Humboldt Universität Informatik

Humboldt University

Computer Science Department
Systems Architecture Group
http://sar.informatik.hu-berlin.de

Data Retrieval in Intermittedly Connected Networks

Dirk Neukirchen

Interplanetary Internet Seminar, 2006

June 22, 2006

Overview

- Traditional Approaches vs. IR Approach
- Document Scores and SNAQ (new)
- Evaluation Client Server – does it work ?
- Evaluation DT(M)N

History

- Previous Routing Strategies / Buffer Updates:

- Flooding
- Epidemic Routing
- Planned/Unplanned Connectivity Patterns

- Information Retrieval Technology

Differences - Scenarios

- previous:
  - Private End Users
  - Providing „Internet Services“ in (disaster) areas
  - Packet/Bytestream based
- now:
  - Organisation (WHO, ...)
  - Sharing,Retrieving,Updating of essential information
  - Text based

  - Example:

  Medical Manuals

  Medical References

  Immunization Algorithms

Differences – Buffer Update

- previous:
  - >1 packet per „text“
  - Data fragmentation
- now:
  - Application network
  - More scenario specific

- Privacy ?
- Internet not Possible ?

Document Scoring

- Document score -> buffer drop

- „simple“ (query likelihood)

$$Score(D) = P(Q|D) = \prod_{w \in Q} P(w|D)$$

$$Score(D) = \prod_{w \in Q} \lambda P_{doc}(w|D) + (1-\lambda) P_{coll}(w|C)$$

P_M (a|M): # a occuring in M / # terms in M

D: document

C: collection

P (w|C) : smoothing

Document Scoring - Problems

(1) Different collections at different peers

– Different word distribution

(2) Query dependent (long query score)

$$Score(D) = \prod_{w \in Q} \lambda P_{doc}(w|D) + (1-\lambda) P_{coll}(w|C)$$

Document Scoring - Collection

- Collection Independence

$$Score(D) = \prod_{w \in Q} \lambda P_{doc}(w|D) + (1-\lambda) P_{coll}(w|C)$$

Smoothing: general purpose English collection known to all peers

Document Scoring - Query

- Query Independence

„trivial“ Negative KL-divergence ! ;-)

Idea: combine document model with query model

formal: $$Score(D) = -KL(\theta_Q \| \theta_D) = \sum_w \theta_Q(w) \log \frac{\theta_D(w)}{\theta_Q(w)}$$

model is estimated by relative frequency of terms in query

- KL ~ distance of 2 language models

Document Scoring - SNAQ

- Score Normalization Across Queries

$$Score(D) = -KL(\theta_Q \| \theta_D) - (-KL(\theta_Q \| \theta_C))$$

- Query/Document and Query/Corpus

$$Score(D) = \frac{1}{|Q|} \sum_{w \in Q} \log \frac{\lambda P_{doc}(w|D) + (1-\lambda) P_{coll}(w|C)}{P_{coll}(w|C)}$$

P_M (a|M): # a occuring in M / # terms in M

Evaluation SNAQ

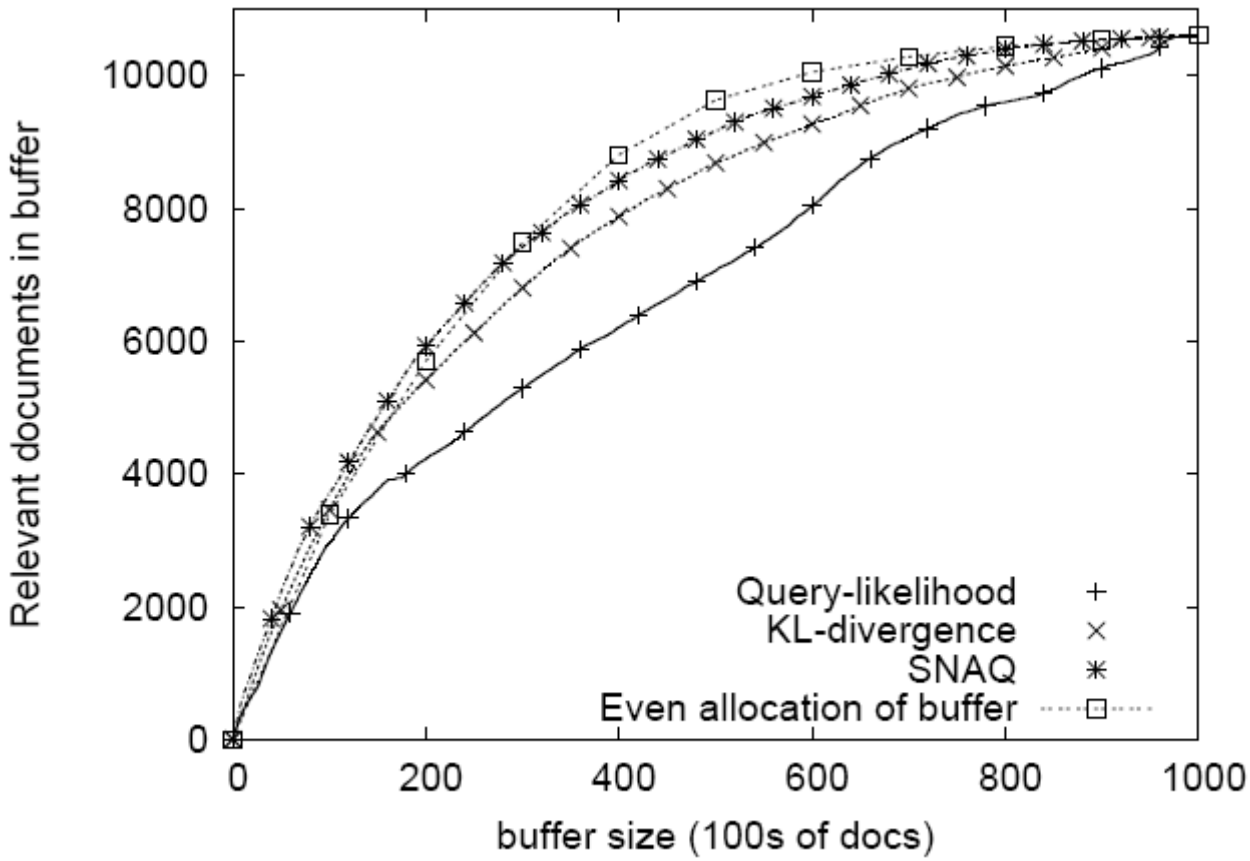
- Score Models
  - Query-likelihood
  - KL divergence
  - SNAQ
  - Even allocation of the buffer

