The Effectiveness of *GLOSS* for the Text Database Discovery Problem

Lius Gravano, Hector Garcia-Molina
Stanford University

Anthony Tomasic
Princeton University

Proceeding of ACM SIGMOD Conference, 1994

Outline

- **Introduction**
- ***GLOSS*: Glossary of Servers Server**
- **Experimental Framework**
- **Improving *GLOSS***
- **Conclusions**

Reference

The Efficacy of GLOSS for the Text Database Discovery Problem

Lius Gravano, Hector Garcia-Molina, Anthony Tomasic

Stanford University Technical Note Number STAN-CS-TN-93-2

Introduction

- **Motivation**
  - **find a scalable solution to the text database discovery problem**
  - **obvious solutions**
    - » **forwarding the queries to all known databases**
    - » **central full index for all of the documents**
- **Main idea**
  - **suggest potentially good databases to search**
    - » **present the query to server to select a set of promising databases**
    - » **evaluate the query at the chosen databases**
  - **estimate by the word-frequency information for each database**
    - » **how many documents at that database actually contain each word**

Introduction

- **Extended semantics**
  - **exhaustive search**
  - **all-best search**
  - **only-best search**
  - **sample search**
- **Example 1. find** *Subject* computer

GLOSS: Glossary of Servers Server

- **Query representation**
  - **atomic subquery is a keyword field-designation pair**
  - **only consider boolean `and` queries**
    - » **find** *Author* Knuth **and** *Subject* computer
- **Database histograms**
  - ***DBSize*(db) : the total number of documents in database db**
  - ***freq*(t, db) : the number of documents in db that contain t**
- **Estimate of the result size of a query**
  - **ESize/*Est*(q, db) => RSize(q, db)**
  - ***Chosen/Est*(q, DB) = { db in DB | ESize/*Est*(q, db) > 0 ^ ESize/*Est*(q, db) = max ESize/*Est*(q, db`), for all db` in DB }**

NTHU db-lab

GLOSS: Glossary of Servers Server

- **Estimators**
  - **ESize/*Ind*(*find* t1^..^tn, db) = [*freq*(t1,db)/*DBSize*(db)]*..* [*freq*(tn,db)/*DBSize*(db)]*DBSize(db)**
    - » **keywords appear in the different documents of a database following independent and uniform probability distributions**
  - **ESize/*Min*(*find* t1^..^tn, db) = min[*freq*(ti,db)], for i = 1..n**
  - **ESize/*Binary*(*find* t1^...^tn, db) = 0, if *freq*(ti,db) = 0 for some i**
    = 1, otherwise
  - **example 2. find** *Author* Knuth **and** *Subject* computer

NTHU db-lab

GLOSS: Glossary of Servers Server

- **Evaluation criteria**
  - **compare the prediction of the estimator against what actually is the `right subset` of DB to query**
  - ***C/ex : Relevant* <= *Chosen/est***
  - ***C/ab : Best* <= *Chosen/est***
  - ***C/ob : Chosen/est* <= *Best***
  - ***C/sm : Chosen/est* <= *Relevant***

- **Performance metrics**
  - ***Success(C,Est)* = 100 * [| {q in Q | *Chosen/est* satisfies *C*} | / |Q|]**
  - ***Alpha(C,Est)* = 100 - *Success(C,Est)***
  - ***Beta(C,Est) = Success(C,Est)* - 100 * [| {q in Q | *Chosen/est* strictly satisfies *C*} | / |Q|]**

NTHU db-lab

Experimental Framework

- **Configuration**
  - **query traces from the FOLIO library IR system**
  - ***Relevant*(q, DB) = { db in DB | *RSize*(q, db) > 0 }, *Best*(q, DB)**
  - ***Ind* : tend to underestimate the result size of the queries**

- **Results**
  - **distinguish two databases**
    - » ***Chosen/Ind* = 0 only if *Relevant* = 0 (or the case of *Best* = 0)**
    - » ***Success(C/ex, Ind)* are much lower than others**
    - » **the more unrelated subject domains of the databases considered were, the better *Ind* behaved in distinguiching the two databases**
  - **evaluate over six databases**

NTHU db-lab

Experimental Framework

NTHU db-lab

Improving *GLOSS*

- **Elimination of `*Subject*` index**
  - *Subject* **is a compound index built by merging together other `primitive` indexes**
  - **implicit `or` query : find** *Subject* computer
    - **find** *Title* computer **or** *Abstract* computer **or** ...
  - **two estimates of *freq*(***Subject* **<w>, <db>)**
    - **lower bound : max [*freq*(index(i) <w>, <db>)]**
    - **upper bound : sum [*freq*(index(i) <w>, <db>)]**
- **Reduction of histograms**
  - **threshold : drop the entries of very low frequency**
  - **classification : define a set of ranges of frequencies**
- **More flexible definitions**

NTHU db-lab

Improving *GLOSS*

Conclusions

- **Contributions**
  - **a formal framework for the text database discovery problem**
  - **concept of routing queries to appropriate information sources based on previously collected frequency statistics about sources**
  - **some estimators that may be used to make decisions**
  - **an experimental evaluation according to different semantics**
- **Future research**
  - **hybrid estimator for *GLOSS***
    - » ***C/ex : Est/Binary; C/ab, C/ob, C/sm : Est/Ind***
  - **incorporate the cost of charge into the computation of *ESize/est***
  - **extend the boolean model to the *vector-space* retrieval model**

    Generalizing GLOSS to Vector-Space Databases and Broker Hierarchies
    Proceeding of VLDB Conference, 1995