Approximate Object Location and Spam Filtering on Tapestry

Feng Zhou (zf@cs.berkeley.edu)
Li Zhuang (zl@cs.berkeley.edu)
Ben Y. Zhao (ravenben@cs.berkeley.edu)
Ling Huang (hling@cs.berkeley.edu)

Motivation

- **Objective: Search for similar content published on Tapestry**
  - Text documents: same news article altered due to formatting
  - General content with descriptive fields: MP3 file with textual tags
- **Content-hashed GUID**
  - Hashing object content to get GUID
  - Good for locating exact copies
- **Our Way**
  - Describe each object using a set of feature values
  - Build an index of these feature values on top of Tapestry
  - Search this index to find GUID of matching objects

Approximation Extension to Tapestry

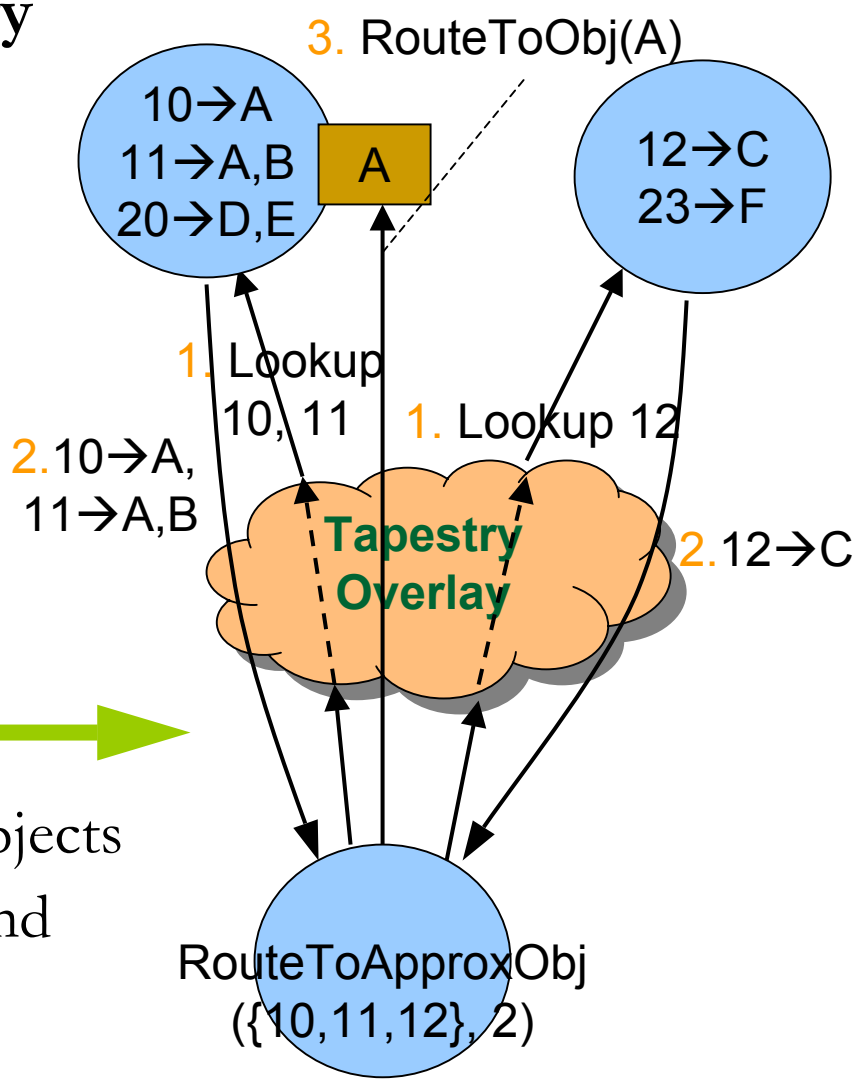
- **Publication using features**
  - Objects are described using a set of features:
    AO ≡ Feature Vector (FV) = $\{f_1, f_2, f_3, \ldots, f_n\}$
  - Location ≡ find all objects in the network with
    $|FV^* \cap FV| \geq THRES, 0 < THRES \leq |FV|$
- **Primitives**
  - PublishApproxObject(Object ID, FV)
  - UnpublishApproxObject (Object ID, FV)
  - RouteToApproxObject (FV, THRES)

