não foi recuperado, mesmo contendo a informação desejada. O motivo de não ter sido recuperado é que esta frase não possui ambas as partes temporal e não temporal da consulta, pois, ao invés do termo *assassinated* que havia sido pesquisado, possui um sinônimo. Para resolver esse problema, pode ser utilizada a técnica de expansão de consultas.

## 5. Trabalhos Relacionados

TISE [Jin 2008], TERN [Vicente-Díez e Martínez 2009] e [Pasca 2008] são propostas de motores de busca temporal. TISE indexa apenas uma expressão temporal por página, selecionando a expressão temporal que descreve mais apropriadamente os eventos da página Web. Isto acarreta em perda de várias informações temporais presentes no conteúdo da página. TERN e [Pasca 2008] indexam todas as expressões temporais do conteúdo textual da página Web. No entanto, TERN associa todas as expressões temporais com a página, enquanto que [Pasca 2008] associa cada expressão temporal com um *nugget* temporal, formando um pseudo-documento. TISE e TERN permitem consultas de seleção temporal, onde o usuário utiliza uma restrição temporal para filtrar os resultados da consulta. Já [Pasca 2008] permite consultas de saída temporal, onde o usuário está interessado em saber qual o tempo em que um determinado evento ocorreu.

O diferencial da ferramenta Chronos é aplicar a indexação de *nuggets* temporais para consultas de seleção temporal, a fim de melhorar os resultados deste tipo de consulta associando a uma expressão temporal apenas termos que se referem a ela.

## 6. Considerações Finais

Este artigo apresentou a ferramenta Chronos, que permite a realização de consultas temporais por palavras-chave em páginas Web. As principais contribuições deste

trabalho são a identificação dos termos relacionados com cada expressão temporal presente no conteúdo da página Web e a criação de um método de indexação temporal.

Experimentos mostram que Chronos melhora a qualidade dos resultados para consultas temporais em comparação com um motor de busca convencional. Como trabalhos futuros destacam-se: (i) a realização de testes com diversos usuários e inúmeras consultas para validar a ferramenta; (ii) a utilização da estrutura da página Web para melhorar os resultados; e (iii) a utilização de expansão de consultas para tratar sinônimos.

## Referências

- Allen, J. F. (1983). “Maintaining knowledge about temporal intervals”. *Commun. ACM*, 26(11).
- ANNIE. (2009). “Open source information extraction”, <http://www.aktors.org/technologies/annie/>. Abril, 2010.
- Baeza-Yates, R. A. e Ribeiro-Neto, B. (1999). “Modern Information Retrieval”. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- Frozi, A. (2010a). “Um mecanismo de consulta temporal por palavras-chave em páginas Web”. Monografia de Graduação, Instituto de Informática, UFRGS.
- Frozi, A.; Manica, E.; Galante, R. e Dorneles, C. F. (2010b). “Chronos - um motor de busca temporal”. Sessão de Demos, SBBD.
- GUTime. “Adding TIMEX3 tags”. <http://www.timeml.org/site/tarsqi/modules/gutime/index.html>. Abril, 2010.
- Jin, P.; Lian, J.; Zhao, X. e Wan, S. (2008). “Tise: A temporal search engine for web contents”. In *IITA '08*, pages 220–224, Washington, USA. IEEE Computer Society.
- Lucene. <http://lucene.apache.org>. Abril, 2010.
- Manica, E.; Dorneles, C. F. e Galante, R. (2010). “Supporting temporal queries on xml keyword search engines”. In *SBBD '10*, Belo Horizonte, MG, Brasil. SBC.
- Page, L. et al. (1998). “The pagerank citation ranking: Bringing order to the web”. Technical report, Stanford Digital Library Technologies Project.
- Pasca, M. (2008). “Towards temporal web search”. In *SAC*, pages 1117–1121. ACM.
- TimeML. “Markup Language for Temporal and Event Expressions”. <http://www.timeml.org>. Abril, 2010.
- TreeTagger. “A Language Independent Part-of-Speech Tagger”. <http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/>. Abril, 2010.
- Vicente-Díez, M. T. e Martínez, P. (2009). “Temporal semantics extraction for improving web search”. In *DEXA Workshops*, pages 69–73. IEEE Computer Society.
- Wikipedia. “Wikipedia, the free encyclopedia”. <http://en.wikipedia.org>. Abril, 2010.