CSE636: Data Integration

Information Retrieval Indexes for XML Documents

Anmol Bhasin   Mohit Devnani
abhasin[at]cedar.buffalo.edu   mdevnani[at]cse.buffalo.edu

Spring 2005
Overview

- Information VS Data Retrieval
- Foundations of Information Retrieval
- Vector Space Model
- RSS / Atom
- Berkeley DB XML
- Objectives
- System Architecture
- Features & Query Support
- Implementation
- Future Work
Information VS Data Retrieval

- Information Retrieval
  - Typically carried out on Unstructured Data
  - Concerned with Retrieving Information about a subject
  - Content of Information FUZZY
    (Semantically Ambiguous)
  - Degree of Relevance critical
  - Absence of Formal Query Language

- Data Retrieval
  - retrieving objects satisfying Suitable for well formulated Structured Data
  - Aims at clearly defined conditions
    (Region Algebra/Regex)
  - Relevance not important as ‘everything’ is relevant
  - Formal Query Language Specifications
Foundations of Information Retrieval
Vector Space Model

- Documents are represented as ‘vectors’ in term space
  - Terms are usually ‘stems’
  - Documents represented by ‘weighted’ vectors of terms

- Queries are also modeled in term space as boolean / weighted vectors
Vector Space Model cont...

**TFIDF**

Term Frequency * Inverse Document Frequency

\[ w_{ij} = tf(t_i, d_j) \cdot idf(t_i) = f_{i,j} \cdot \log\left(\frac{N}{n_i}\right) \]

\[ f_{i,j} = \frac{freq_{i,j}}{\max(freq_{i,j})} \]

**Cosine Similarity**

Document to Query Similarity metric

\[ sim(d_j, q) = \frac{\vec{d}_j \cdot \vec{q}}{|\vec{d}_j| \times |\vec{q}|} \]

\[ sim(d_j, q) = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{j=1}^{t} w_{i,q}^2}} \]
RSS and Atom

- **RSS**: *Really Simple Syndication*
- RSS is a dialect of XML / XML based syndication specification
- RSS files conform to the XML 1.0 **specification**, as published by W3C
- RSS standards .91, .92, 1.0, 2.0

**Sample RSS Document**

**Experimental RSS Schema** (Jorgin Thelin)

- Atom – another form of XML based syndication
Native XML Database Engine

Embedded XML Database linked to Application

Layered on top of the Berkeley DB database (a key-value pair based database)

Stores XML documents in collections and provides ability to access multiple collections at the same time.

Recently started to support XQuery, XPath, and XML Namespaces
Objectives

- Proof of concept for XML IR using traditional IR techniques
- Project Objectives
  - Platform for indexing and integration of RSS news feeds from multiple sources
  - Provide support for keyword searches and focused queries on the index
  - Semantically cluster news feeds based on XML feed data
System Architecture

WEB
RSS Feeds

Feed Aggregation, Cleaning

Download
RSS Feeds

Clean
(Resolve Entity References)

Parse
(Extract title, descr. etc.)

Pre-process
(Eliminate stopword, stem)

XML IR
Index Generator

Parsing, Pre-Processing, Index Generation

Container Querying Framework

RSS Feeds
Container

XML IR
Index

XML IR
Index
Generator

Querying
Client
Interface

Feed Clustering Framework

Berkeley DB XML
System Architecture

- Feed Aggregator
- Data Cleaner
  - XML Encoding
  - Date Formatting
  - Filter non-interest entities
- Data Preprocessor
  - Stop Word Removal
  - Word Stemmer (Porter Stemmer)
- IR Indices Generation
- Clustering Framework for Clustering Item Feeds
  - K – Means Implementation
  - Cosine Similarity as Distance Metric
- Index & Document Container (Berkeley DB XML)
  - XML All IR Indices are themselves Documents
- Query Framework (Keyword Searches & Focused Top 5 Queries)
Keyword based searching of news feed data
e.g. ‘President of Palestine’

Daily news item clustering into ‘Top Five Stories’
using K-means clustering

Popular Story Search using Google API as well as Corpus Statistic
System Implementation

IR INDICES

- **Document Dictionary**
  
  ```xml
  <?xml version="1.0" encoding="ISO-8859-1"?>
  <DocDictionary>
    <Document>
      <ID>0</ID>
      <LINK>http://www.abz.com/permalinker.html</LINK>
    </Document>
  </DocDictionary>
  ```

