

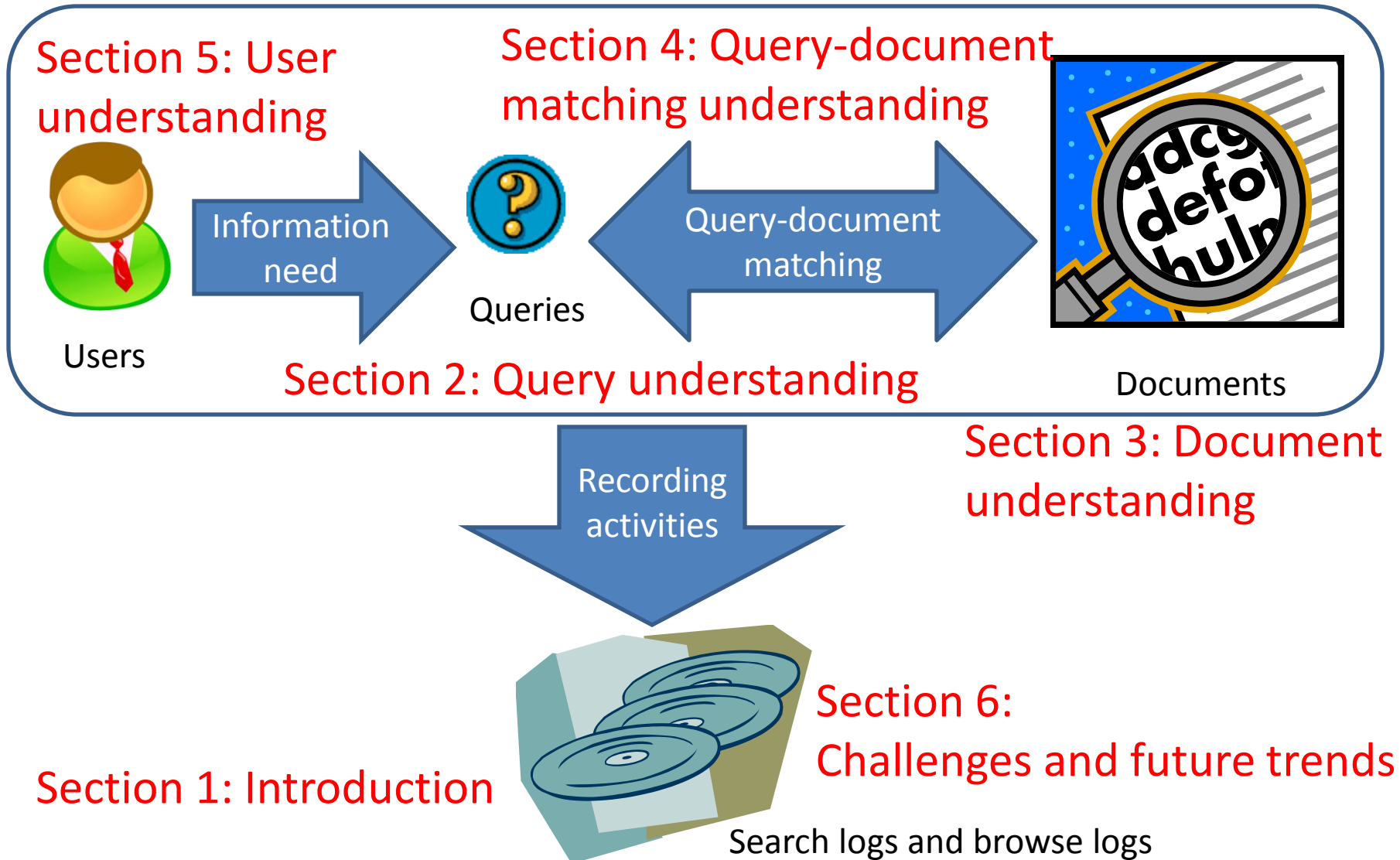
Web Search/Browse Log Mining: Challenges, Methods, and Applications

Daxin Jiang
Microsoft
Research Asia

Jian Pei
Simon Fraser
University

Hang Li
Microsoft
Research Asia

A Road Map



Agenda

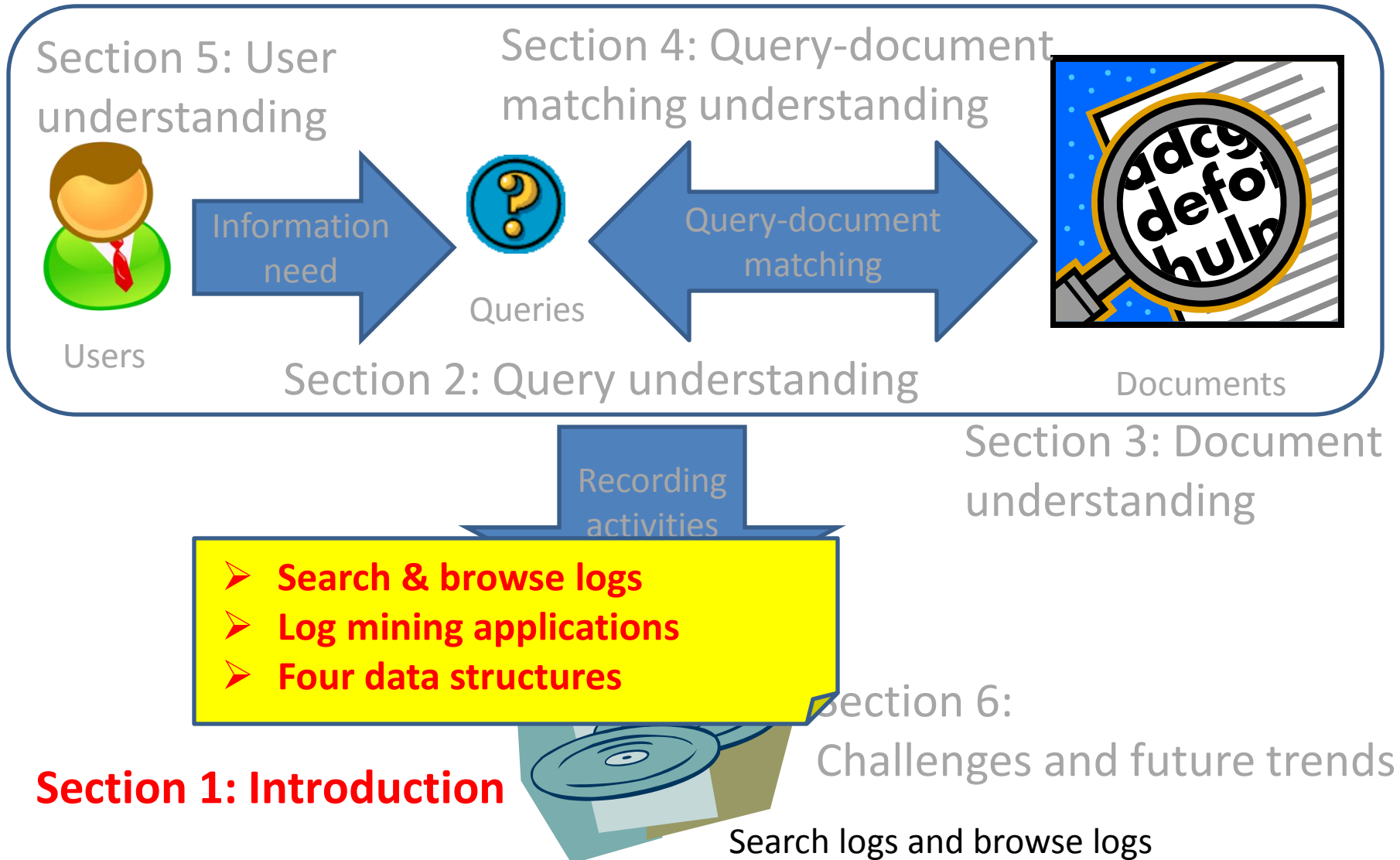
Morning

- [9:00-9:30] Introduction (Daxin Jiang)
- [9:30-10:30] Log mining for query understanding – Part I (Hang Li)
- 15 minutes Coffee Break
- [10:45-11:45] Log mining for query understanding – Part II (Hang Li)
- [11:45-12:30] Log mining for document understanding – Part I (Jian Pei)

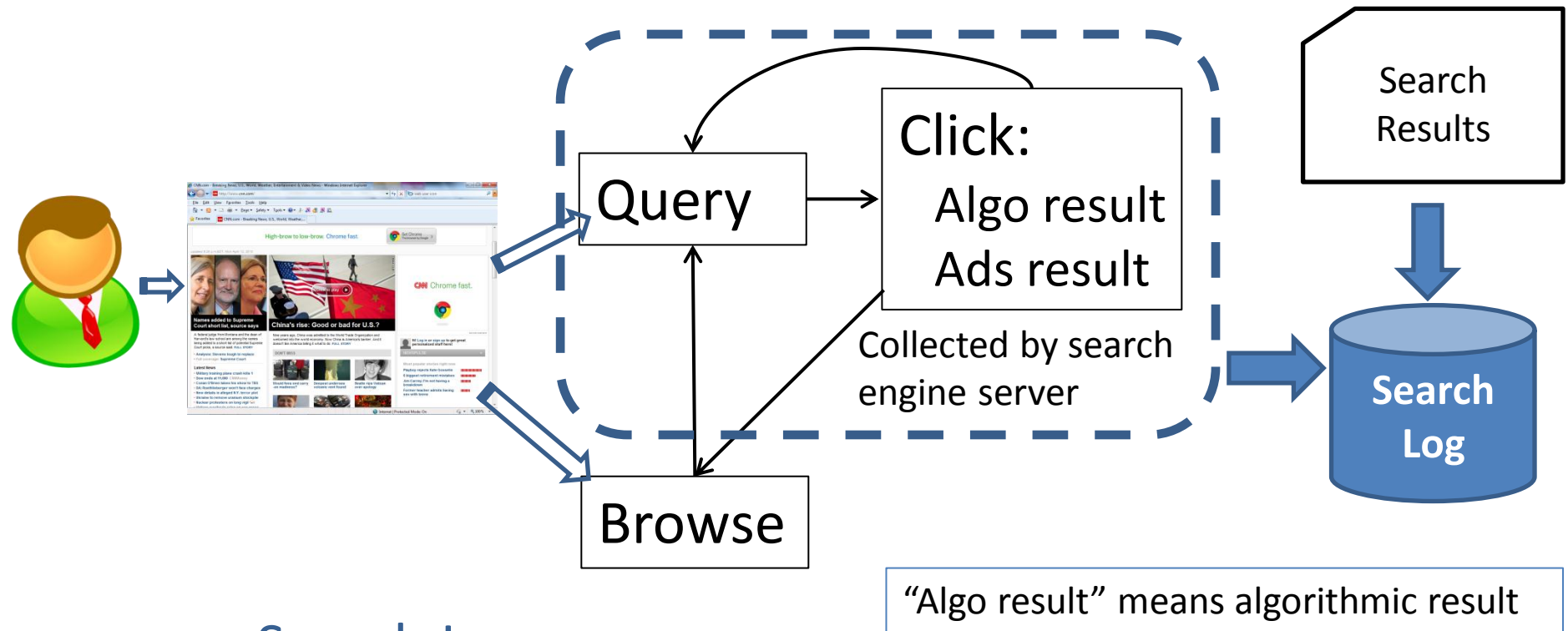
Afternoon

- [14:00-14:30] Log mining for document understanding – Part II (Jian Pei)
- [14:30-15:30] Log mining for query-document matching (Jian Pei)
- 15 minutes Coffee break
- [15:45-16:45] Log mining for user understanding (Daxin Jiang)
- [16:45-17:15] Challenges and future trends (Daxin Jiang)
- [17:15-17:30] Q & A (all)