- Client / Server structure
- 100 clients (unique collections)
- Use premade collections / corpuses
- Return top 10 documents
- 100 queries to all clients

- $\lambda = 0{,}6$

Evaluation SNAQ

- 100 queries, 100 clients, 10hits ->100.000 retrievals



Scenario Model

- Homogenous peers
  - Wireless radio
  - Local collection of documents – Unique, Static
  - Finite buffer
- No long network paths
  - connectivity instead of mobility
  - small diameter network
- Use buffer to store & foward
- Meeting Value Routing
- Transfer opportunities

Evaluation Distributed SNAQ

- Routing Algorithm

Rank(D) ~ (1-α)Deliverable(D) + α Score(D)

- α=0 traditional

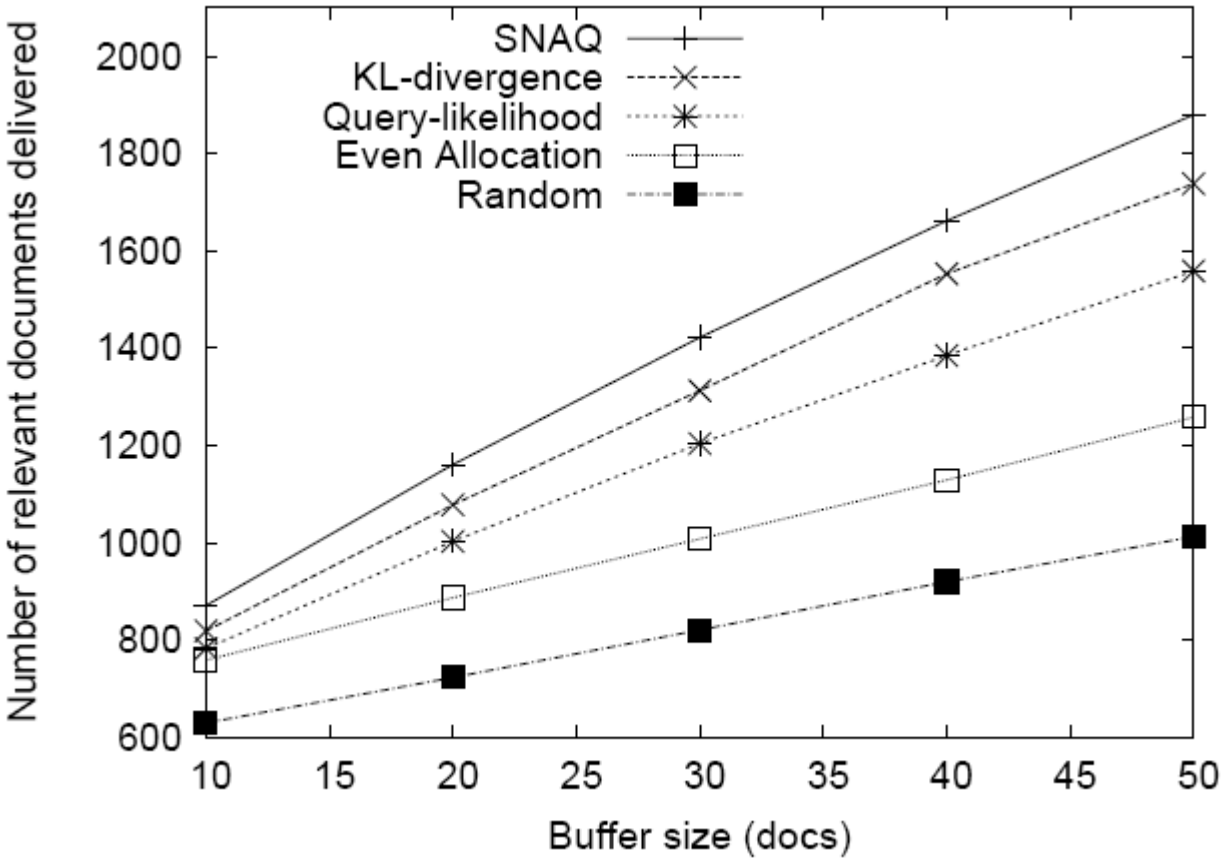
- Score = SNAQ etc.