- **Term Dictionary**
  
  ```xml
  <?xml version="1.0" encoding="ISO-8859-1"?>
  <!-- Term Dictionary-->
  <TermDict>
    <Term>
      <ID>0</ID>
      <String>azb</String>
    </Term>
  </TermDict>
  ```

<table>
<thead>
<tr>
<th>DocID</th>
<th>permaLink</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><a href="http://www.msn">http://www.msn</a>....</td>
</tr>
<tr>
<td>2</td>
<td><a href="http://www.bbc">http://www.bbc</a>....</td>
</tr>
<tr>
<td></td>
<td>n</td>
</tr>
<tr>
<td></td>
<td><a href="http://www.cnn">http://www.cnn</a>....</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TermID</th>
<th>Term</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>movie</td>
</tr>
<tr>
<td>2</td>
<td>actor</td>
</tr>
<tr>
<td></td>
<td>n</td>
</tr>
<tr>
<td></td>
<td>director</td>
</tr>
</tbody>
</table>
System Implementation cont...

IR INDICES

- Forward Map

<table>
<thead>
<tr>
<th>DocID</th>
<th>Pointer</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>n</td>
<td></td>
</tr>
</tbody>
</table>

TermID | Freq
-------|-----
1      | 3   
2      | 2   

TermID | Freq
-------|-----
1      | 6   
2      | 8   
n      | 3   

.xml

<html>
<body>
<table>
  <tr><td><ForwardMap></td></tr>
  <tr><td><Posting></td></tr>
  <tr><td><DID>9</DID></td></tr>
  <tr><td><Term></td></tr>
  <tr><td><TID>5</TID><Freq>3</Freq></td></tr>
  <tr><td></td></tr>
</table>
</body>
</html>
System Implementation cont...

**IR INDICES**

- Inverted Map

```
<InvertedMap>
  <Posting>
    <TID>2</TID>
    <Document>
      <DID>3</DID><FREQ>3</FREQ>
    </Document>
  </Posting>
  ...
  <Posting>
    <TID>n</TID>
    <Document>
      <DID>n</DID><FREQ>4</FREQ>
    </Document>
  </Posting>
</InvertedMap>
```

<table>
<thead>
<tr>
<th>TermID</th>
<th>Pointer</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>n</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DocID</th>
<th>Freq</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>n</td>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DocID</th>
<th>Freq</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>n</td>
<td>5</td>
</tr>
</tbody>
</table>
NEWS CLUSTERS

K Means Clustering Basics …

An algorithm for partitioning (or clustering) \( N \) data points into \( K \) disjoint subsets \( S_j \) containing \( N_j \) data points so as to minimize the sum-of-squares criterion

\[
J = \sum_{j=1}^{K} \sum_{n \in S_j} (x_n - \mu_j)^2
\]

where \( x_n \) is a vector representing the \( n \)th data point and \( \mu_j \) is the geometric centroid of the data points in \( S_j \).
NEWS CLUSTERS

K Means Implementation Specification

- K = 5 : Top 5 Stories per day
- Feature Selection : Posting Files of a Document
- Distance Metric : Cosine Similarity On Title & Description Text
- Data Set : RSS Feeds for a particular day
- Criterion Function : Least Mean Squares
System Implementation cont...

Query Framework ...

Query [keywords]

Query [Term IDs]

Query [Term / Doc IDs]

Posting Files

Term Dictionary

Inverted Map

Forward Map

Calc. Cosine Similarity

Results [Ranked List of Document IDs]

Dept. of CSE, Univ. at Buffalo

CSE 636 Information Retrieval Indexes for XML Search
Conclusions & Future Work

- Data should be conducive to Information Retrieval
- Custom parsers required for different schemas
- Adding Precision & Recall Metrics to measure Retrieval Performance
- Hierarchical clustering in place of K Means
- Client / Server based implementation
References

5. Mihajlovic V., et. al., *XML-IR DB Sandwich*
6. Thelin J.,
   [www.thearchitect.co.uk/weblog/archives/2003/03/00118.html](http://www.thearchitect.co.uk/weblog/archives/2003/03/00118.html)
Thank you.