Potential Applications

- **Approximate Text Addressing**
  - Problem: find similar text document copies
  - Feature Vector: **a text fingerprint vector**
  - Application: P2P spam filter, P2P content based "**e-pinions**", etc.
- **Database Queries on P2P**
  - Hash values of a tuple into a feature vector
  - Feature Vector: **hashes of values to query**
  - Approximate query: **THRES < |FV|**
- **Media Retrieving on P2P**
  - Most of pattern recognition results of image or video are represented as a feature vector.
  - Discretize feature values

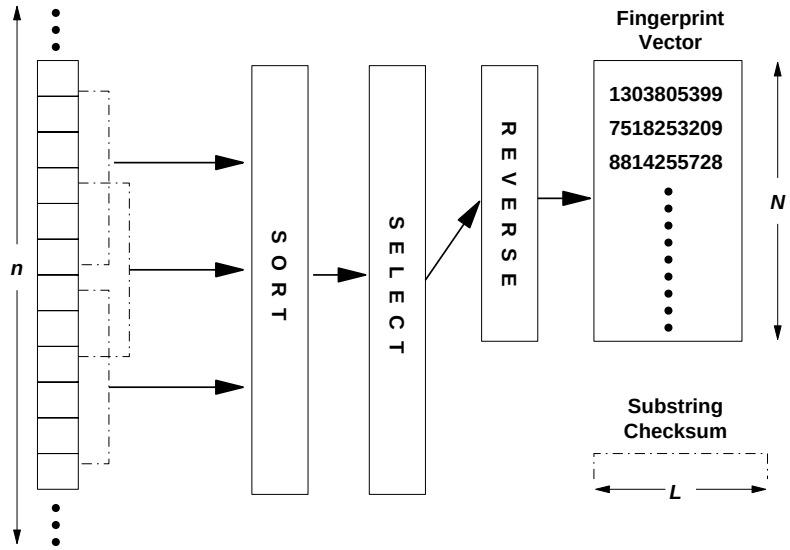
Prototype on Tapestry

- **A substrate on top of Tapestry**
- **Feature Object**
  - The set of IDs of all objects matching a feature value.
- **PublishApproxObject**
  - Add Object ID to all involved Feature Objects
  - Publish new Feature Objects if needed
- **RouteToApproxObject**
  - Lookup all involved Feature Objects
  - Count occurrence of each ID and compare with THRES
  - RouteToObject



Approximate Text Addressing

- **Fingerprint Vector [manber94finding]**
  - Divide document into (n-L+1) overlapping substrings (length L)
  - Calculate checksums of substrings
  - The largest N checksums → FV
- **Parameters**
  - Length of substrings: L
  - Length of checksums: L_{ck}
  - FV Size: |FV|=N
- **Two sets of experiments**
  - **"Similarity"**: digest slightly changed documents into the same or similar FV ? – **The higher the better! (1 - False-negative)**
    - We developed an **analytical model** for this
  - **False-positive**: digest totally different documents into the same or similar FV ? – **The lower the better!**



