Temporal Search in Web Archives

Klaus Berberich
(kberberi@mpi-inf.mpg.de)

Promotionskolloquium / PhD Defense
July 19th, 2010
Web Archives are highly valuable...

Web archives have grown in volume and number thanks to improved digitization techniques and the awareness that born-digital contents are worth preserving.

- [timesonline](http://archive.timesonline.co.uk)
  - All issues since 1785 digitized

- [Wayback Machine](http://archive.org/web)
  - 150B web pages captured since 1996

**Characteristics:**

- Large scale (>> $10^6$ documents)
- Long time span (>> 10 years)
son of Henry VIII, the promise was visible, but it was blighted too early. Yet Edward VI, though he was not yet 17 when he died, left his mark on the history of his country during the Reformation, and the controversies in the Church of England during his short reign are still related with those of the present day.

The name which the King has assumed thus takes us back step by step to the most critical periods in the development of the English nation, to the Reformation, to the Wars of the Roses, to the "Hundred Years' War" with France, to the ending of the Saxon Monarchy, and to the early conflicts with the Danes. Those who value the historical continuity of a nation will rejoice that King Edward VII will thus take a place in our annals which will bring him into relation with the whole stream of national development. It is well to be in sympathy with modern movements and, at the same time, to remember where the roots of those movements are implanted. The King is a man of the world, of wide experience and matured sagacity. His reign has, beyond doubt, begun auspiciously, so far as this can be said while a heavy cloud of sorrow hangs over the Royal Family and the nation. The attitude of the people has been respectful and subdued, but it is clear that only the restraining forces of the melancholy situation however earnest, can do but little to assuage the first keen pangs of suffering caused by so great a loss. The genealogical table we give this morning may help to remind our readers how truly patriarchal the Queen's family relations had become. She has had eighty-six descendants, and of these no less than seventy-four, of whom thirty-seven are great-grandchildren, survive her. She was the head of a kind of Royal clan whose members are scattered all over Europe and from which in the natural course of events the future rulers of Germany, Russia, Greece, and Rumania will be drawn. Her Majesty loved her children and her children's children dearly, and the withdrawal from this world of her affection is a bereavement to them which time alone can soften.

There can hardly be much doubt as to the place in which the Queen will be buried. There are reasons, and strong reasons, which might make us wish to see the greatest of English Sovereigns rest with some of the most illustrious of her predecessors in the most venerable of English churches. But the sentiment of the whole nation will be unanimous that those reasons are overborne by one decisive consideration. The Queen has been the model of a loving wife and widow, and there can be no question that she herself would have chosen to sleep her last sleep beside her beloved.
Web Archives are highly valuable...

- Web archives have grown in volume and number thanks to improved digitization techniques and the awareness that born-digital contents are worth preserving.

http://archive.timesonline.co.uk
All issues since 1785 digitized

http://archive.org/web
150B web pages captured since 1996

Characteristics:
- large scale (>> $10^6$ documents)
- long time span (>> 10 years)
Temporal Search in Web Archives (Klaus Berberich)

Web Archives are highly valuable…

Web archives have grown in volume and number thanks to improved digitization techniques and the awareness that born-digital contents are worth preserving.

Characteristics:

Large scale (>> 10^6 documents)
Long game span (>> 10 years)

[www.archive.plmesonline.co.uk](http://archive.plmesonline.co.uk) All issues since 1785 digizd

[www.archive.org/web](http://archive.org/web) 150B web pages captured since 1996

Welcome to the Max-Planck-Institut für Informatik

Im Stadtwald, Gebäude 46.1, D-66123 Saarbrücken, Germany
Phone: +49 681 9325-0, Fax: +49 681 9325-999

- Scientific Working Groups
  - Algorithms and Complexity Group - AG 1, Kurt Mehlhorn
    - Offers and Vacancies
  - Programming Logics Group - AG 2, Harald Ganzinger
- Central Services Groups
- Publications including Research Reports, Diploma Theses, Phd Theses
- Software Projects
- Library
- Bibliographic Information
- Activities including presentations, conferences, workshops, foundations, seminars
- Miscellaneous including the photo gallery and web server statistics
- Phone and Room Directory
- How to reach MPII
- Intranet (access restricted)

Links to Max-Planck-Gesellschaft zur Förderung der Wissenschaften e. V.
Universität des Saarlandes-PhilFak - FB14 | DFKI | IBFI

www page design by Uwe Brahm <webmaster@mpi-sb.mpg.de>, 20 february 1997
Web Archives are highly valuable...

