RDFVault: A Compact In-Memory Dictionary For RDF data

Hamid R. Bazoobandi, Steven de Rooij, Jacopo Urbani, Annette ten Teije, Frank van Harmelen, and Henri Ba

Vrije Universiteit Amsterdam
University Van Amsterdam

3rd June, ESWC 2015
Dictionary Encoding is designed to maximize the data compaction, the dictionary itself is not compact.
D of HDT
Dictionary Encoding

- For static data, encoding/decoding can be done at pre/post processing phases

- For dynamic/streaming data, applications have to maintain the dictionary in memory
The Goal

• A compact dictionary encoder for in-memory that supports frequent updates
Method

- Minimize the overhead of data structure
- Minimize the storage of common prefixes
Common Prefixes Memory Consumption

- Freebase
  - Literal: 80.10%
  - IRI: 87.00%
- IRI
  - All: 84.30%
- BTC 2014
  - Literal: 36.40%
  - IRI: 81.90%
- DBpedia (EN)
  - Literal: 15.40%
  - IRI: 63.60%
- Bioportal
  - Literal: 26.10%
  - IRI: 85.90%

Common Prefixes Memory Saving
Trie Properties

- Keys are implicitly stored in the body of trie
- Keys can be reconstructed with $O(L)$ complexity where $O(L)$ is the length of string.
Trie Example

ABC, ADC

Extremely BAD memory efficiency
Compact Trie

Path Compression

Lazy Expansion
Compact Trie Memory Efficiency

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Used Pointers</th>
</tr>
</thead>
<tbody>
<tr>
<td>bioportal</td>
<td>1.58%</td>
</tr>
<tr>
<td>dbpedia (EN)</td>
<td>1.19%</td>
</tr>
<tr>
<td>freebase</td>
<td>1.91%</td>
</tr>
<tr>
<td>btc 2014</td>
<td>1.23%</td>
</tr>
</tbody>
</table>
Adaptive Radix Tree (ART)
Adaptive Radix tree Memory Efficiency

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Used Pointers</th>
</tr>
</thead>
<tbody>
<tr>
<td>bioportal</td>
<td>47.90%</td>
</tr>
<tr>
<td>dbpedia (EN)</td>
<td>46.60%</td>
</tr>
<tr>
<td>freebase</td>
<td>48.03%</td>
</tr>
<tr>
<td>btc 2014</td>
<td>44.79%</td>
</tr>
</tbody>
</table>
Burst Trie & HAT

ABC, ADC
List Trie

- Very memory efficient
- Very slow for generic data

NO ONE USES THIS
Children distribution

![Bar chart showing children distribution by dataset: bioportal, freebase, btc2014, dbpedia. The bar chart shows the percentage of children in each age group (>= 12, >= 4 & <= 12, < 4) across the datasets.](image)
• For majority of nodes, the cost of using linked lists is not significant

• RDF data is highly skewed, therefore some paths in the trie are traversed much more frequently
Trie used in RDFVault

Compact List Trie, with Move-to-front Policy
Preserving Pointers

(a) ABC

(b) "ABC"
Memory Efficiency

- **freebase**
  - Dictionary
  - RDFVault

- **btc 2014**
  - Dictionary
  - RDFVault

- **dbpedia (EN)**
  - Dictionary
  - RDFVault

- **bioportal**
  - Dictionary
  - RDFVault

Data Structure

Strings

Memory Usage (MB)
Memory Efficiency

![Memory Efficiency](image_url)

Chart 1: Memory usage for different datasets and versions of RDFVault and Dictionary, including bioportal, dbpedia (EN), btc 2014, and freebase.

Chart 2: Memory usage as (IRIs) and (Literals) for DS and STR.
Performance

(Encode)

(Decode)
Move-to-front Effectiveness

Performance (ns)

- Enabled
- Disabled
- bioportal
- dbpedia (EN)
- btc 2014
- freebase

- IRI
- Literal
- All
Conclusion

https://github.com/bazoohr/RDFVault.git

Art is never finished, only abandoned.  
- Leonardo da Vinci