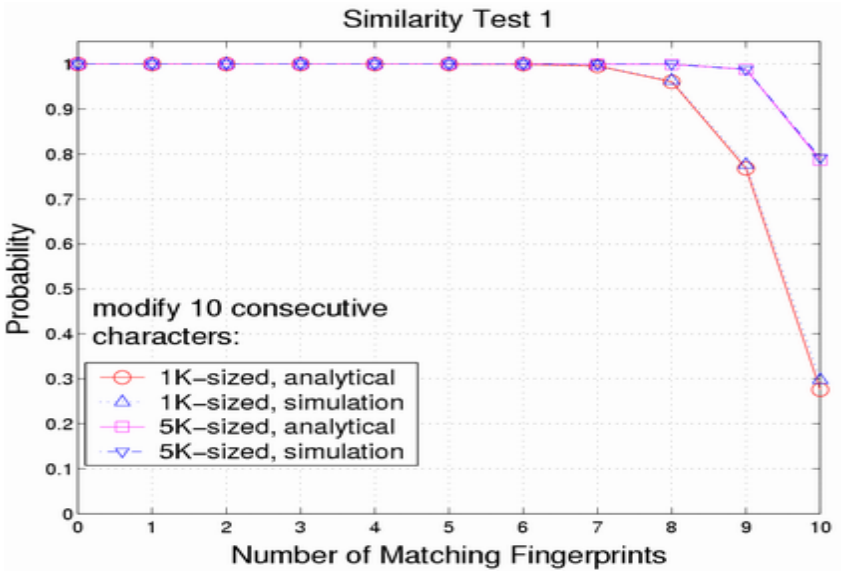
P2P Spam Filtering

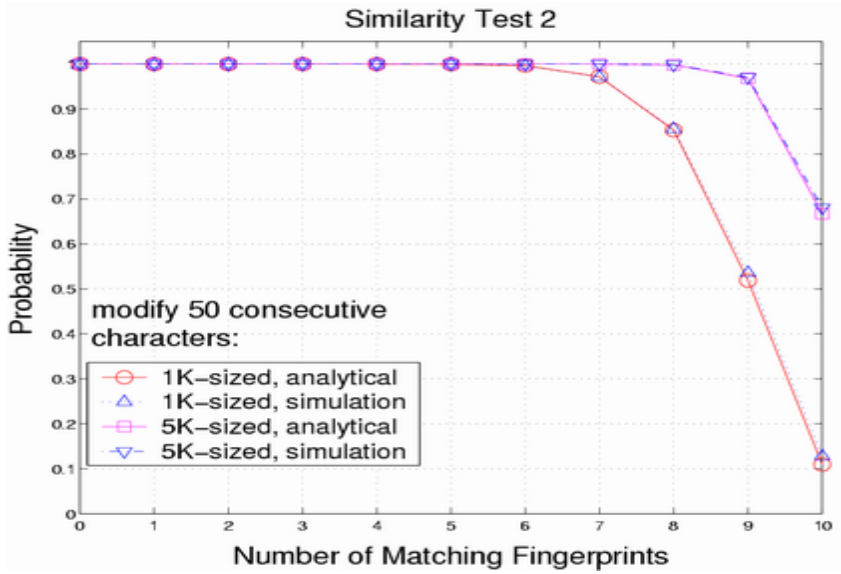
- **Collaborative Spam Filtering**
- **Observation: Human recognition is the only fool-proof spam identification tool.**
- **Basic idea: connecting university or company-wide servers into a Tapestry network.**
- **Why P2P?**
  - Compared with existing university/company-wide systems:
    The effectiveness of collaborative spam filtering systems grows with the number of users
  - Compared with existing periodically updated systems
    Timeliness of information is vital in spam filtering systems

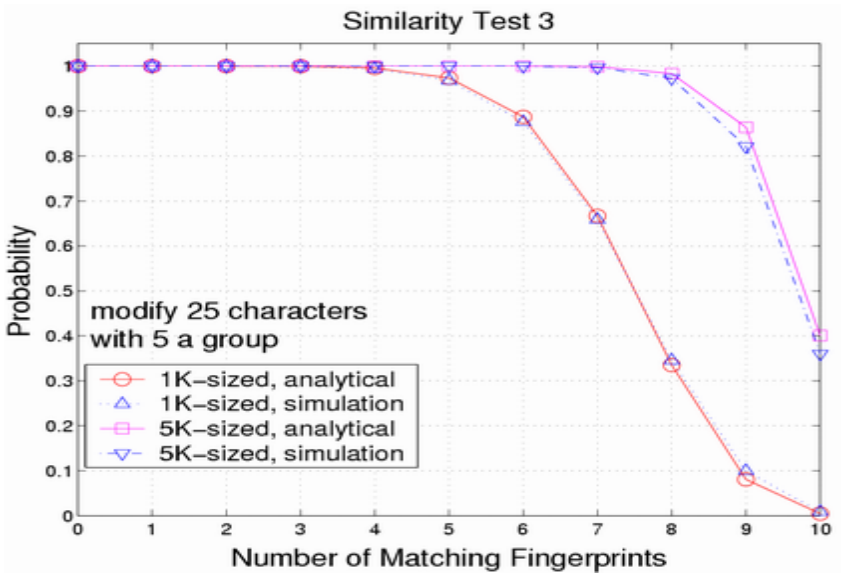
P2P Spam Filtering (cont.)