- Web archives have grown in volume and number thanks to improved digitization techniques and the awareness that born-digital contents are worth preserving.

http://archive.timesonline.co.uk
All issues since 1785 digitized

http://archive.org/web
150B web pages captured since 1996

Characteristics:
- large scale (>> $10^6$ documents)
- long time span (>> 10 years)
...but still hard to search!

Efficient and effective tools to search web archives are missing:
- Internet Archive offers only look-up by URL but no full-text search over its contents
- Times Archive offers full-text search over its contents but mostly ignores their temporal dimension

My dissertation addresses three important problems toward more efficient and effective search in web archives
Time-Travel Text Search

- Text search ignores the temporal dimension inherent to web archives and fails for information needs like

  Contemporary commentaries about the 2003 invasion of Iraq
  Early opinions about the movie “The Big Lebowski”

- **Idea:** Enrich full-text search with time-travel functionality so that queries can be evaluated “as of” a specified time

  - iraq invasion commentary @ [2003/03/01, 2003/03/31]
  - “big lebowski” review @ 1998/03/08

- **Challenges:** Large scale of web archives, exploiting their high level of redundancy, efficiency of approach
Terminology Evolution

- **Terminology**, as language in general, evolves constantly!
- Users formulate queries using today’s terminology and thus don’t find **old but highly-relevant documents**

**Idea:** Reformulate queries automatically to retrieve old but highly-relevant documents

**Challenges:** Large scale of web archives, no additional information (e.g., click logs), interactive response times
Temporal Information Needs

- Existing retrieval models don’t perform well for information needs that have a temporal dimension

FIFA World Cup tournaments of the 1990’s
Crusades of the 12th century

- Idea: Leverage temporal expressions (e.g., in 1998) contained in documents and queries

- Challenges: Meaning of temporal expressions uncertain, not clear how to seamlessly integrate them into a retrieval model

In the last decade the FIFA World Cup was won by Germany (in 1990), Brazil (in 1994), and France (in 1998).

fifa world cup 1990’s
Contributions

- Efficient Approach to Time-Travel Text Search
  - Time-Travel Inverted Index (TTIX) as a versatile framework
  - Temporal coalescing techniques to keep the index compact
  - Partitioning strategies to fine-tune the index

- Across-Time Query Reformulation Technique
  - based on a Hidden-Markov Model (HMM)
  - leverages time-dependent term co-occurrence statistics

- Retrieval Models for Temporal Information Needs
  - based on a language modeling approach to IR
  - seamlessly integrate temporal expressions
Publications

- Efficient Time-Travel Text Search on Versioned Text Collections, BTW ’07 (joint work with S. Bedathur and G. Weikum)
- A Time Machine for Text Search, SIGIR ’07 (joint work with S. Bedathur, T. Neumann, and G. Weikum)
- FluxCapacitor: Efficient Time-Travel Text Search, VLDB ’07 (joint work with S. Bedathur, T. Neumann, and G. Weikum)
- Tunable Word-Level Index Compression for Versioned Corpora, EIIR ’08 (joint work with S. Bedathur and G. Weikum)
- Bridging the Terminology Gap in Web Archive Search, WebDB ’09 (joint work with S. Bedathur, M. Sozio, and G. Weikum)
- Time Will Tell: Leveraging Temporal Expressions in IR, WSDM ’09 (joint work with I. Arikan and S. Bedathur)
- A Language Modeling Approach for Temporal Information Needs, ECIR ’10 (joint work with S. Bedathur, O. Alonso, and G. Weikum)
Outline

- Motivation
- Efficient Time-Travel Text Search in Web Archives
- Retrieval Models for Temporal Information Needs
- Conclusion
Motivation

Information needs that are best satisfied by documents that existed at a specified time

Contemporary commentaries about the 2003 invasion of Iraq
Early opinions about the movie “The Big Lebowski”

Idea: Enrich full-text search with time-travel functionality so that queries can be evaluated “as of” a specified time

Iraq invasion commentary @ [2003/03/01, 2003/03/31]
“big lebowski” review @ 1998/03/08

Objective: Support for Boolean, keyword, and phrase queries enriched with time points or time intervals
Collection Model