- Deliverable : meeting value metric
  - List of network nodes with „meeting values“ (init = zero)
  - P meet Q -> +1 mv
  - Add Routes over Q -> Multihop mv
  - Degrade mv over time

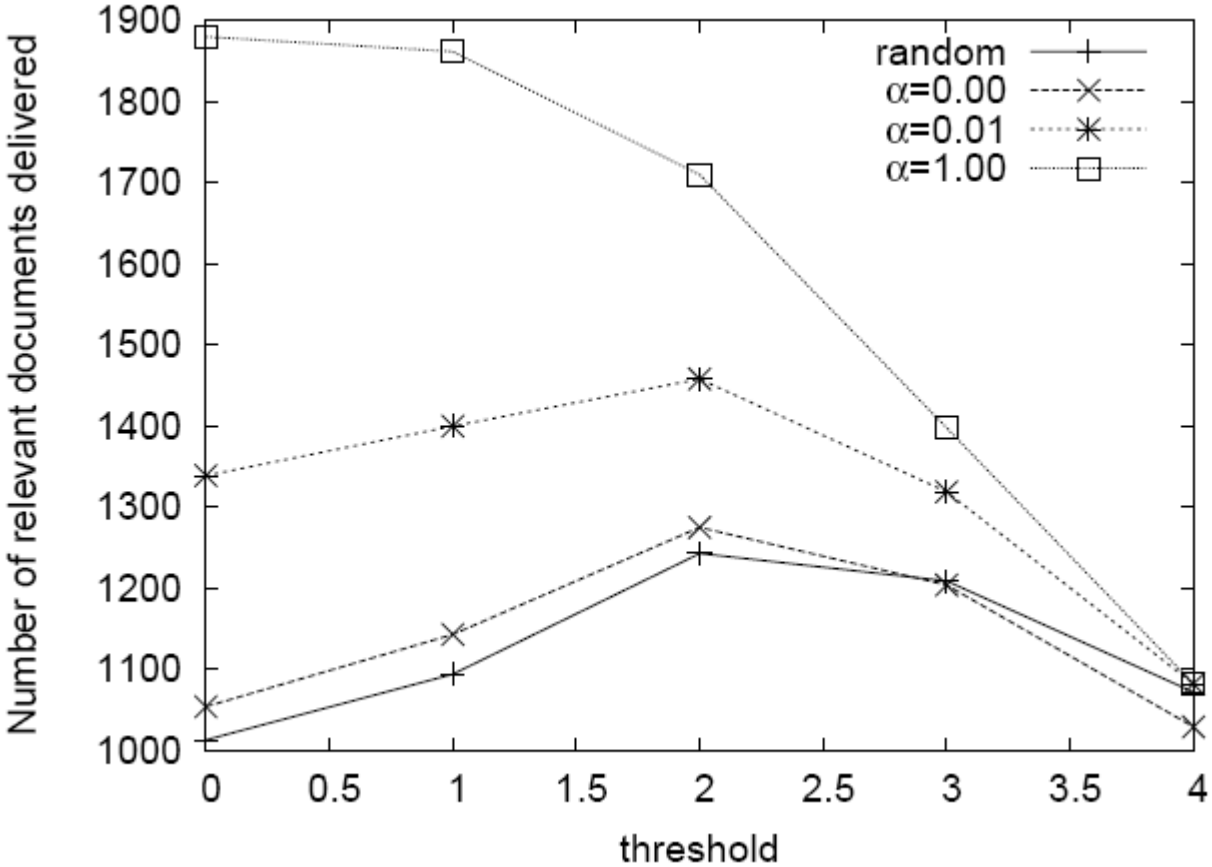
Transfer Opportunities

- Delivery check
  - B queries A
  - A delivers document
  - A delete document from local buffer
- Query exchange
  - Exchange new queries Q=((w1,...,wn),source)
- Retrieval
  - Get >1 top-ranked documents N(A)
  - Exchange scores: score [N(A)] + score [buffer A]
- Scoring / Buffer management
  - Create document ranking (in a: score : buffer A,N(B),buffer B)
  - Drop decisions
- Document exchange

Evaluation Distributed SNAQ



Evaluation Distributed SNAQ



Threshold : document score

Conclusions

- Random Drop is bad

→ use language models

- Less documents delivered with SNAQ
- More „important“ documents delivered !

- Document Scores „easier“ than Routing Scores

- Problem/use case: document based system