A Road Map



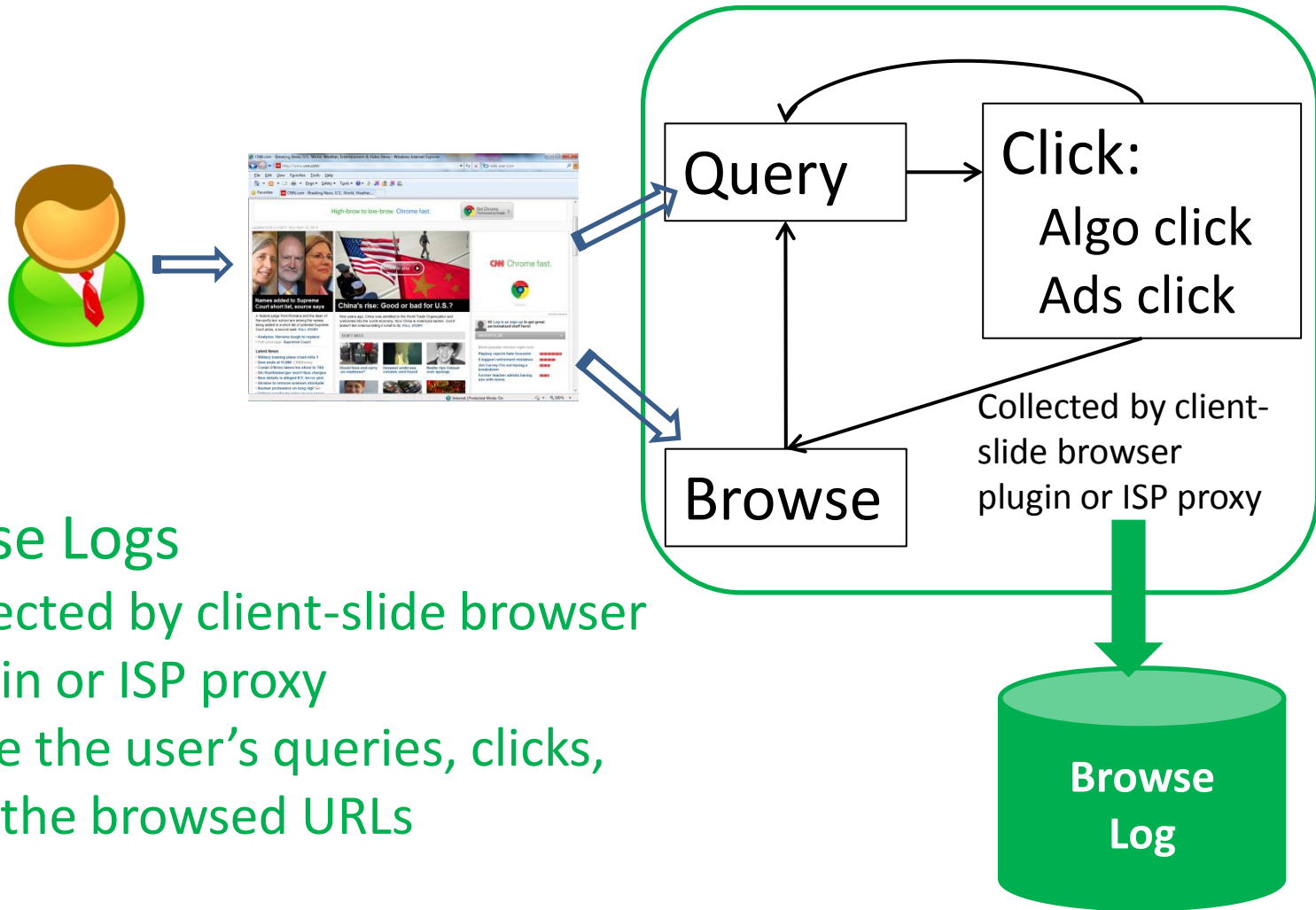
Different Types of Log Data: Search Logs



Search Logs

- Collected by search engine server
- Record the user queries, clicks, as well as the search results provided by the search engine

Different Types of Log Data: Browse Logs



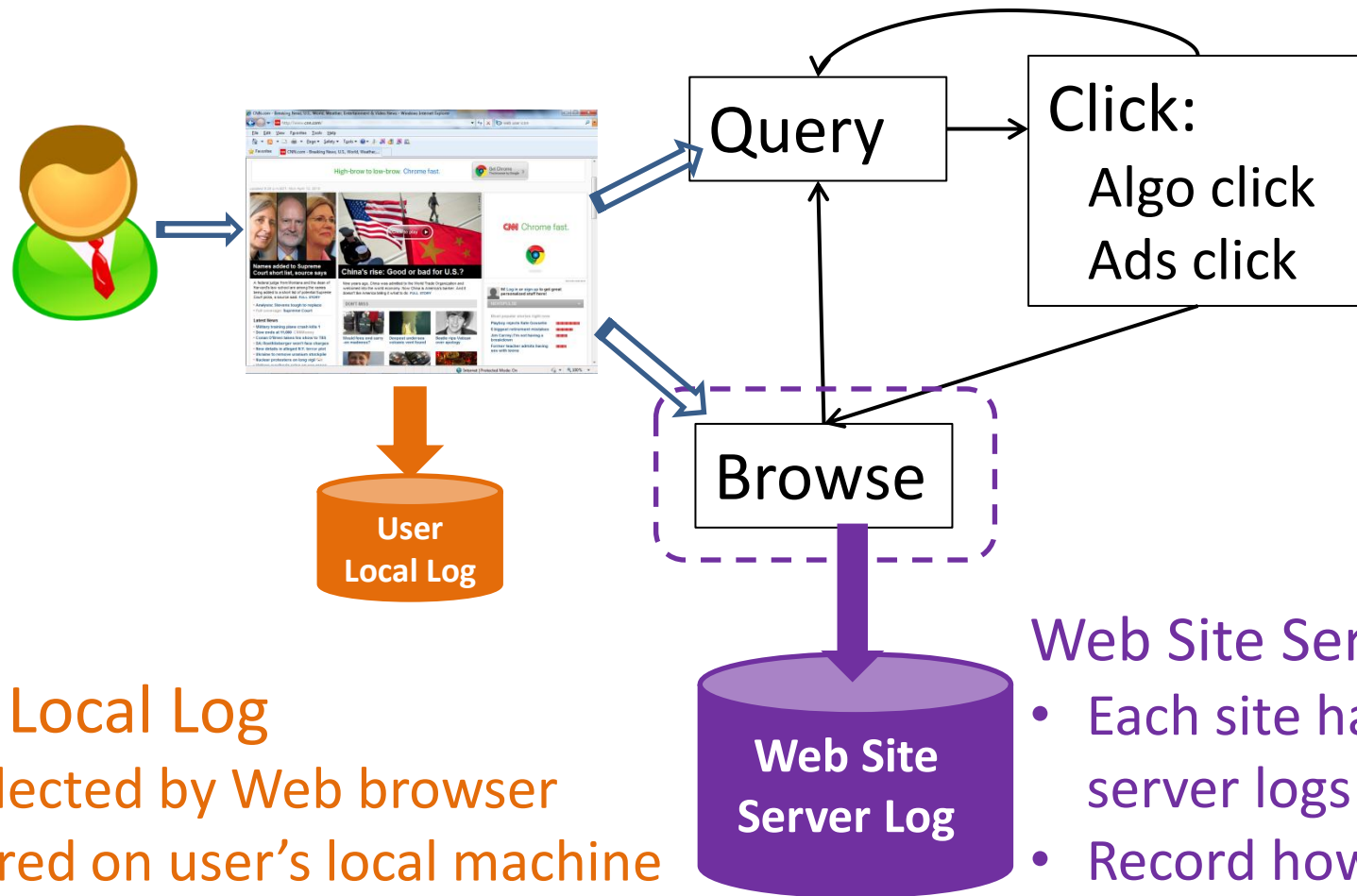
Browse Logs

- Collected by client-side browser plugin or ISP proxy
- Store the user's queries, clicks, and the browsed URLs

The Lemur toolkit. <http://www.lemurproject.org/querylogtoolbar/>.

White, R.W., et al. Studying the use of popular destinations to enhance web search interaction. SIGIR'07.

Different Types of Log Data: Other Logs



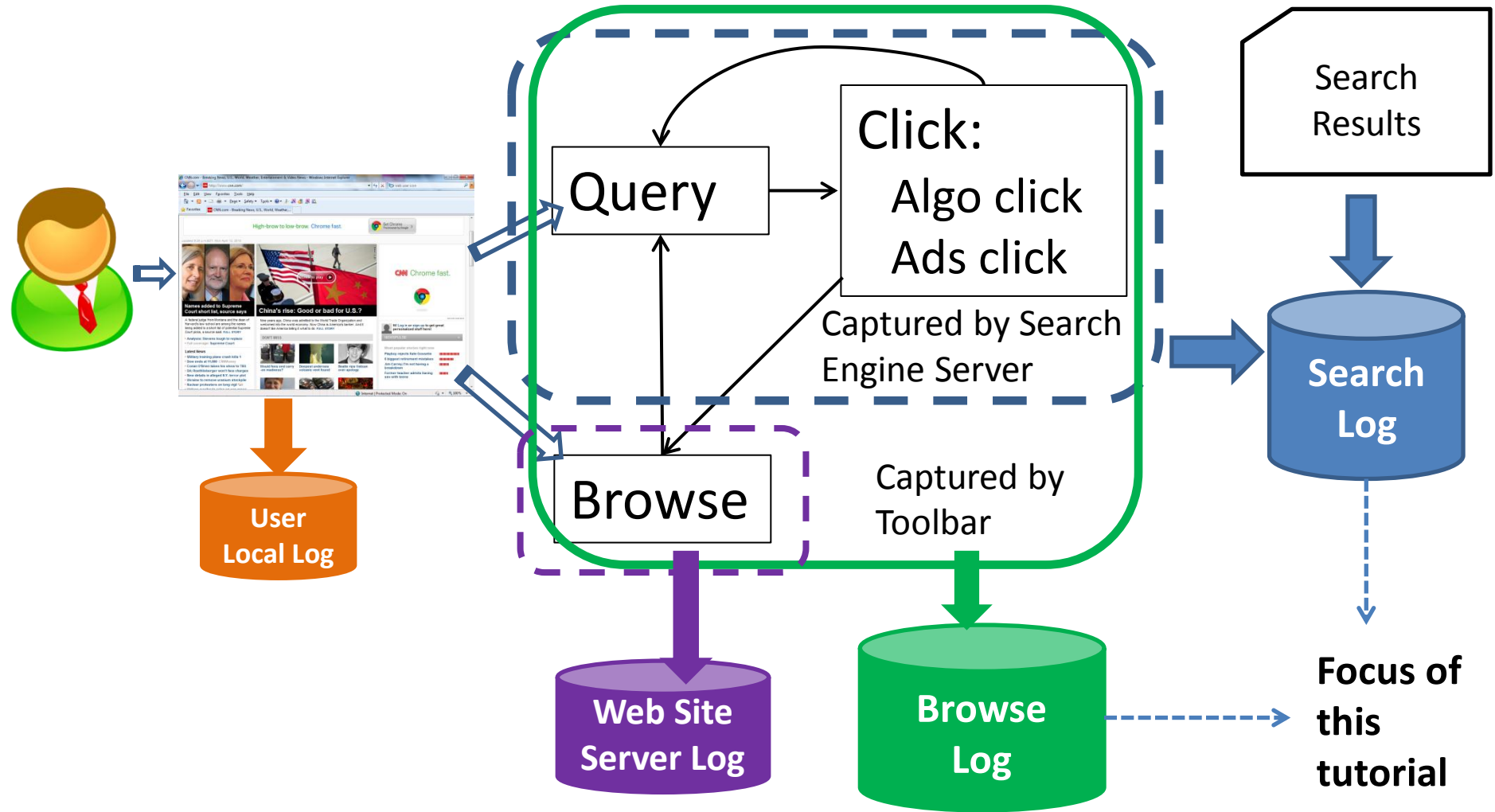
User Local Log

- Collected by Web browser
- Stored on user's local machine
- Contains richer information, e.g., user's every input in browser

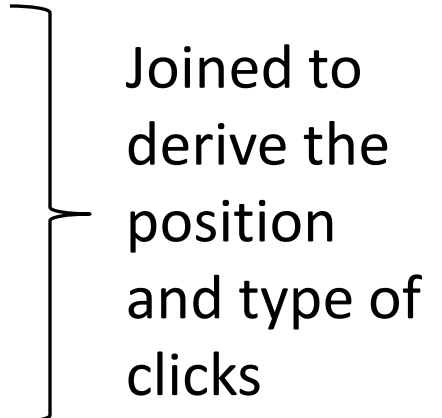
Web Site Server Logs

- Each site has its own server logs
- Record how users visit the site

Putting them Together



Major Information in Search Logs

- Recorded by search engine servers
 - Four categories information
 - User info: user ID & IP
 - Query info: query text, time stamp, location, search device, etc
 - Click info: URL, time stamp, etc
 - Search results
 - Algo results, Ads results, query suggestions, deep links, instant answers, etc.
- 
- Joined to derive the position and type of clicks

Major Information in Browse Logs

- Captured by client-side browser plug-in or ISP proxy
- Major information
 - User ID & IP, query info, click info
 - Browse info: URL, time stamp
- Client-side browser plug-in has to follow strict privacy policy
 - Only collects data when user's permission is granted
 - User can choose to opt-out at any time