- Discrete notion of time (i.e., integers as timestamps)
- Special timestamp **now** always points to the current time
- Document \( d \) is a sequence of time-stamped versions
  \[
  d = \langle d^{t_1}, d^{t_2}, \ldots \rangle
  \]
- Document deletion results in tombstone version \( \perp \)
- Valid-time interval of document version \( d^{t_i} \)
  \[
  \text{val}(d^{t_i}) = \left\{ \begin{array}{l}
  [t_i, t_{i+1}) : d^{t_{i+1}} \in d \\
  [t_i, \text{now}) : \text{otherwise}
  \end{array} \right.
  \]
- State of the document collection \( D \) during time interval
  \[
  D^{[t_b, t_e]} = \bigcup_{d \in D} \left\{ d^{t_i} \in d \mid \wedge \left[ t_b, t_e \rceil \cap \text{val}(d^{t_i}) \neq \emptyset \right. \right. \right. \right. \}
  \]
Query & Retrieval Model

- Time-travel query $q^{[t_b, t_e]}$ consists of
  - query $q$ (e.g., Boolean query, keyword query, or phrase query)
  - time interval $[t_b, t_e]$ or time point $t$ (i.e., $t = t_b = t_e$)

- Time-travel query $q^{[t_b, t_e]}$ evaluated on $D^{[t_b, t_e]}$, i.e., considering only document versions alive at any time during the query time-interval $[t_b, t_e]$

- Okapi BM25, as a state-of-the-art retrieval model, adapted for time-travel keyword queries by
  - using collection statistics computed on $D^{[t_b, t_e]}$
  - assuming a stationary average document length $\text{avdl}$
Inverted Index – IR’s workhorse

- **Inverted (File) Index:**
  - Deployed in many of today’s information retrieval systems
  - Easy to understand and implement
  - Wealth of available extensions and optimizations (e.g., compression)

- **Lexicon** keeps pointers to posting lists, often implemented using a hash table or B⁺-Tree

- **Posting lists** record information about term occurrences in documents (e.g., term frequency or positions)
Time-Travel Inverted Index (TTIX)

- **Idea:** Extend inverted index in a transparent manner to keep existing extensions applicable

- **Postings** enriched with valid-time intervals

- **Temporal partitioning** allows multiple posting lists per term with postings judiciously replicated across them

- **Posting payloads** arbitrary (e.g., scalar or positional)

- **Posting-list order** arbitrary but consistent for the whole index
Temporal Coalescing for Scalar Payloads

- **Scalar payloads** represent, e.g., term-frequency information as needed for time-travel keyword queries.

- **Observation:** Changes between document versions
  - are often *minor* (e.g., corrected typos)
  - have *little effect* on scalar payloads (e.g., 41 vs. 42 x dog)
  - and thus *little impact* on query results

- **Idea:** Coalesce sequences of postings that belong to the same document and have almost-identical scores.
Temporal Coalescing for Scalar Payloads

- **Scalar payloads** represent, e.g., term-frequency information as needed for time-travel keyword queries.

- **Observation:** Changes between document versions
  - are often **minor** (e.g., corrected typos)
  - have **little effect** on scalar payloads (e.g., 41 vs. 42 x dog)
  - and thus **little impact** on query results

- **Idea:** Coalesce sequences of postings that belong to the same document and have **almost-identical scores**
Temporal Coalescing for Scalar Payloads

- **Scalar payloads** represent, e.g., term-frequency information as needed for time-travel keyword queries.
- **Observation:** Changes between document versions
  - are often **minor** (e.g., corrected typos)
  - have **little effect** on scalar payloads (e.g., 41 vs. 42 x dog)
  - and thus **little impact** on query results
- **Idea:** Coalesce sequences of postings that belong to the same document and have **almost-identical scores**

![Diagram](image-url)
Temporal Coalescing for Scalar Payloads

**Problem Statement:** Given sequence \( I \) of postings for term \( v \) in document \( d \), determine minimal-length output sequence \( O \) that keeps relative approximation error below threshold \( \varepsilon \)

\[
\forall i : \left| p_i - p' \right| / |p_i| \leq \varepsilon
\]

Optimal output sequence \( O \) can be determined using one-pass greedy algorithm that has time complexity in \( O(|I|) \)

- **Non-coalesced**
- **Coalesced**
- **Bounds**

```
\begin{align*}
\text{score} & \quad p_2 \\
\text{score} & \quad p' \\
\text{score} & \quad p_1 \\
\text{score} & \quad p_3 \\
\end{align*}
```
Temporal Coalescing for Scalar Payloads

