XML Compression

Sebastian Graf

SS 06 Seminar „XML & Databases“
University of Konstanz

Outline

1. Compression and XML
2. XMill
3. XMLPPM
4. Conclusion

Possibilities

Compression:
- just the text as it is
  (gzip, bzip2, ppm)
- taking the XML structure into account
  (XMill, XMLPPM)

XML Compression

Outline

1. Compression and XML
2. XMill
3. XMLPPM
4. Conclusion

Possibilities

Compression:
- just the text as it is
  (gzip, bzip2, ppm)
- taking the XML structure into account
  (XMill, XMLPPM)

Outline

1. Compression and XML
2. XMill
3. XMLPPM
4. Conclusion

Possibilities

Compression:
- just the text as it is
  (gzip, bzip2, ppm)
- taking the XML structure into account
  (XMill, XMLPPM)

Outline

1. Compression and XML
2. XMill
3. XMLPPM
4. Conclusion

Possibilities

Compression:
- just the text as it is
  (gzip, bzip2, ppm)
- taking the XML structure into account
  (XMill, XMLPPM)

Outline

1. Compression and XML
2. XMill
3. XMLPPM
4. Conclusion

Possibilities

Compression:
- just the text as it is
  (gzip, bzip2, ppm)
- taking the XML structure into account
  (XMill, XMLPPM)

Outline

1. Compression and XML
2. XMill
3. XMLPPM
4. Conclusion

Possibilities

Compression:
- just the text as it is
  (gzip, bzip2, ppm)
- taking the XML structure into account
  (XMill, XMLPPM)
**Arithmetic encoding:**
Input: "banan"
Output: 0.45654

Input: text & interval ([0,1] for example) & model for encoding (probability of each char in input string for example).
Output: float to represent start point of decoding.

**PPM**
Input: "bananas"
Order 0: order 1: order 2:

"Prediction by Partial Matching" PPM is an encoding method for arithmetic encoding. Target is to adapt the probability for every char to a context of a given length. PPM needs no extra model like a probabilistic table to build the table. This table is build incrementally by the algorithm itself. The model of the last encoded/decoded char encodes/decodes the next one. The single chars are encoded through a shared alphabet (Build of tables similar to “Lempel Ziv Welch” compression).

**XML Compression**
Different approaches shown:
- XMLPPM (MHM)
- SCMPPM / HCMPPM
- XMll

XMLPPM: builds one PPM model for each syntactic class --> MHM = multiplexed hierarchical modelling
SCMPPM (= structural context modelling): builds one PPM model for the content under each element symbol
HCMPPM (= hybrid context modelling): First like MHM, later on like SCM
XMll: splits XML file into different containers and compresses each of them using a specific compressor

**Compression of a XML Document means:**
1. Separate structure from data
2. Group data items with related meaning
3. Apply different compressors to different containers

1. Structure consists of tags and attributes while data consists of element contents and attribute values. These are compressed separately.
2. Data is grouped in containers. Each container is compressed separately.
3. Some items are text, others are numbers. Applying of different specialized compressors to different containers (semantic compressors).
- atomic compressors for integers, text, enumerations
- combined compressors: combine atomic compressors with constants
- user-defined compressors: user declares compressors for specific uses e.g. dna sequences

User can specify the data to store it best.
E.g. IP-Address: store it as integer with \( \cdot \) as delimiter instead of a string.
XMLPPM

- Encoding of element names through single bytes (ESAX)

- Compression like PPM but with modified model (multiplexed hierarchical model):
  - take tree structure into account
  - insert bytes to preserve tree structure through encoding
  - take one model for different kinds of nodes

XMLPPM, an Example

Different Models for elements(elts), attributes(atts), text(chars) and symbols(syms). Each node has the byte-code of its parent. Symbols are the encoded table for the reference between byte and string at its first occurrence.

Before each byte the byte of the parent in the tree structure is inserted.

ESAX

Encoding of single tags into bytes. When a tag first occurs, it is stored just after the byte which is used for encoding.

SCMPPM / HCMPPM

SCMPPM: Model not just for different kind of nodes but for different kind of elements (e.g. book and author have different models in SCMPPM).

HCMPPM (hybrid context modelling): First use single model with context symbol injection as in MHM. In addition, track the number of symbols seen in each context. When this number exceeds a given bound m, give it its own PPM model as in SCM.
Results XMill / XMLPPM

Results XML-/SCMPPM

Conclusion

Text compression:
- easy to use
- But no satisfying result comparing to XMill / XMLPPM
XMill:
- one of the first tries to compress XML
- Only offline compression
- Better compression than text (depending on the compressor)
- Easy to use and fully implemented
XMLPPM:
- new approach using fixed compressor
- Online compression available through encoding
- Better compression than XMill
- Is going to be further developed

Finished

Thanks for your attention!

Questions, Remarks...?
[1] Hartmut Lieinke, Dan Suciu. XMill: an efficient compressor for XML Data