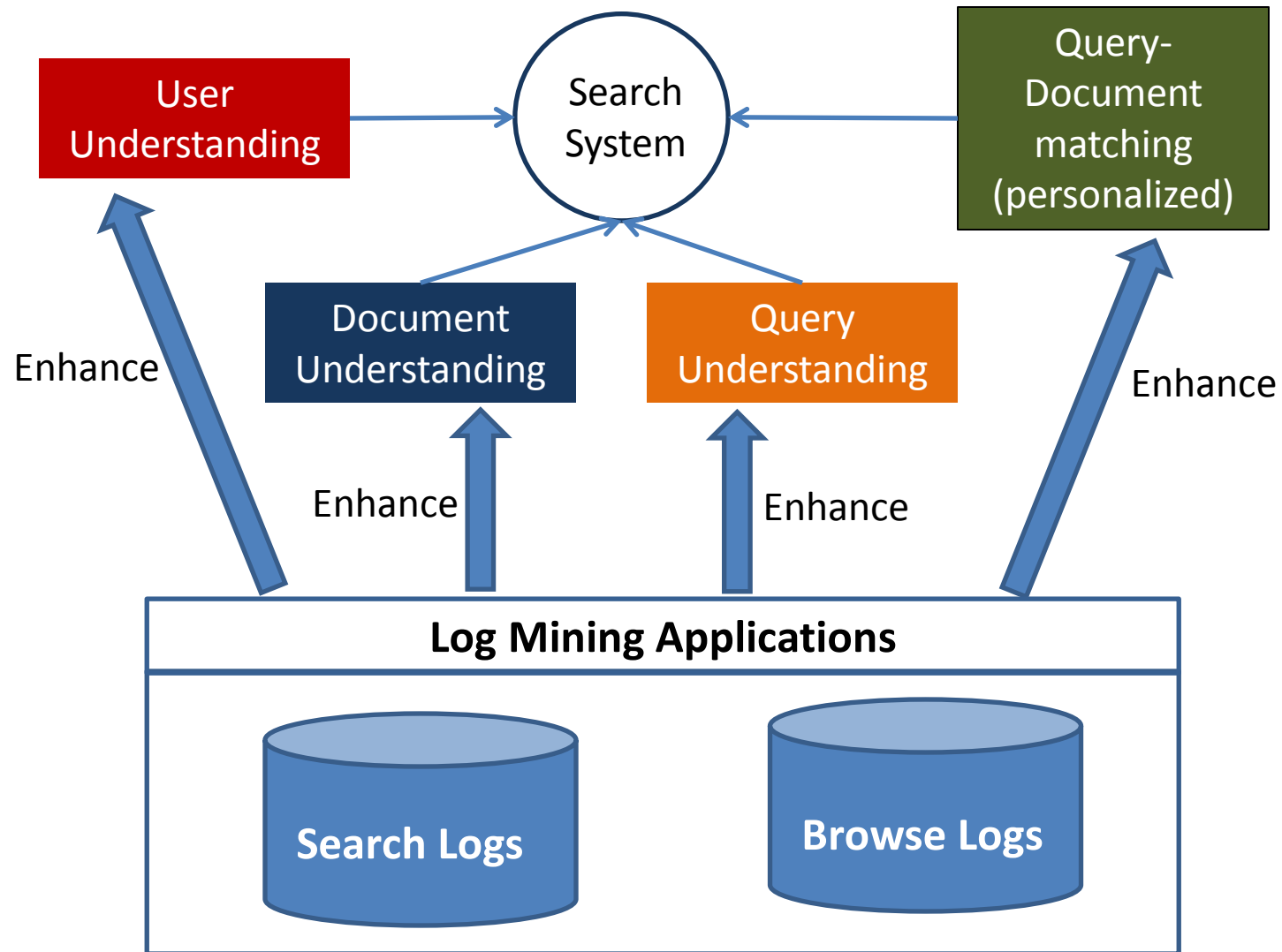
Log Mining Applications

- According to Silvestri and Baeza-Yates [Silvestri09]
 - Enhancing efficiency of search systems
 - Enhancing effectiveness of search systems

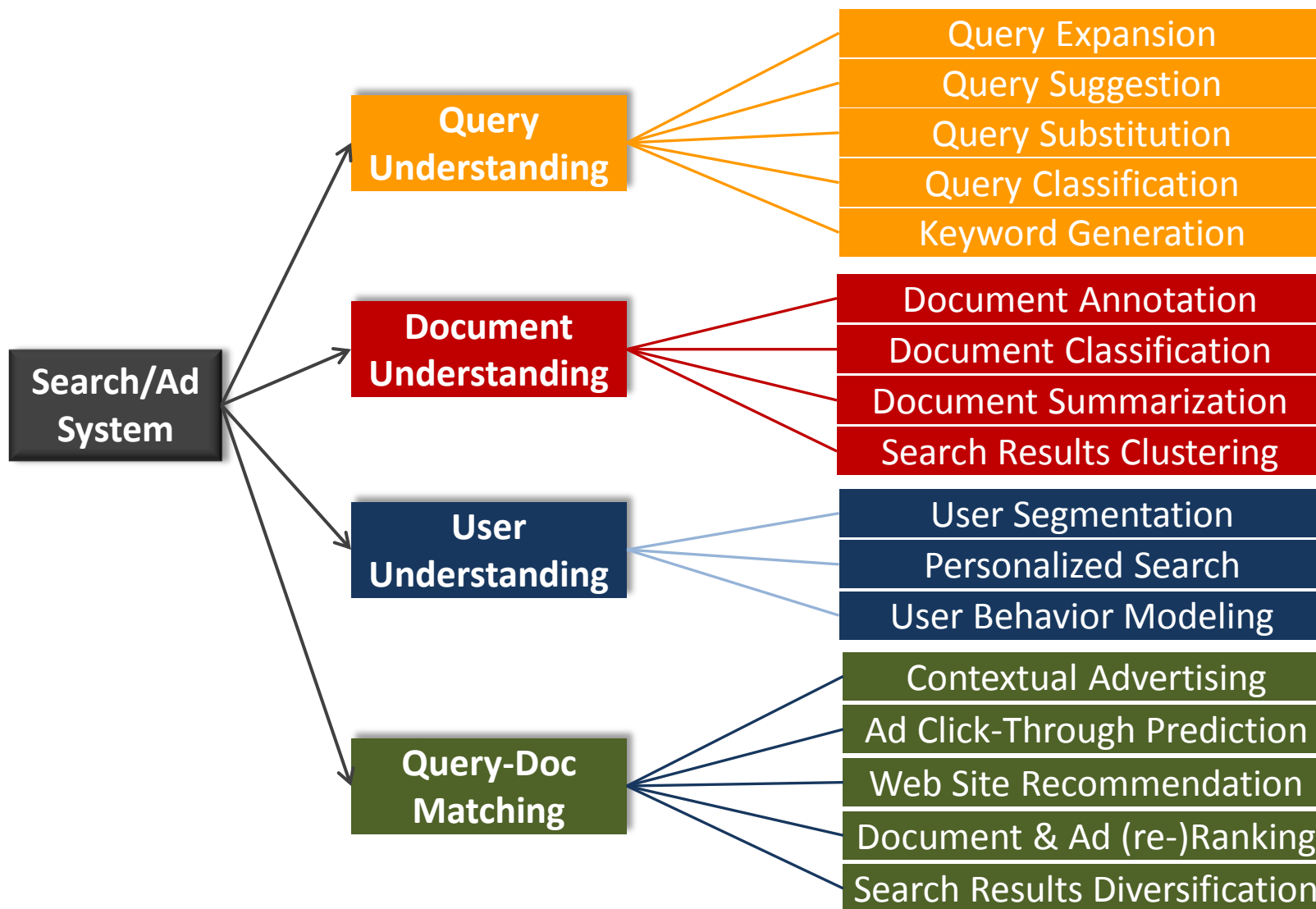
In this tutorial, we only focus on the effectiveness part

A search system provides both algo and Ads results

A View of Search System



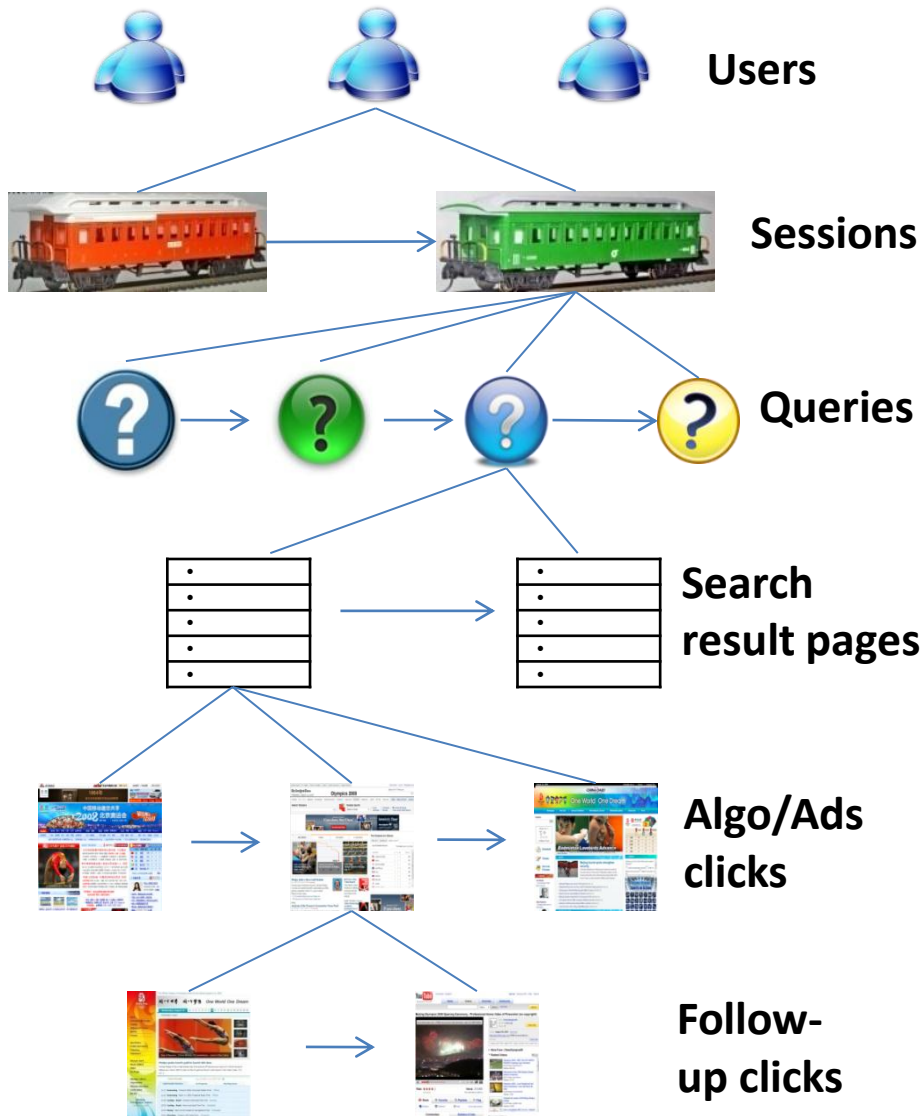
Log Mining Applications



Organizing Raw Logs by Common Data Structures

- Raw log data are stored in the format of plain text: unstructured data
- Can we summarize some common data structures from the textual logs to facilitate various log mining applications?
- Challenges: complex objects, complex applications

Complex Objects



- Various types of data objects in log data
- Complex relationship among data objects
 - Hierarchical relationship
 - Sequential relationship

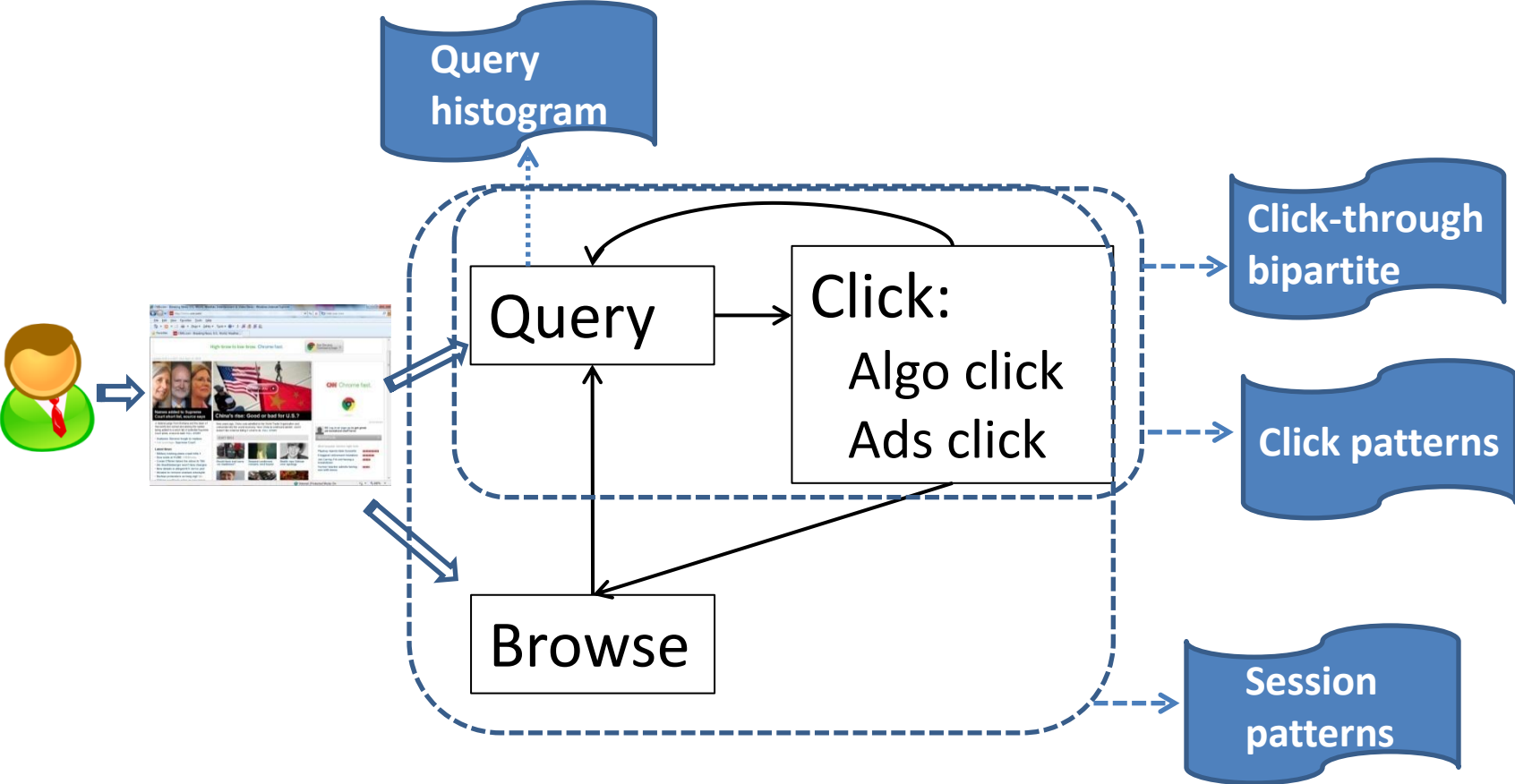
How to describe the various objects as well as their relationships?