**Problem Statement:** Given sequence $I$ of postings for term $v$ in document $d$, determine minimal-length output sequence $O$ that keeps relative approximation error below threshold $\varepsilon$

$$\forall i : \frac{|p_i - p'|}{|p_i|} \leq \varepsilon$$

Optimal output sequence $O$ can be determined using one-pass greedy algorithm that has time complexity in $O(|I|)$
Temporal Coalescing for Scalar Payloads

**Problem Statement:** Given sequence \( I \) of postings for term \( v \) in document \( d \), determine minimal-length output sequence \( O \) that keeps relative approximation error below threshold \( \epsilon \)

\[
\forall i : \left| p_i - p' \right| / |p_i| \leq \epsilon
\]

Optimal output sequence \( O \) can be determined using one-pass greedy algorithm that has time complexity in \( O(|I|) \)

score

- non-coalesced
- coalesced
- bounds

time
**Problem Statement:** Given sequence $I$ of postings for term $v$ in document $d$, determine minimal-length output sequence $O$ that keeps relative approximation error below threshold $\epsilon$

\[
\forall i : \left| p_i - p' \right| / \left| p_i \right| \leq \epsilon
\]

Optimal output sequence $O$ can be determined using one-pass greedy algorithm that has time complexity in $O(\left| I \right|)$.
Temporal Coalescing for Scalar Payloads

**Problem Statement:** Given sequence $I$ of postings for term $v$ in document $d$, determine minimal-length output sequence $O$ that keeps relative approximation error below threshold $\varepsilon$

\[ \forall i : \frac{|p_i - p'|}{|p_i|} \leq \varepsilon \]

Optimal output sequence $O$ can be determined using one-pass greedy algorithm that has time complexity in $O(|I|)$
Problem Statement: Given sequence $I$ of postings for term $v$ in document $d$, determine minimal-length output sequence $O$ that keeps relative approximation error below threshold $\varepsilon$.

$$\forall i : \frac{|p_i - p'|}{|p_i|} \leq \varepsilon$$

Optimal output sequence $O$ can be determined using one-pass greedy algorithm that has time complexity in $O(|I|)$.
Temporal Coalescing for Scalar Payloads

**Problem Statement:** Given sequence \( I \) of postings for term \( v \) in document \( d \), determine *minimal-length* output sequence \( O \) that keeps relative approximation error below threshold \( \epsilon \)

\[
\forall i : \frac{|p_i - p'|}{|p_i|} \leq \epsilon
\]

Optimal output sequence \( O \) can be determined using one-pass greedy algorithm that has time complexity in \( O(|I|) \).
Temporal Coalescing for Scalar Payloads

**Problem Statement:** Given sequence $I$ of postings for term $v$ in document $d$, determine minimal-length output sequence $O$ that keeps relative approximation error below threshold $\varepsilon$.

\[
\forall i : \left| p_i - p' \right| / |p_i| \leq \varepsilon
\]

Optimal output sequence $O$ can be determined using one-pass greedy algorithm that has time complexity in $O(|I|)$.
Temporal Coalescing for Scalar Payloads

Problem Statement: Given sequence $I$ of postings for term $v$ in document $d$, determine minimal-length output sequence $O$ that keeps relative approximation error below threshold $\varepsilon$

$$\forall i : |p_i - p'| / |p_i| \leq \varepsilon$$

Optimal output sequence $O$ can be determined using one-pass greedy algorithm that has time complexity in $O(|I|)$
Temporal Coalescing for Scalar Payloads

**Problem Statement:** Given sequence $I$ of postings for term $v$ in document $d$, determine minimal-length output sequence $O$ that keeps relative approximation error below threshold $\epsilon$.

\[ \forall i : \frac{|p_i - p'|}{|p_i|} \leq \epsilon \]

Optimal output sequence $O$ can be determined using one-pass greedy algorithm that has time complexity in $O(|I|)$.
**Problem Statement:** Given sequence I of postings for term v in document d, determine minimal-length output sequence O that keeps relative approximation error below threshold $\varepsilon$.

\[ \forall i : \frac{|p_i - p'|}{|p_i|} \leq \varepsilon \]

Optimal output sequence O can be determined using one-pass greedy algorithm that has time complexity in $O(|I|)$.
Problem Statement: Given sequence $I$ of postings for term $v$ in document $d$, determine minimal-length output sequence $O$ that keeps relative approximation error below threshold $\varepsilon$.

$$\forall i : \frac{|p_i - p'|}{|p_i|} \leq \varepsilon$$

Optimal output sequence $O$ can be determined using one-pass greedy algorithm that has time complexity in $O(|I|)$. 
Temporal Coalescing for Scalar Payloads

**Problem Statement:** Given sequence I of postings for term v in document d, determine minimal-length output sequence O that keeps relative approximation error below threshold ε.

