FLEXIBLE AND EFFICIENT XML SEARCH WITH COMPLEX FULL-TEXT PREDICATES

EMIRAN CURTMOLA
University of California San Diego
ecurtmola@cs.ucsd.edu

SIHEM AMER-YAHIA
AT&T Research Labs
sihem@research.att.com

ALIN DEUTSCH
University of California San Diego
deutsch@cs.ucsd.edu

Motivation

• Motivation
  – in GRID scenario, many sources and web services
    • each described by annotations (XML)
    • need information discovery and ranking: querying XML structure and text
  – Problem:
    – a variety of query languages with different expressive power, semantics, and scoring methods
  – Our solution:
    – unified framework for XQFT-class languages
      • formal semantics
      • enable query optimizations
      • leverage relational query evaluation techniques
      – single solution by reducing from existing languages to this framework

Full-Text Query Example

• A query example expressed in natural language:
  find all elements containing the terms “Jefferson” and “education” within a window of 10 words, with “Jefferson” ordered before “education”

• The same query example in XQuery Full-Text:
  /*[. ftcontains “Jefferson” & “education” window 10 words ordered]

• Its XFT algebraic expression is
  \( \sigma_{\text{ordered}}(\text{Jefferson} \land \text{education}) \times \left( \sigma_{\text{window}}(10) \times \left( \sigma_{\text{Jefferson}}(\text{Jefferson}) \land \sigma_{\text{education}}(\text{education}) \right) \right) \)

XFT Algebra

• Formalization of full-text language semantics in terms of keyword pattern matches
• XFT algebra consists of operators that manipulate pattern matches
  – conjunctions, disjunctions, difference and flexible interpretations of selection predicates (they may be satisfied by different or the same term matches within an answer)
• Algebra admits
  – Efficient algorithms for operator evaluation
    • integrate efficient LCA computation with efficient match manipulation
  – Rewritings of queries into more efficient forms
    • analogous to relational algebra rewritings
  – Extension with scoring
    • works for a large family of scoring functions that generalizes the standard vector-based model
    • incremental score computation for an element from the scores of its descendants

The XFT Algorithms

• Propose two evaluation strategies: AllNodes and SCU
  – AllNodes algorithm
    – straight forward application from XFT operators
    – self-contained pattern match representation
  – SCU algorithm combine
    – stack-based techniques to cope with element nesting
    – compute the smallest number of elements and matches that are relevant to all query answers
    – distributed pattern match representation

Relational-style query processor architecture

Experiments

• Ongoing Work:
  – Integrate XFT algorithms into GalaTex
  – Efficient topK evaluation for XFT plans
  – Optimizations across structure navigation and text search
    • Query rewritings and minimization
    • Consistent scoring: same scores for equivalent expressions
    • Consistent ranking: same topK results for equivalent expressions

SCU plan evaluation

Minimize size of intermediate results by representing only the most specific nodes and by not duplicating matches across nested elements

Ongoing Work

• GalaTex is the first complete implementation of W3C XQuery Full-Text language

http://www.galaxquery.com/galatex

• Ongoing Work:
  – Integrate XFT algorithms into GalaTex
    – Efficient topK evaluation for XFT plans
  – Optimizations across structure navigation and text search
    • Query rewritings and minimization
    • Consistent scoring: same scores for equivalent expressions
    • Consistent ranking: same topK results for equivalent expressions