Complex Applications

- Query understanding
 - Given a query q , what are the top-K queries following q in the same session?
- Query-Document matching
 - Given a query q , what are the top-K clicked Urls?
 - Given a Url u , what are the top-K queries lead to a click on u ?
- Document understanding
- User understanding

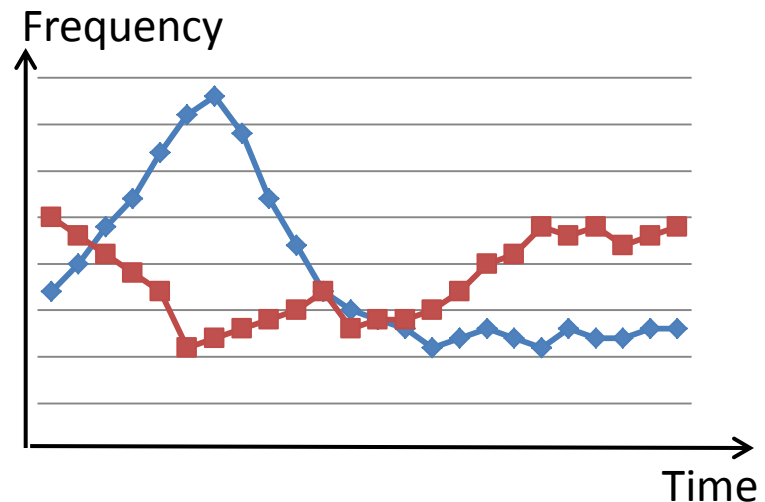
How to summarize the common data structures to support various applications?

Major Data Structures in Log Mining



Data Structure: Query Histogram

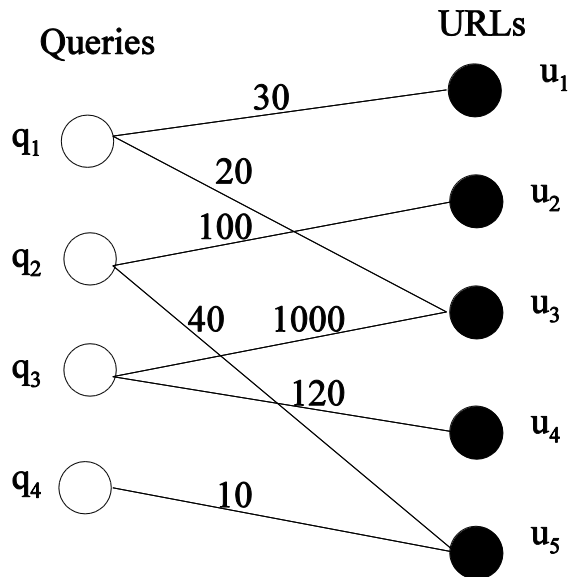
Query String	Count
facebook	3,157 K
google	1,796 K
youtube	1,162 K
myspace	702 K
facebook com	665 K
yahoo	658 K
yahoo mail	486 K
yahoo com	486 K
ebay	486 K
facebook login	445 K



Example applications:

- Query auto completion
- Query suggestion: given query q , find the queries containing q
- Semantic similarity & event detection: temporal changes of query frequency

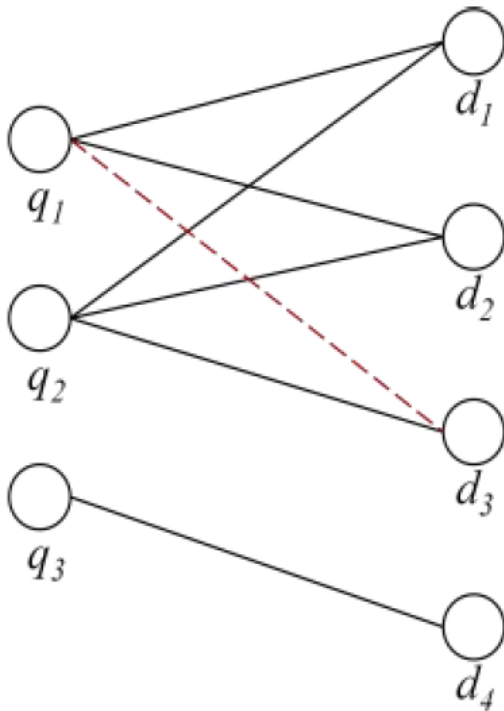
Data Structure: Click-through Bipartite



An example of click-through bipartite

- Example applications
 - Document (re-)ranking
 - Search results clustering
 - Web page summarization
 - Query suggestion: find similar queries

Random Walk



Construct matrix $A_{ij} = P(d_i | q_j)$ and matrix $B_{ij} = P(q_i | d_j)$

Random walk using the probabilities

Before random walk, document d_3 is connected with q_2 only; after a random walk expansion, d_3 is also connected with q_1 , which has similar neighbors as q_2

Data Structure: Click Pattern

Query

×	Doc 1
	Doc 2
	...
×	...
	...
	...
	...
	...
	...
	Doc N

Pattern 1
(count)

	Doc 1
×	Doc 2
	...
	...
	...
	...
	...
	...
	...
×	Doc N

Pattern 2
(count)

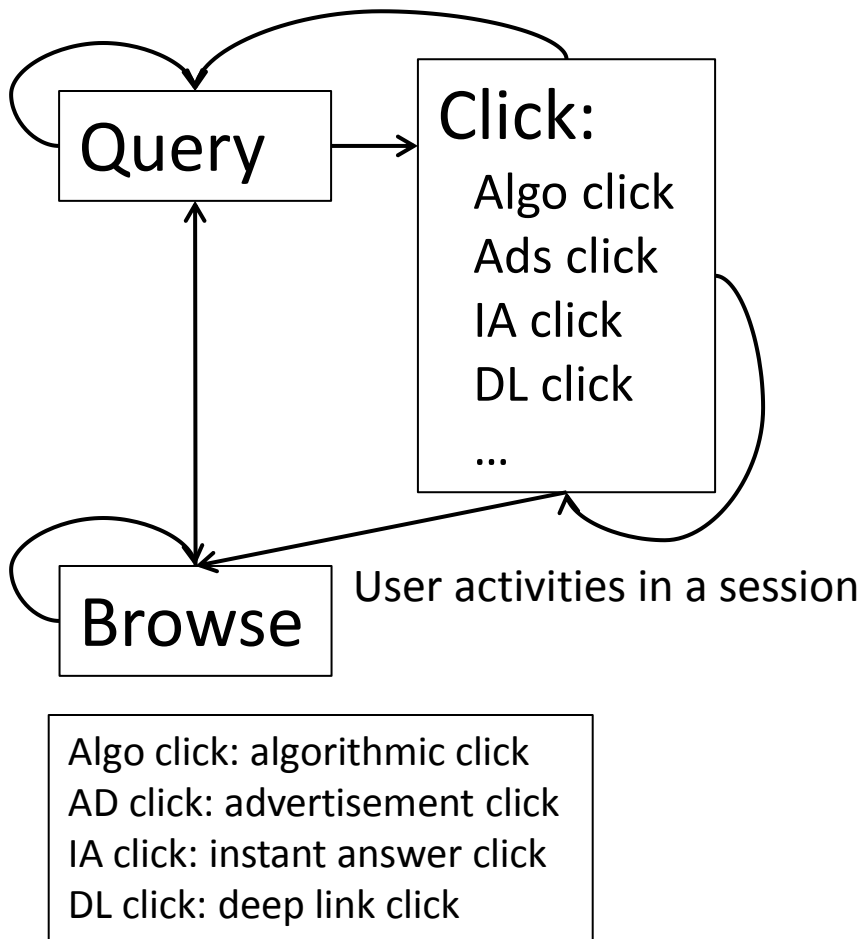
...

×	Doc 1
×	Doc 2
	...
×	...
	...
	...
	...
	...
	...
	Doc N

Pattern n
(count)

- More information than click-through bipartite
 - Relationship between a click and its position
 - Relationship between the clicked docs with un-clicked docs
- Example applications
 - Estimate the “true” relevance of a document to a query
 - Predict users’ satisfaction
 - Classify queries (navigational/informational)

Data Structure: Session Patterns



- Sequential patterns
 - E.g., behavioral sequences
 - SqLrZ [Fox05]

S: session starts; Q: query
L: receives a search result page
R: click; Z: session ends

- Example applications
 - Doc (re-)ranking
 - Query suggestion
 - Site recommendation
 - User satisfaction prediction

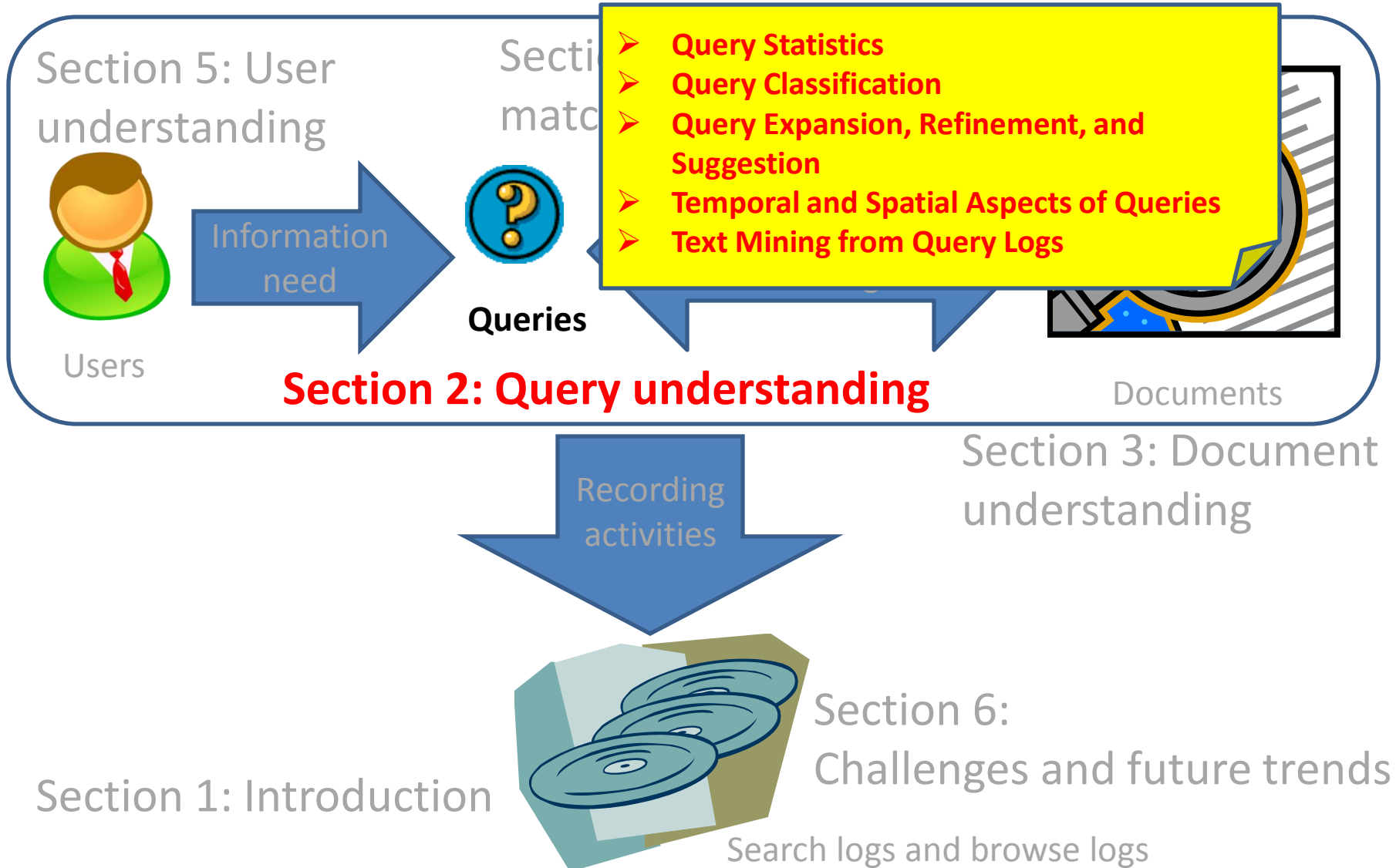
Session Segmentation

- Problem: given a sequence of user queries, where to cut the session boundary
- Features for session segmentation
 - Timeout threshold (e.g., [Silverstein99])
 - 30 minutes timeout is often used
 - Common words or edit distance between queries (e.g., [He02])
 - Adjacency of two queries in user input sequences (e.g., [Jones08])
 - Similarity between the top K search results of two queries (e.g., [Radlinski05])
- Tradeoff between cost and accuracy