∀ \( i \): \( \frac{|p_i - p'|}{|p_i|} \leq \varepsilon \)

Optimal output sequence O can be determined using one-pass greedy algorithm that has time complexity in \( O(|I|) \).
Temporal Coalescing for Scalar Payloads

**Problem Statement:** Given sequence $I$ of postings for term $v$ in document $d$, determine minimal-length output sequence $O$ that keeps relative approximation error below threshold $\varepsilon$.

$$\forall i : \frac{|p_i - p'|}{|p_i|} \leq \varepsilon$$

Optimal output sequence $O$ can be determined using one-pass greedy algorithm that has time complexity in $O(|I|)$. 

---

Temporal Search in Web Archives (Klaus Berberich)
Problem Statement: Given sequence I of postings for term v in document d, determine minimal-length output sequence O that keeps relative approximation error below threshold $\varepsilon$.

\[ \forall i : \frac{|p_i - p'|}{|p_i|} \leq \varepsilon \]

Optimal output sequence O can be determined using one-pass greedy algorithm that has time complexity in $O(|I|)$. 
Temporal Coalescing for Scalar Payloads

**Problem Statement:** Given sequence $I$ of postings for term $v$ in document $d$, determine minimal-length output sequence $O$ that keeps relative approximation error below threshold $\varepsilon$

$$\forall i : \frac{|p_i - p'|}{|p_i|} \leq \varepsilon$$

Optimal output sequence $O$ can be determined using

one-pass greedy algorithm

that has time complexity in $O(\ |I|\ )$
Temporal Coalescing for Scalar Payloads

**Problem Statement:** Given sequence $I$ of postings for term $v$ in document $d$, determine minimal-length output sequence $O$ that keeps relative approximation error below threshold $\varepsilon$

\[ \forall i : \frac{|p_i - p'|}{|p_i|} \leq \varepsilon \]

Optimal output sequence $O$ can be determined using one-pass greedy algorithm that has time complexity in $O(|I|)$
Temporal Coalescing for Scalar Payloads

Problem Statement: Given sequence I of postings for term v in document d, determine minimal-length output sequence O that keeps relative approximation error below threshold ε.

∀ i : |p_i − p' | / |p_i| ≤ ε

Optimal output sequence O can be determined using one-pass greedy algorithm that has time complexity in O(|I|)
Temporal Coalescing for Scalar Payloads

**Problem Statement:** Given sequence \( I \) of postings for term \( v \) in document \( d \), determine **minimal-length** output sequence \( O \) that keeps relative approximation error below threshold \( \varepsilon \).

\[
\forall i : \frac{|p_i - p'|}{|p_i|} \leq \varepsilon
\]

Optimal output sequence \( O \) can be determined using a **one-pass greedy algorithm** that has time complexity in \( O(|I|) \).
Temporal Coalescing for Scalar Payloads

**Problem Statement:** Given sequence $I$ of postings for term $v$ in document $d$, determine minimal-length output sequence $O$ that keeps relative approximation error below threshold $\varepsilon$.

\[ \forall i : \frac{|p_i - p'|}{|p_i|} \leq \varepsilon \]

Optimal output sequence $O$ can be determined using one-pass greedy algorithm that has time complexity in $O(|I|)$.
Temporal Search in Web Archives (Klaus Berberich)

Partitioning Strategies

Space Optimal (SOpt)
- keeps one posting list $L_v$ per term $v$
- consumes minimal space
  but achieves only bad query-processing performance

Performance Optimal (POpt)
- keeps one posting list $L_v$: $[t_i, t_{i+1})$ for each elementary time interval
- achieves optimal performance for time-point queries
  but consumes a lot of space (in the worst case $O( |L_v|^2)$)
Temporal Search in Web Archives (Klaus Berberich)

Partitioning Strategies

- **Space Optimal (SOpt)**
  - keeps one posting list $L_v$ per term $v$
  - consumes minimal space
  - but achieves only bad query-processing performance