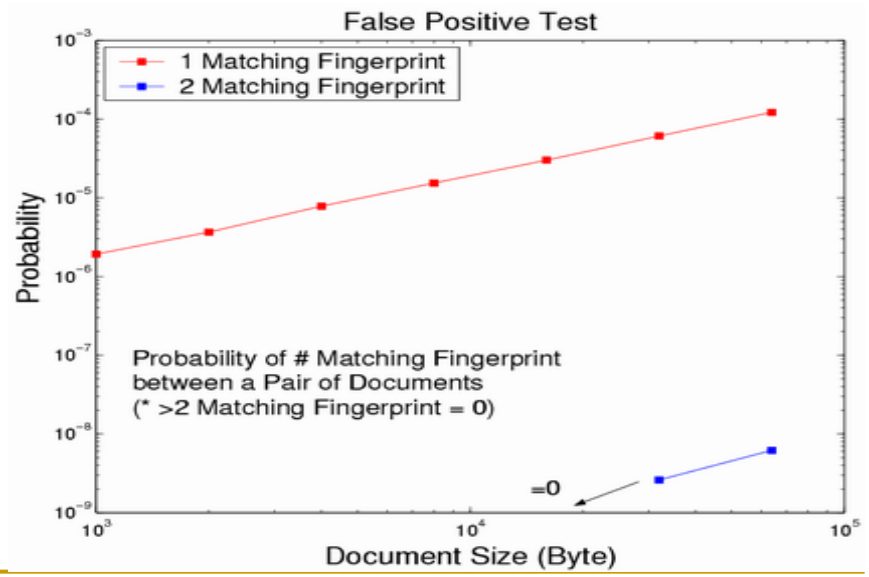
- **Fingerprint Vectors for Spam Filtering**
  - Length of substring (L): one to several phases
  - |FV|: large enough to avoid collisions
  - THRES is decided by doing **"Similarity" Test** and **"False Positive" Test**
- **Locating Spam Using Extended API**
  - Vote: RouteToApproxObject() to vote or PublishApproxObject() new one
  - Check: RouteToApproxObject() to get current votes
  - Performance Considerations
    - **N↑→ Accuracy↑, Network bandwidth consumption↑**
    - Non-Spam: never published in the network → need to route to ROOT before getting negative result
      - Solution: **TTL** (tradeoff between accuracy and bandwidth)

Evaluation of FV on Random Text



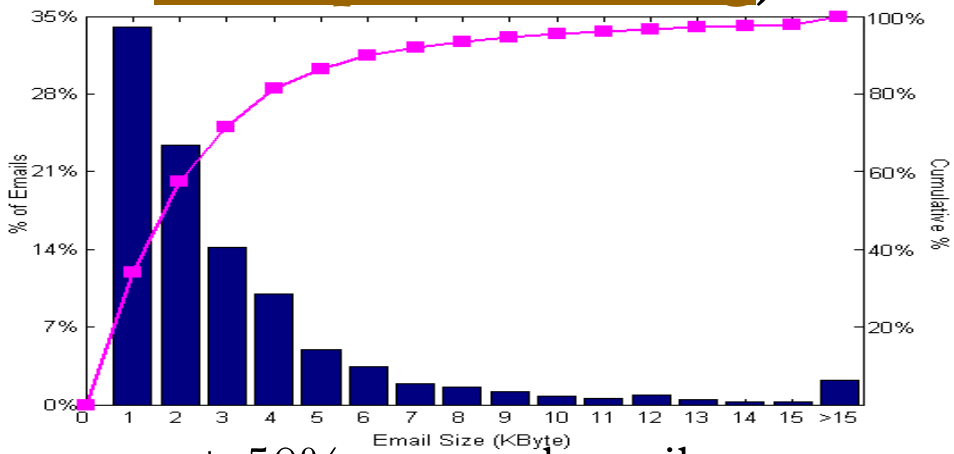






Evaluation of FV on Real Emails

- **Spam (29631 Junk Emails from www.spamarchive.org)**
  - 14925 (unique)
    5630 (exact copies)
    9076 (modified copies of 4585 unique ones)
    86% of spam ≤ 5K
- **Normal Emails**
  - 9589 (total)=50% newsgroup posts + 50% personal emails



"Similarity" Test

3440 modified copies of 39 emails, 5~629 copies each

THRES	Detected	Fail	%
3/10	3356	84	97.56
4/10	3172	268	92.21
5/10	2967	473	86.25

"False Positive" Test

9589(normal)×14925(spam) pairs

Match FP	# pair	probability
1/10	270	1.89e-6
2/10	4	2.79e-8
>2/10	0	0

Evaluation and Status

- **Effective Fingerprint Routing w/ TTL**
  - Network of 5000 nodes
  - Diameter latency=400ms
  - 4096 Tapestry nodes
- **Status**
  - Approximate Text Addressing prototype implemented on Tapestry.
  - **SpamWatch** – P2P spam filtering system prototype implemented
  - Outlook add-in usable!
  - Website: **http://www.cs.berkeley.edu/~zf/spamwatch/**