Summary of Introduction

- Search & browse logs
 - Search logs: collected by search engine servers; store queries, clicks, and search results
 - Browse logs: collected by client-side browser plug-ins or ISP proxy servers; store queries, clicks, and browse information
- Log mining applications
 - Query understanding, document understanding, user understanding, query-document matching,
- Four data structures
 - Query histogram, click-through bipartite, click patterns, session patterns

A Road Map



Outline of Query Statistics

- Overview of Query Statistics
- Results of Query Statistics
- Summary of Query Statistics

Overview of Query Statistics

- Understand users' search behavior at macro level
 - **How users search:** query length, query & term frequencies, number of viewed search result pages, session length
 - **What users search for:** topic distribution
- Results of query statistics
 - [Cacheda01a, Cacheda01b, Holscher00, Jansen00, Jansen01, Jansen04, Jansen05, Jansen06, Silverstein99, Spink02, Spink02a, Wolfram01]
- Conclusion
 - Web search is different from traditional IR

Data Sets

Region	Data	Engine	Date	# of queries	# of sessions	Reference
US	Excite97	Excite	16 Sept, 1997	51 K	18K ¹	Jansen00, Jansen01, Spink02, Jansen06
	Excite99	Excite	1 Dec 1999	1M	326 K	Wolfram01, Spink02, Jansen06
	Excite01	Excite	30 Apr 2001	1M	262K	Spink02, Spink02a, Jansen06
	AV98	AltaVista	2 Aug- 13 Sept, 1998	575 M	285 M	Silverstein99, Jansen01, Jansen06
	AV02	AltaVista	8 Sept, 2002	1 M	369 K	Jansen04, Jansen06
Europe	FB98	Fireball	1-31 Jul. 1998	16 M	-	Holscher00, Jansen01, Jansen06
	BWIE00	BWIE	3-18 May, 2000	72K	83K	Cacheda01a, Cacheda01b, Jansen06
	FAST01	FAST	6 Feb, 2001	452 K	153K	Spink02a
	ATW01	AlltheWeb	6 Feb 2001	452K	153K	Jansen05, Jansen06
	ATW02	AlltheWeb	28 May 2002	957K	345K	Jansen05, Jansen06

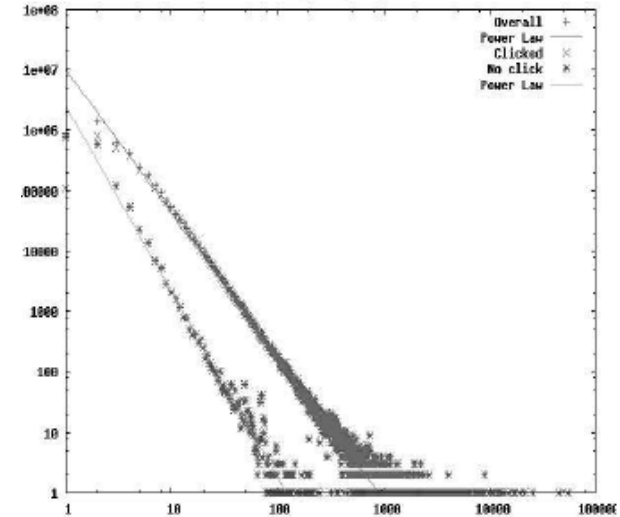
Query Length

Region	Data	Avg	1	2	3	>3	Reference
US	Excite97	2.21	31%	31%	18%	15%	Jansen00
	Excite99	2.4	29.8%	33.8%	36.4%		Wolfram01
	Excite01	2.6	26.9%	30.5%	42.6%		Spink02
	AV98	2.35	25.8%	26.0%	15.0%	12.6%	Silverstein99
	AV02	-	20%	-	-	-	Jasen06
Europe	FB98	1.66	54.6%	30.8%	10.4%	4%	Jansen01
	FAST01	2.3	25%	36%	39%		Spink02a
	ATW01	2.4	25.1%	35.8%	22.4%	15.9%	Jansen05
	ATW02	2.3	33.1%	32.6%	18.9%	15.1%	

- Average length: 1.66~2.6 words
- Much shorter than in traditional IR (6-9)
- Average length remains constant over time and across regions

Query & Term Frequencies

Region	Data	Head	Tail	Reference
US	Excite97 (term)	0.34% unique terms (occurrence >100) account for 18.2% traffic	44.8% unique terms (occurrence=1) account for 8.6 traffic	Jansen00
	Excite99 (term)	Top 100 terms account for 19.3% traffic	61.6% unique terms occurs only once	Wolfram01, Spink02
	Excite01	Top 100 terms account for 22.0% traffic	61.7% unique terms occurs only once	Spink02
	AV98	Top 25 queries account for 1.5% traffic	63.7% unique queries occur only once	Silverstein99
Europe	BWIE00	-	23.4% unique queries only occur once	Cacheda01a
	FAST01	Top 100 terms account for 14% traffic	-	Spink02a
	ATW01	Top 100 terms account for 15% traffic	7% unique queries only occur once	Jansen05
	ATW02	Top 100 terms account for 14% traffic	10% unique queries only occur once	



Head and tail parts are highly skewed

- Head: few queries/terms account for large traffic
- Long tail: consists of large percentage of unique queries/terms

Middle region follows Zipf distribution (the distribution of words in long English texts)

Number of Viewed Search Result Pages

	Data	Avg	1	2	3	>3	Source
US	Excite97	1.7	58%	19%	9%	-	Jansen00 Wolfram01
	Excite99	1.6	42.7%	21.2%	36.1%		Wolfram01
	Excite01	1.7	50.5%	20.3%	29.2%		Spink02
	AV98	1.39	85.2%	7.5%	3.0%	4.3%	Silversten99
	AV02	<2	73%	-	-	-	Jasen06,
Europe	FAST01	2.2	-	-	-	-	Spink02a
	FB98	<2	59.5%	-	-	-	Jansen01,
	BWIE00	<2	67.9%	13.2%	6.0%	-	Cacheda01a
	ATW01	<2	83.5%	9.6%	3.0%	-	Jansen05
	ATW02	<2	76.3%	13.1%	3.9%	-	

- On average, users view less than two search result pages
- Over half of users do not access result beyond first page
- Relevance of top 10 search results is critical

Session Length

Region	Data	Avg	1	2	3	>3	Source
US	Excite97	1.6	67%	19%	7%	7%	Jansen00 Jansen01,
	Excite99	1.7	60.4%	19.8%	19.8%		Wolframe01
	Excite01	2.3	55.4%	19.3%	25.3%		Spink02 Spink02a
	AV98	2.02	77.6%	13.5%	4.4%	4.5%	Silverstein99
	AV02	~2	47%	-	-	-	Jansen06
Europe	FAST01	2.9	53%	18.9%	29%		Spink02a
	ATW01	3.0	52.9%	18.3%	9.4%	19.4%	Jansen05
	ATW02	2.8	58.7%	16.1%	7.9%	17.3	

- Average session length is around 2-3 queries
- More than half of sessions consist of only one query
- Europe sessions are longer than US sessions

Topic Distribution

Name	People & Place	Commerce	Health	Entertainment	Internet & Computer	Porn	Source
Excite97	6.7% (6)	13.3% (3)	9.5% (5)	19.9% (1)	12.5% (4)	16.8% (2)	Wolfram01
Excite99	20.3% (2)	24.4% (1)	7.8% (4)	7.5% (6)	10.9% (3)	7.5% (5)	Wolfram01
Excite01	19.7% (2)	24.7% (1)	7.5% (6)	6.6% (7)	9.6% (4)	8.5% (5)	Spink02,
AV02	49.3% (1)	12.5% (2)	7.5% (4)	4.6% (6)	12.4% (3)	3.3% (7)	Jasen06
FAST01	22.5% (1)	12.3% (3)	7.8% (6)	9.1% (5)	21.8% (2)	10.8% (4)	Spink02a
ATW01	22.5% (1)	12.3% (3)	7.8% (6)	9.1% (5)	21.8% (2)	10.8% (4)	Jansen05
ATW02	41.5% (1)	12.7% (3)	4.9% (5)	9.5% (4)	16.3% (2)	4.5% (6)	

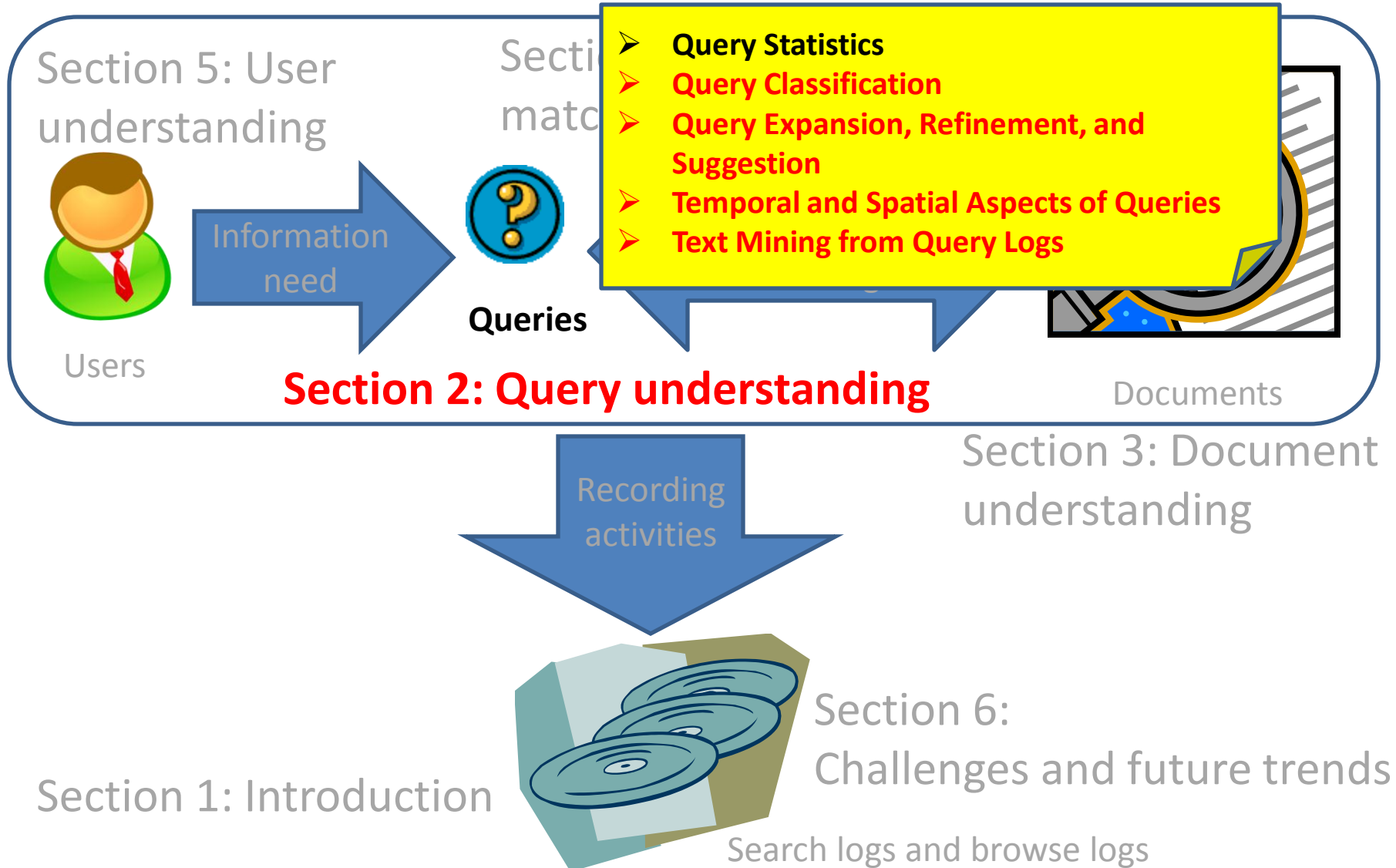
- Top six topics are same over time and across regions
- Percentage of individual topics change over time
 - In US, percentage of porn searches decreases, while percentages of commerce and people & place increase

Summary of Query Statistics

- Web search is quite different from traditional IR

	Traditional IR	Web search
Query length	6-9	2-3
Query frequency	Zipf distribution	Zipf distribution + skewed head and tail
Num. of viewed result page	~10	1-2
Session length	7-16	1-2
Topics	More focused	Diverse

A Road Map



Outline of Query Classification

- Overview of Query Classification
- Challenges in Query Classification
- Methods for Query Task Classification
- Methods for Query Topic Classification
- Summary of Query Classification