- **Performance Optimal (POpt)**
  - keeps one posting list $L_v$: $[t_i, t_{i+1})$ for each elementary time interval
  - achieves optimal performance for time-point queries
  - but consumes a lot of space (in the worst case $O(|L_v|^2)$)
Partitioning Strategies

Space Optimal (SOpt)
- keeps one posting list $L_v$ per term $v$
- consumes minimal space
  but achieves only bad query-processing performance

Performance Optimal (POpt)
- keeps one posting list $L_v: [t_i, t_{i+1})$ for each elementary time interval
- achieves optimal performance for time-point queries
  but consumes a lot of space (in the worst case $O(|L_v|^2)$)
Performance-Guarantee Partitioning Strategy (PG)

- **Idea:** Minimize total number of postings kept in the index, while guaranteeing that for any time-point query the number of postings read is at most a factor $\gamma$ worse than optimal

$$\arg\min_{\mathcal{P}_v} S(\mathcal{P}_v) \quad \text{s.t.} \quad \forall t \in [t_1, t_n) : PC(t) \leq \gamma \cdot |L_v : t|$$

- Optimal solution computable in time $O(|L_v| + n^2)$ and space $O(n^2)$ using dynamic programming over prefix subproblems $[t_1, t_k)$

- 2-approximation of an optimal solution computable in time $O(|L_v|)$ and space $O(n)$ using a greedy algorithm
Space-Bound Partitioning Strategy (SB)

**Idea:** Minimize expected number of postings read for a time-point query, while ensuring that the index contains at most $\kappa$ times the optimal number of postings

$$\arg\min_{P_v} \sum_{t \in [t_1, t_n]} P(t) \cdot PC(t) \quad \text{s.t.} \quad S(P_v) \leq \kappa \cdot |L_v|$$

- Optimal solution computable in $O(|L_v| \times n^2)$ time and $O(|L_v| \times n)$ space using dynamic programming over prefix subproblems $[t_1, t_k)$ and space bounds $s \leq \kappa \cdot |L_v|$

- Approximate solutions computed using Simulated Annealing (run for $R$ rounds) in time $O(|L_v| + n^2 + n \times R)$ and space $O(n^2)$ work well in practice
FluxCapacitor Prototype Implementation

Your query **iraq war**@**Jun 18, 2002 8:30 PM** needed 266ms and has 50 results

**Iran-Iraq War**
The b Iran Iraq War b was a border war between Iran and Iraq that took place between September 22 1980 and August 20 1988 It is also known as the b First Persian Gulf War b and the b Gulf War b

**Score:** 13,385 | **Created:** May 27, 2002 5:59 PM

- **Java** (keeping it as “low-level” as possible)
- **Lexicon** kept in **main memory**, implemented using a **hash table**
- **Posting lists** kept on **disk**, implemented using **flat files**
- **Postings** thoroughly encoded (e.g., using **gap and 7-Bit encoding**)
- **AJAX-based GUI** implemented using **Google Web Toolkit**
Experimental Evaluation

- Experiments on three document collections:
  - **New York Times Annotated Corpus (NYT)**
    1.8M newspaper articles / 1.8M versions / 1987–2007
  - **Revision History of the English Wikipedia (WIKI)**
    1.5M distinct encyclopedia articles / 15M versions / 2001–2005
  - **Weekly Crawls of U.K. Governmental Websites (UKGOV)**

- **Query workload** for each document collection consists of:
  - 150 queries from AOL query log with most clicks on a relevant domain (e.g., en.wikipedia.org or raf.mod.uk)
  - each of the queries is combined with ten time points randomly chosen within the respective collection’s lifetime
Sizes of indexes with scalar payloads built using space-optimal partitioning and temporal coalescing for scalar payloads.
Relative recall for top-10 query results on indexes with scalar payloads built using temporal coalescing for scalar payloads.
Time-point keyword query workload processed on indexes built using temporal coalescing for scalar payloads ($\varepsilon = 0.1$)
Outline

- Motivation
- Efficient Time-Travel Text Search in Web Archives
- Retrieval Models for Temporal Information Needs
- Conclusion
Motivation

- Information needs that have a **temporal dimension**
  
  *FIFA World Cup tournaments of the 1990’s*
  
  *Movies that won an Academy Award in 2007*
  
  *Crusades of the 12th century*
  
  *London Summer Olympics 2012*

- Queries that contain a **temporal expression** (e.g., *in 1998*)
  
  - indicate an underlying temporal information need
  
  - account for **1.5%** of general web queries
  
  - are **more common** for specific domains (e.g., News or Sports) and/or specific user groups (e.g., historians or journalists)

- **But**: Not well-supported by existing retrieval models
Temporal Expressions

- Temporal expressions can be categorized as:
  - **explicit** (e.g., July 19th 2010 or September 1872)
  - **implicit** (e.g., Christmas 2009 or New Year’s Eve 2000)
  - **relative** (e.g., yesterday or last month)

- Meaning of a temporal expression is often uncertain, e.g.,

  *France won the FIFA World Cup in 1998*
  *In 1998 Bill Clinton was President of the U.S.*
  *Nagano hosted the Winter Olympics in 1998*
Temporal Expression Model

- We model a temporal expression as a four-tuple

\[ T = (t_{b_l}, t_{b_u}, t_{e_l}, t_{e_u}) \]

that records the earliest/latest begin/end of any time interval \([b, e]\) that \(T\) may refer to

- The temporal expression in 1998, e.g., is represented as

\[(1998/01/01, 1998/12/31, 1998/01/01, 1998/12/31)\]
Temporal Expression Model

We model a temporal expression as a four-tuple

\[ T = ( t_{b_l}, t_{b_u}, t_{e_l}, t_{e_u} ) \]

that records the earliest/latest begin/end of any time interval \([b, e]\) that \(T\) may refer to

The temporal expression in 1998, e.g., is represented as (1998/01/01, 1998/12/31, 1998/01/01, 1998/12/31)
Temporal Expression Model

- We model a temporal expression as a four-tuple
  \[ T = (tb_l, tb_u, te_l, te_u) \]
  that records the earliest/latest begin/end of any time interval \([b, e]\) that \(T\) may refer to.

- The temporal expression in 1998, e.g., is represented as
  \((1998/01/01, 1998/12/31, 1998/01/01, 1998/12/31)\)

\(T\)
Temporal Expression Model

- We model a temporal expression as a four-tuple
  \[ T = ( t_{b_l}, t_{b_u}, t_{e_l}, t_{e_u} ) \]
  that records the earliest/latest begin/end of any time interval \([b, e]\) that \(T\) may refer to.
- The temporal expression in 1998, e.g., is represented as
  \((1998/01/01, 1998/12/31, 1998/01/01, 1998/12/31)\)

(a) \([1998/01/01, 1998/12/31]\)
Temporal Expression Model

- We model a temporal expression as a four-tuple

\[ T = (tb_l, tb_u, te_l, te_u) \]

that records the earliest/latest begin/end of any time interval \([b, e]\) that \(T\) may refer to

- The temporal expression in 1998, e.g., is represented as \((1998/01/01, 1998/12/31, 1998/01/01, 1998/12/31)\)

(a) \([1998/01/01, 1998/12/31]\)
(b) \([1998/07/12, 1998/07/12]\)
Temporal Expression Model

- We model a temporal expression as a four-tuple
  \[ T = (tb_l, tb_u, te_l, te_u) \]
  that records the earliest/latest begin/end of any time interval \([b, e]\) that \(T\) may refer to

- The temporal expression in 1998, e.g., is represented as
  \((1998/01/01, 1998/12/31, 1998/01/01, 1998/12/31)\)

\(\text{(a)} \ [1998/01/01, 1998/12/31] \)
\(\text{(b)} \ [1998/07/12, 1998/07/12] \)
\(\text{(c)} \ [1998/02/07, 1998/02/22] \)
Document & Query Model

- We distinguish between the **textual part** and the **temporal part** of documents and queries
  - \( d_{\text{text}} \) and \( q_{\text{text}} \) are bags of textual terms
  - \( d_{\text{time}} \) and \( q_{\text{time}} \) are bags of temporal expressions
Language Models in Information Retrieval

- **Probabilistic generative model** estimated for each document
- **Query-likelihood approaches** rank documents based on their probability of generating a given query
- **Unigram language model** estimates probability of generating query $q$ from document $d$ as
  \[
  P(q | d) = \prod_{v \in q} P(v | d)
  \]
- **Jelinek-Mercer smoothing** estimates the probability $P(v | d)$ of generating the term $v$ from document $d$ as
  \[
  P(v | d) = \lambda \cdot \frac{tf(v, d)}{|d|} + (1 - \lambda) \cdot \frac{tf(v, C)}{|C|}
  \]
## Language Model Framework