Overview of Query Classification

- Queries can be categorized on multiple dimensions
 - Task (navigational, informational)
 - Topics (ODP categories, auto-created concepts)
 - Entity and Attribute (e.g., ‘avatar game’)
 - Time-sensitiveness (e.g., ‘WWW conference’)
 - Location-sensitiveness (e.g., ‘pizza’)
 - Data Source (e.g., wiki, image, video)
- Intent of query can be represented by the categories
- Applications of query classification
 - Relevance ranking
 - Faceted search or categorized search
 - Online advertisement

Challenges in Query Classification

- Queries are
 - Usually very short
 - Often ambiguous
 - Meaning changes over time and location

Search Tasks

- High level task categories [Broder02]
 - Navigational: to reach particular site
 - Informational: to acquire some information assumed to be present on one or more web pages.
 - Transactional: to perform some web-mediated activity.
- Distribution
 - Varies according to different studies
 - Navigational: 20%, Informational: 48%, Transactional: 30%

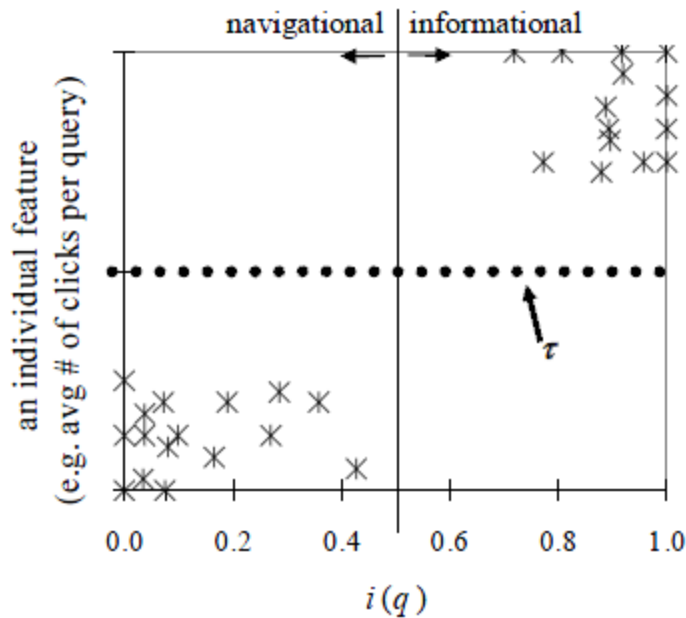
Methods for Query Task Classification

- Using web pages
 - [Kang03]
- Using click-through data and anchor text data
 - [Lee05]

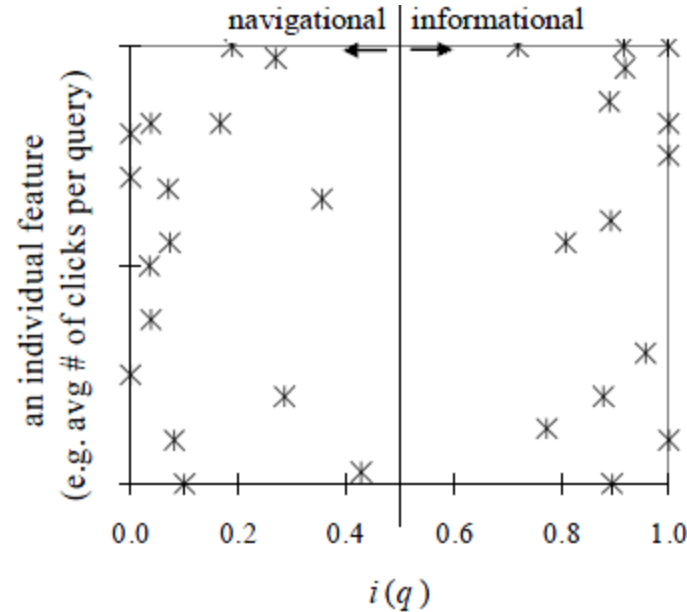
Query Task Classification Using Click-through and Anchor Text Data [Lee05]

- Only two categories considered, i.e., navigational and informational
- Basic idea
 - Navi query \Leftrightarrow click distribution is skewed
 - Navi query \Leftrightarrow anchor text distribution is skewed
- Method
 - Using mean, median, skewness, and kurtosis to characterize distributions of clicks and anchor texts
 - Linear combination of features
- Accuracy: 90%
- Challenge: difficult for tail queries

Result of Single Feature



An Effective Feature



An Ineffective Feature

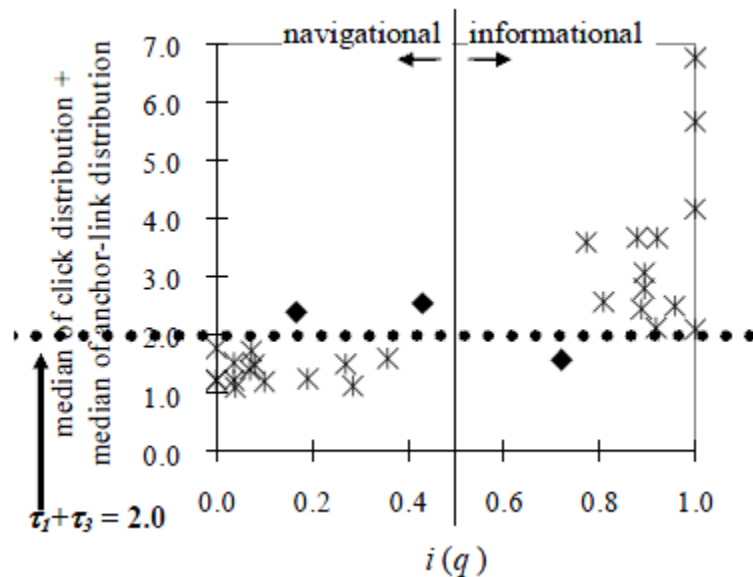
- 50 head queries labeled by 28 graduate students
- Each point represents query
- $i(q)$ is percentage of informational labels of query q
- A feature is effective if we can set horizontal bar (i.e., threshold) to separate navigational queries from informational queries

Result of Linear Combination

- Linear combination

$$f = w_1 \cdot f_1 + w_2 \cdot f_2 + \dots + w_n \cdot f_n$$

- A simple combination shows a better accuracy



- Combines two features
- Equal weights
- Accuracy reaches 90%

$$f = (\text{median of click distribution}) \\ + (\text{median of anchor distribution})$$

Search Topics

- ODP categories
- Automatically constructed concepts (clusters)
- Query can have multiple topics (is ambiguous)
 - e.g., ‘Jaguar’ [car][animal]

Methods for Query Topic Classification

- Directly applying text classification techniques
- Using search results of query [Shen05]
- Using search log data
 - Using query log data [Beitzel07]
 - Using click-through data [Fuxman07], [Li08]

Query Topic Classification Using Query Log Data [Beitzel07]

- Four methods of classification
 - Exact-match lookup
 - N-gram lookup
 - Perceptron
 - Selectional Preference
- Combination of four methods
 - exact-match lookup first, followed by the perceptron, 4-gram lookup, and selectional preferences
- Accuracy: F1 score = 0.25

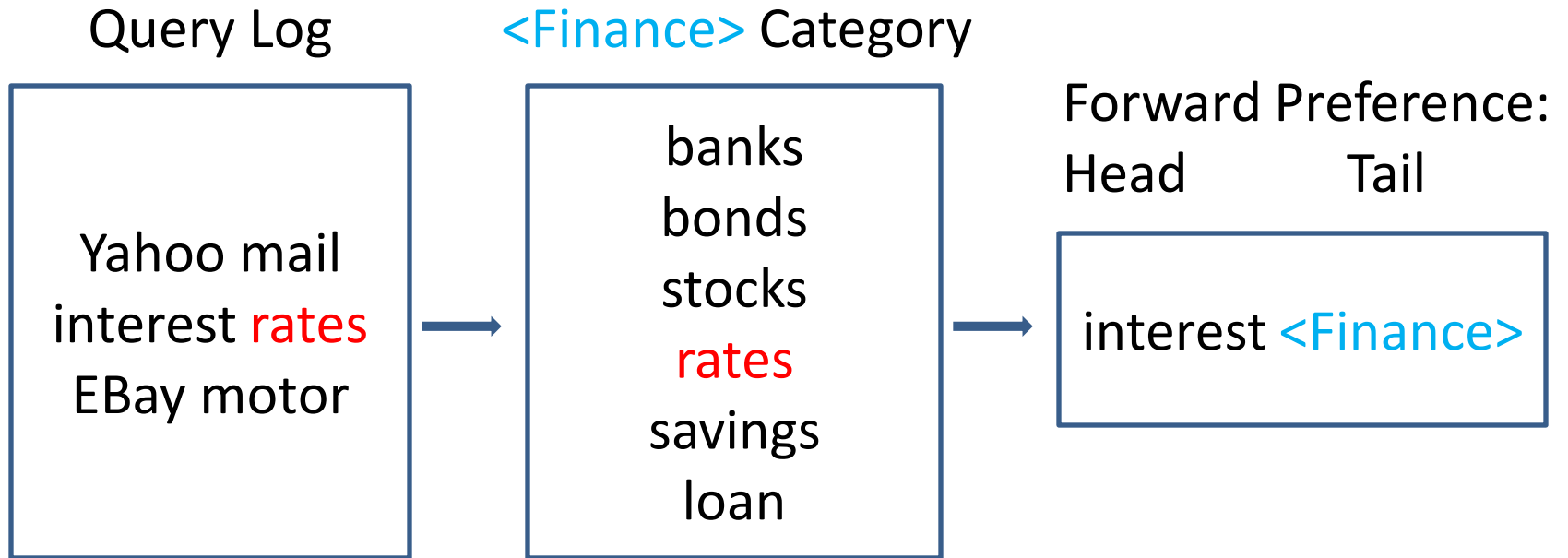
Selectional Preference: Step 1

- View query as pair of lexical units:
 - <head, tail>
 - Queries with n terms form $n-1$ pairs
 - Example: “directions to DIMACS” forms two pairs:
 - <directions, to DIMACS> and <directions to, DIMACS>
 - Only applicable to queries of 2+ terms

Selectional Preference: Step 2

- Manually label some words with categories
- Check head and tail of each pair to see if they appear in manually labeled set
- Convert each <head, tail> pair into:
 - <head, CATEGORY> (*forward* preference)
 - <CATEGORY, tail> (*backward* preference)

Selectional Preference: Step 2



Selectional Preference: Step 3

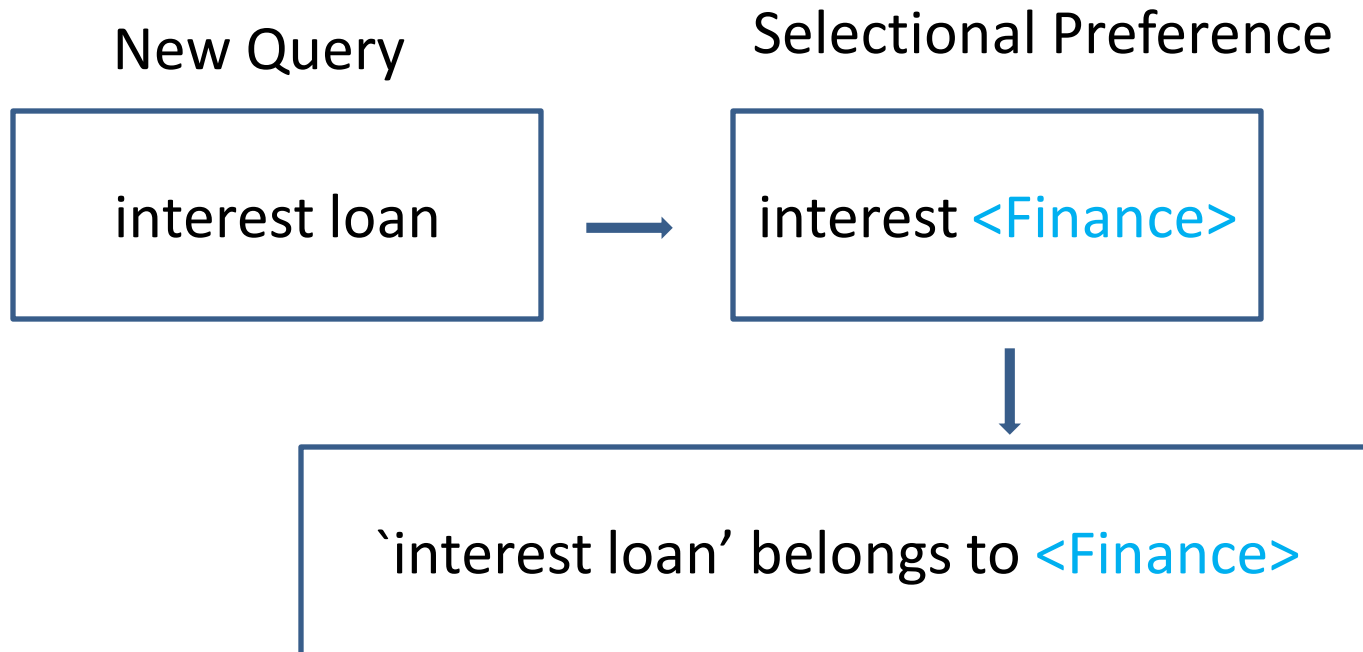
- Score each preference using Resnik's formula

$$S(u | x) = \max_u P(u | x) \log \frac{P(u | x)}{P(u)}$$

x denotes lexical unit and u denotes category of the other lexical unit

Selectional Preference: Step 4

- Use mined selectional preferences to assign categories to unseen queries



Query Topic Classification Using Click-through Data [Fuxman07]