- **Query-likelihood approach** assuming that the textual and temporal query part are **generated independently**

\[
P(q | d) = P(q_{text} | d_{text}) \times P(q_{time} | d_{time})
\]

- **Independent generation of query temporal expressions**

\[
P(q_{time} | d_{time}) = \prod_{Q \in q_{time}} P(Q | d_{time})
\]

- **Two-step generation of temporal expression Q**
  1. **Draw a temporal expression** \( T \) **at uniform random**

\[
P(Q | d_{time}) = \frac{1}{|d_{time}|} \sum_{T \in d_{time}} P(Q | T)
\]
  2. **Generate** \( Q \) **from** \( T \)
Uncertainty-Ignorant Language Model

- The temporal expression $T$ generates only itself

$$P( Q | T) = \begin{cases} 1 & : Q = T \\ 0 & : \text{otherwise} \end{cases}$$

- Profits from extraction of temporal expressions
- Ignores temporal expressions’ inherent uncertainty

In the last decade the FIFA World Cup was won by Germany (in 1990), Brazil (in 1994), and France (in 1998)…
Uncertainty-Aware Language Model

We assume any time interval \([q_b, q_e]\) that the query temporal expression may refer to be equally likely

\[
P(\ Q \mid T ) = \frac{1}{|Q|} \sum_{[q_b, q_e] \in Q} P(\ [q_b, q_e] \mid T )
\]

The temporal expression \(T\) generates any time interval \([q_b, q_e]\) that it may refer to with equal probability

\[
P(\ [q_b, q_e] \mid T ) = \begin{cases} 
1/|T| & : \ [q_b, q_e] \in T \\
0 & : \ otherwise
\end{cases}
\]
Uncertainty-Aware Language Model

\( \text{Intuitively, } P(Q|T) \text{ reflects the probability that the user issuing the query and the author writing the document have the same time interval in mind} \)

\( \text{Our definition of } P(Q|T) \text{ can be simplified as} \)

\[
P(Q|T) = \frac{|T \cap Q|}{|T| \cdot |Q|}
\]

\( \text{treating } Q \text{ and } T \text{ as sets of time intervals} \)

\( \text{\lvert T\rvert, } \lvert Q\rvert, \text{ and } |T \cap Q| \text{ efficiently computable based on our four-tuple representation, i.e., no need to enumerate the huge but finite number of time intervals in } Q \text{ and } T \)
Uncertainty-Aware Language Model

- Profits from extraction of temporal expressions
- Considers temporal expressions’ inherent uncertainty

In the last decade the FIFA World Cup was won by Germany (in 1990), Brazil (in 1994), and France (in 1998)...

fifa world cup
1990’s
Experimental Evaluation

- Experiments on two document collections:
  - New York Times Annotated Corpus
    - 1,855,656 documents
  - Wikipedia (snapshot as of July ’09)
    - 2,955,294 documents

- Queries and binary relevance assessments collected using the crowdsourcing platform Amazon Mechanical Turk

- Query workload consists of 40 queries, e.g.:
  - boston red sox [october 27, 2004], pink floyd [march 1973], wright brothers [1905], siemens [19th century], babe ruth [1921]

- Methods under comparison:
  - LM  
    Unigram Language Model
  - LMT  
    Uncertainty-Ignorant Language Model
  - LMTU  
    Uncertainty-Aware Language Model
Retrieval Effectiveness

New York Times

<table>
<thead>
<tr>
<th></th>
<th>LM</th>
<th>LMT</th>
<th>LMTU</th>
</tr>
</thead>
<tbody>
<tr>
<td>P@10</td>
<td>0.37</td>
<td>0.32</td>
<td>0.51</td>
</tr>
<tr>
<td>nDCG@10</td>
<td>0.38</td>
<td>0.33</td>
<td>0.49</td>
</tr>
</tbody>
</table>
Outline

- Motivation
- Efficient Time-Travel Text Search in Web Archives
- Retrieval Models for Temporal Information Needs
- Conclusion
Conclusion

Web Archives
- play an important role in preserving our cultural heritage
- pose interesting research problems

Efficient Approach to Time-Travel Text Search
- Time-Travel Inverted Index (TTIX) as a versatile framework
- Temporal coalescing keeps index compact and results accurate
- Partitioning strategies allow to fine-tune the index

Retrieval Models for Temporal Information Needs
- based on a language modeling approach to IR
- seamlessly integrate temporal expressions
- improve retrieval effectiveness
Thanks!
Questions?