- View click-through bipartite as undirected graph
- Define random walk model
- Probability on edge represents transition probability (calculated using click-through counts)
- Probability of node represents probability of belonging to class
- Propagate class labels on graph

Random Walk Algorithm

- Add 'null' node to the click through bipartite
 - Each node may walk to null node with probability α

- Iteration between two processes

- Estimate probability of query node

$$P(l_q = c) = (1 - \alpha) \sum_{(q,u)} P(q \rightarrow u) P(l_u = c)$$

- Estimate probability of URL node

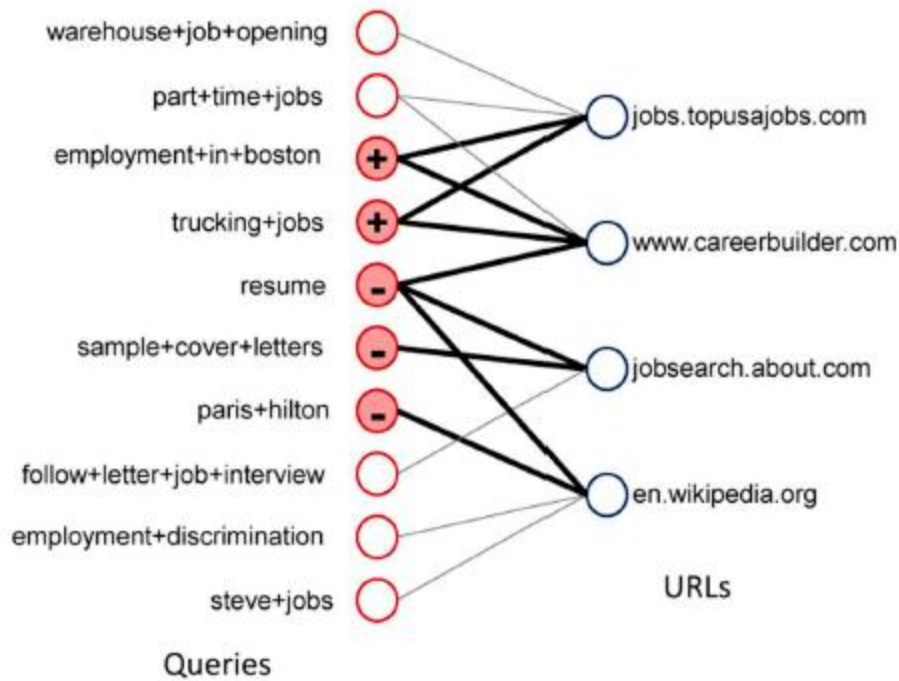
$$P(l_u = c) = (1 - \alpha) \sum_{(q,u)} P(u \rightarrow q) P(l_q = c)$$

- It is guaranteed to converge
- Analogy to electrical network.

Query Topic Classification Using Click-through Data [Li08]

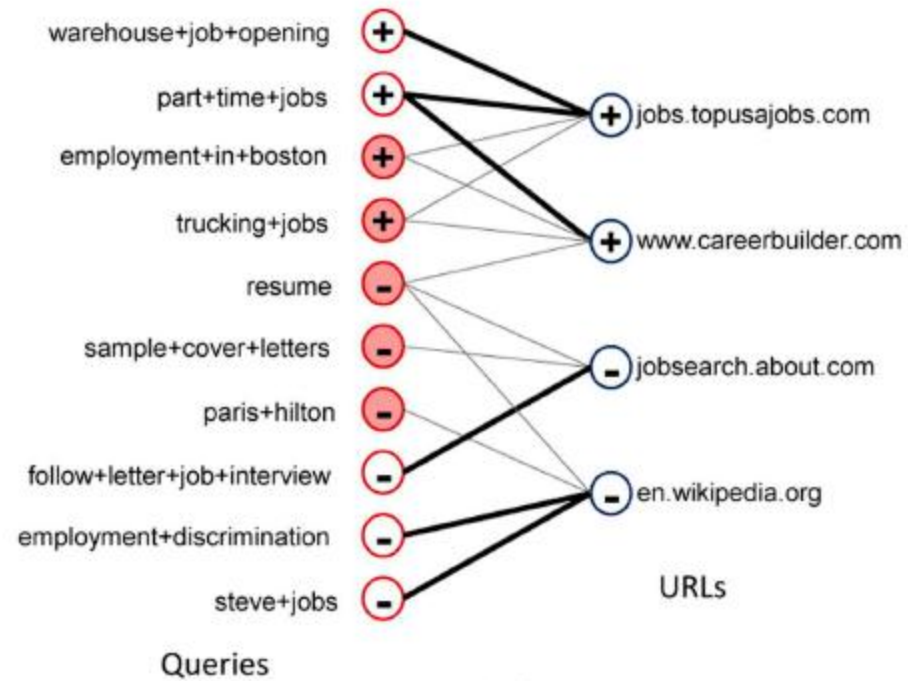
- Given a set of labeled queries (representing same topic)
- Train classifier based on content of queries
- Propagate class labels through click-through bipartite
- Iteratively combining content-based classification and click-based classification
- Accuracy: F score = 0.74 to 0.88

Propagation through Click-Through Bipartite



(a)

Labeled seeds



(b)

After propagation

Content-based Classifier

- Maximum Entropy Classifier

$$P_{\lambda}(y | x) = \frac{\exp(\sum_i \lambda_i \phi_i(x, y))}{\sum_y \exp(\sum_i \lambda_i \phi_i(x, y))}$$

x denotes query, y denotes query topic class, $\phi(x, y)$ denotes feature, λ denotes parameter

- Using n-grams of query or snippets of query as features

Click-based Classifier

- Let W be $m \times n$ matrix where $W[i, j]$ is click count on URL j for query i
- Let F be $m \times 2$ matrix where $W[i, j]$ is non-negative, real number indicate likelihood that query i belongs to class y
- Random walk converges to

$$F^* = (1 - \alpha)(1 - \alpha A)^{-1} F^0$$

where $A = D^{1/2} W W^T D^{-1/2}$, D is diagonal matrix in which element $d_{i,i}$ equals sum of elements in row i of $W W^T$

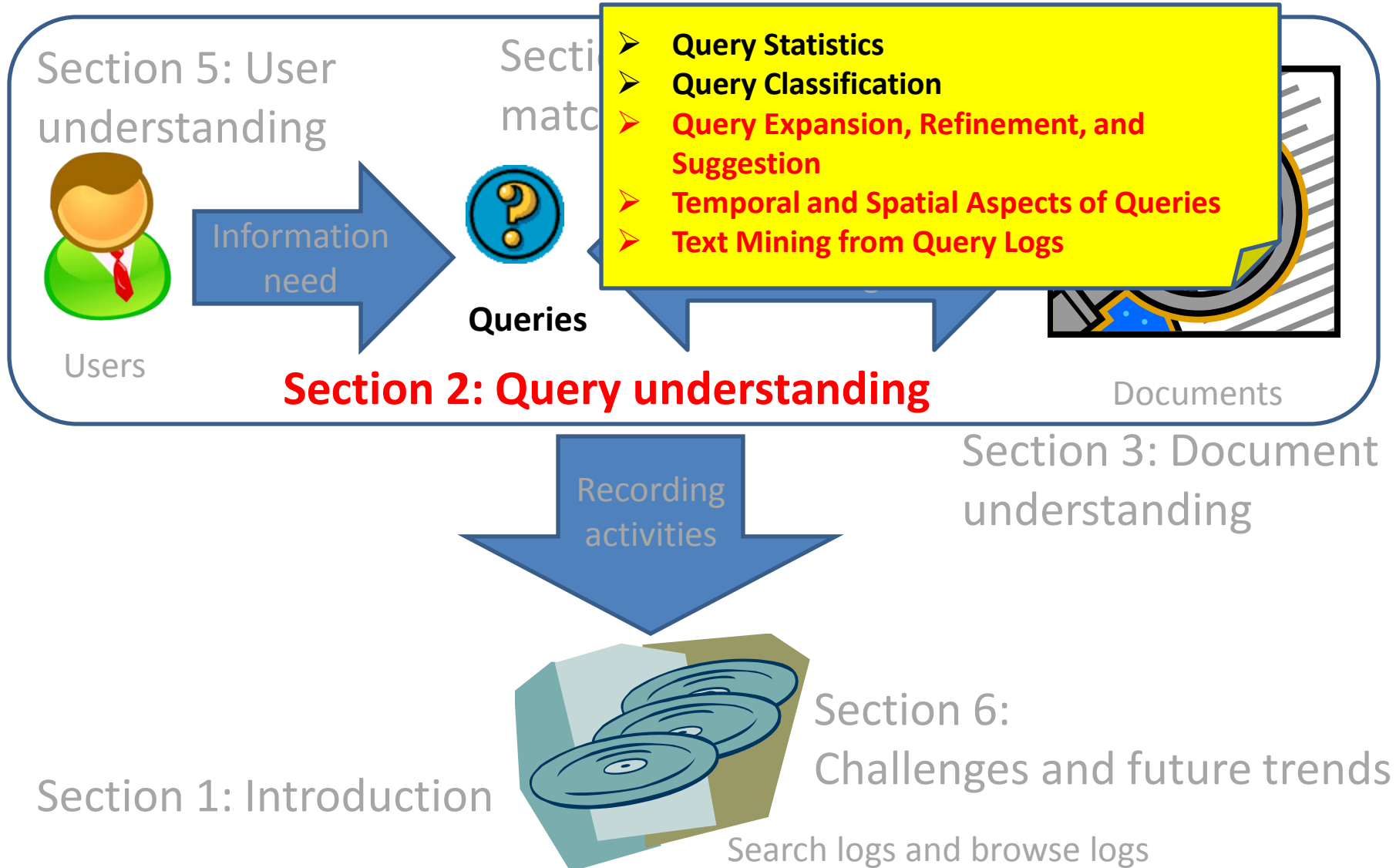
Combining Classifiers

- Step 1: initialize F^* by labeled seeds, initialize λ as random
- Step 2: repeat
 - Train λ^* of content-based classifier using classification results by current F^*
 - Train F^* of click-based classifier use classification results by current λ^*until converge

Summary of Query Classification

- Classify queries based on tasks
 - Using click distribution and anchor text distribution
- Classify queries based on topics
 - Using query log, exact match, selectional preference, etc
 - Using click-through data and random walk

A Road Map



Outline of Query Expansion, Refinement, and Suggestion

- Overview of Query Expansion
- Methods for Query Expansion
- Overview of Query Refinement
- Methods for Query Refinement
- Overview of Query Suggestion
- Methods for Query Suggestion
- Summary of Query Expansion, Refinement, and Suggestion

Overview of Query Expansion

- Re-write query to increase search recall
- Example: 'ny times' → 'ny times new york'
- Has been studied in IR from many years

Methods for Query Expansion

- Traditional approach
 - Global methods
 - Thesauri (e.g., Longman dictionary or WordNet)
 - Automatic thesaurus generation
 - Local methods
 - Explicit relevance feedback
 - Pseudo relevance feedback
 - Combination of global and local methods
- Using log data
 - Using click-through data [Cui02]
 - Using session data [Fonseca05]

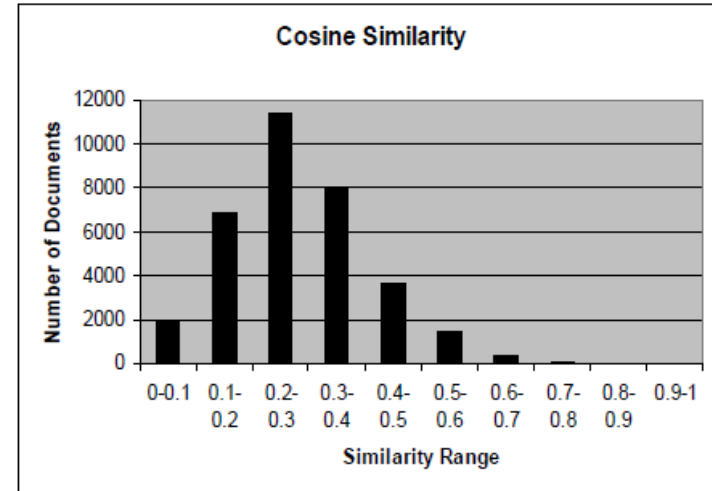
Query Expansion Using Click-Through Data

[Cui02]

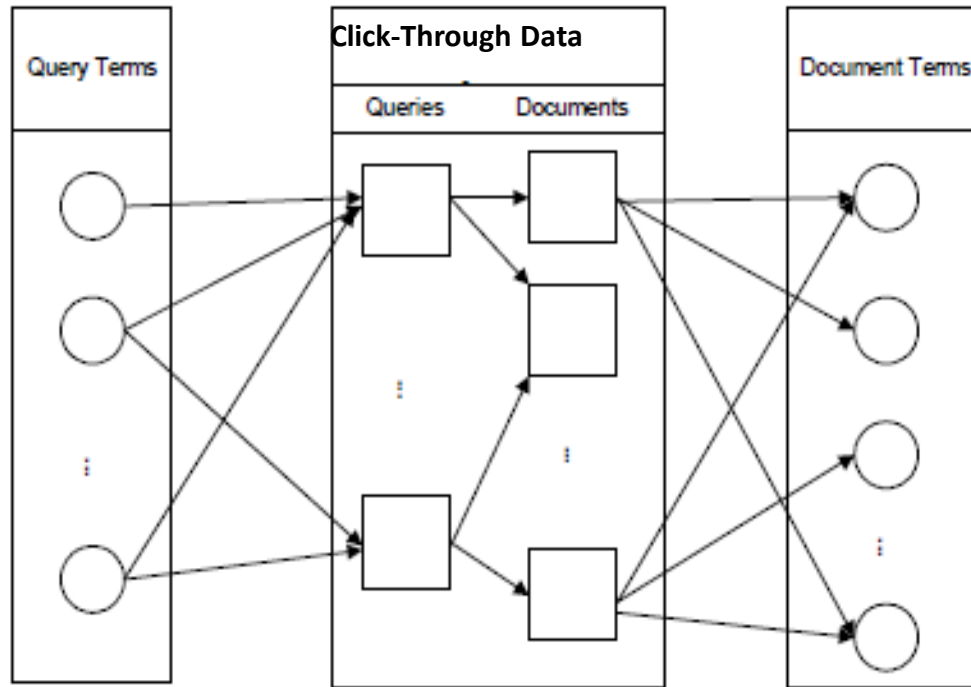
- There is gap between query space and document space
 - Web queries are often short and ambiguous
 - Users may not use the same terms appearing in documents as search keywords
- Query terms are linked to document terms by click-through data
 - If a set of documents is often linked to a set of queries, then the terms in the documents are strongly related to the terms of the queries

Gap between Query Space and Document Space

- Each document d is represented by
 - $W(d)$: terms in d
 - $W(q)$: queries for which document is clicked on
- Calculating cosine similarity between $W(d)$ and $W(q)$
 - Few documents have similarity values above 0.8
 - Average similarity value is 0.28
 - Large gap between two spaces



Mapping Query Terms to Document Terms



$$P(w_j^{(d)} | w_i^{(q)}) = \sum_{D_k} P(w_j^{(d)} | D_k) P(D_k | w_i^{(q)})$$

$w_j^{(d)}$ Document term $w_j^{(q)}$ Query term

Query Expansion by Term Correlations

- Given a query Q , calculate the weight for each term by

$$Weight(w_j^{(d)}) = \ln\left(\prod_{w_i^{(q)} \in Q} (P(w_j^{(d)} | w_i^{(q)}) + 1)\right)$$

- Use the top terms for expansion
- Example: top terms of query 'Steve Jobs'
 - Apple, personal computer, computer

Query Expansion Using Session Data

[Fonseca05]

- Offline part
 - Find all queries “associated” with query q
 - Group associated queries into “concepts”
- Online part
 - Given query q , find all concepts of q
 - Ask user to select concept
 - Expand q with the other queries in selected concept

Offline Step 1: Association Rule Mining

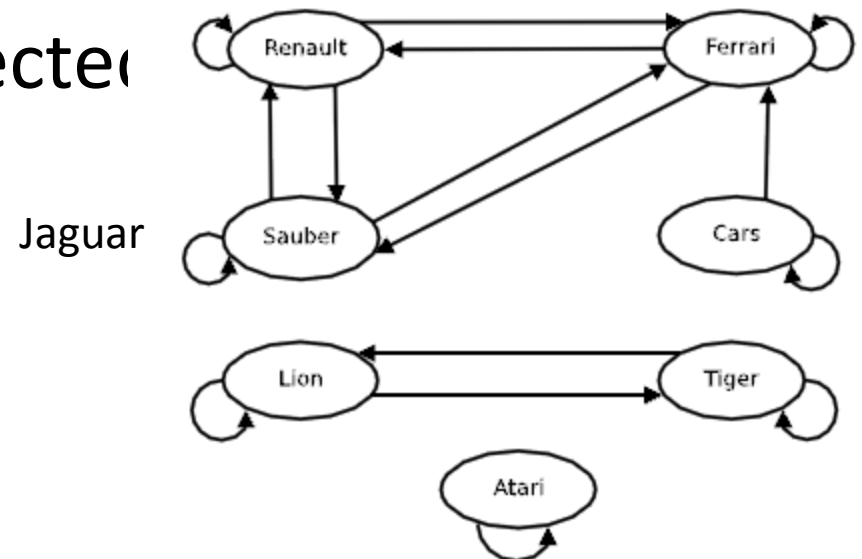
- Session data

Session-ids	Query sequences
S1	{Q _a , Q _b , Q _c }
S2	{Q _a , Q _b , Q _d }
S3	{Q _a , Q _b , Q _c , Q _d , Q _e }

- Mining length-2 frequent sequential patterns
 - Many methods in data mining
- Deriving association rules
 - For frequent pattern {Q_a, Q_b}, if confidence is greater than threshold, generate rule: Q_b → Q_a

Offline Step 2: Finding Concepts

- For each query Q_a , create query set R_a such that for any query $Q_i \in R_a$, rule $Q_i \rightarrow Q_a$ exists
- Build query relation graph G_a with respect to Q_a
 - Each query $Q_i \in R_a$ is vertex
 - Two queries $Q_i, Q_j \in R_a$ are connected with directed edge from Q_i to Q_j if there is rule $Q_j \rightarrow Q_i$
- Concept = strongly connected



Online Part

- Given query Q , return concepts to user
- Ask user
 - Which concept she is interested in
 - Type between Q and selected concepts
- Expand query with terms in selected concept
 - Take different approaches if query-concept type is specified

Overview of Query Refinement

- Reformulate query to better represent search intent
 - Spelling error correction: 'machin learning' → 'machine learning'
 - Stemming
 - Acronym expansion
- Challenges: mapping from X to Y, huge spaces
 - 'papers on machin learn' → 'paper on machine learning'

Methods for Query Refinement

- Spelling error correction:
 - Maximum Entropy Model [Li06]
 - Source Channel Model [Cucerzen04]
- Unified and discriminative model learned from query log
 - CRF Model [Guo08]

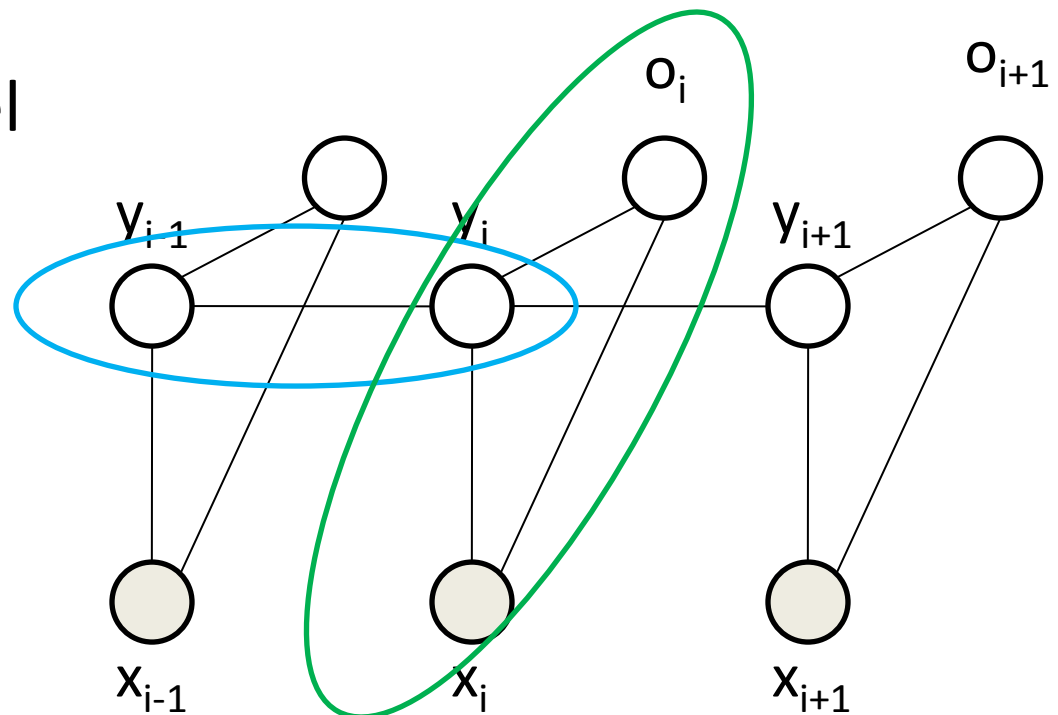
Query Refinement Using Conditional Random Fields Model

- View query refinement as mapping from space of original queries X to space of refined queries Y
- Directly using $P(y/x)$ is not practical as X and Y are huge
- Employ model $P(y, o/x)$ where O denotes operations, reduce the output space
- Define $P(y, o/x)$ as CRF
- Multiple layers of CRF is employed

Conditional Random Fields for Query Refinement [Guo08]

$$\Pr(\mathbf{y}, \mathbf{o} | \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \exp\left(\sum_{i=1}^n \left(\sum_k \lambda_k f_k(y_{i-1}, y_i) + \sum_k \lambda_k h_k(y_i, o_i, \mathbf{x})\right)\right)$$

Basic CRF-QR model

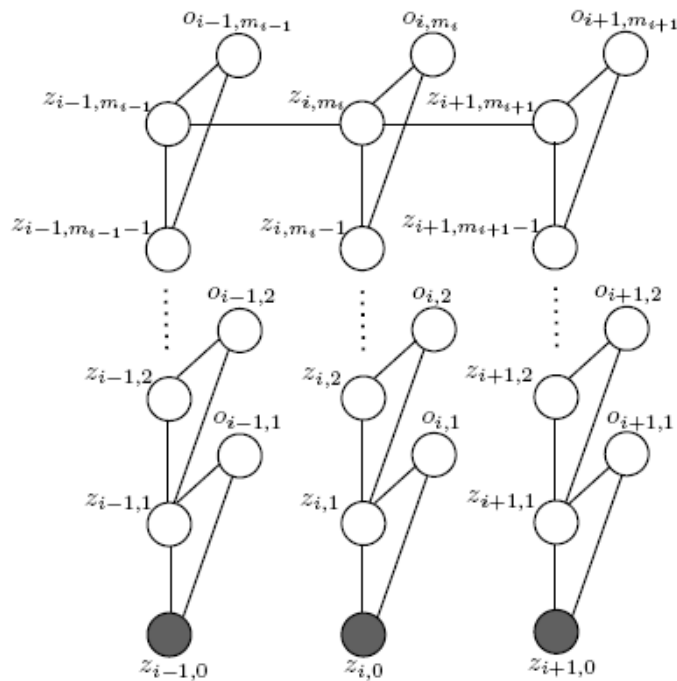


Refinement Operations

Task	Operation	Description
Spelling Error Correction	Deletion	Delete a letter in the word
	Insertion	Insert a letter into the word
	Substitution	Replace a letter in the word with another letter
	Transposition	Switch two letters in the word
Word Splitting	Splitting	Split one word into two words
Word Merging	Merging	Merge two words into one word
Phrase Segmentation	Begin	Mark a word as beginning of phrase
	Middle	Mark a word as middle of phrase
	End	Mark a word as end of phrase
	Out	Mark a word as out of phrase
Word Stemming	+s/-s	Add or Remove suffix '-s'
	+ed/-ed	Add or Remove suffix '-ed'
	+ing/-ing	Add or Remove suffix '-ing'
Acronym Expansion	Expansion	Expand acronym

CRF-QR Extended Model

Multiple Refinement Tasks



$$\Pr(\mathbf{y}, \vec{o}, \vec{z} | \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{i=1}^n (\phi(y_{i-1}, y_i) \prod_{j_i=1}^{m_i} \phi(z_{i,j_i}, o_{i,j_i}, z_{i,j_i-1}))$$

Query Suggestion

- Suggest queries in two types
 - Same search intent, better form
 - Related searches
- Methods
 - Using click-through data
 - Using session data
 - Context aware query suggestion [Cao08]

Methods Using Click-Through Data

- Use similar queries as suggestions for each other
- Measure similarity of queries
 - Overlap of clicked document [Beeferman00], [Wen01]
 - Similarity of category or content of clicked documents, [Wen01], [Yates04],
- Cluster queries
 - Agglomerative hierarchical method [Beeferman00]
 - DBScan [Wen01]
 - K-means [Yates04]

Methods Using Session Data

- Co-occurrence or adjacency in sessions
 - If Q_a and Q_b often co-occur in the same session, they can be suggestions for each other
 - If Q_b often appear immediately after Q_a in the same session, Q_b is a suggestion for Q_a
- Measures to represent correlation between Q_a and Q_b
 - Number of sessions where Q_a and Q_b co-occur (or are adjacent) [Jensen06][Huang03][Jones06]
 - Mutual information, Weighted mutual information [Jensen06]
 - Jaccard similarity, dependency, cosine similarity [Huang03]
 - Log likelihood ratio [Jones06]

Context-Aware Query Suggestion

- User raises query “*gladiator*”



History?



People?



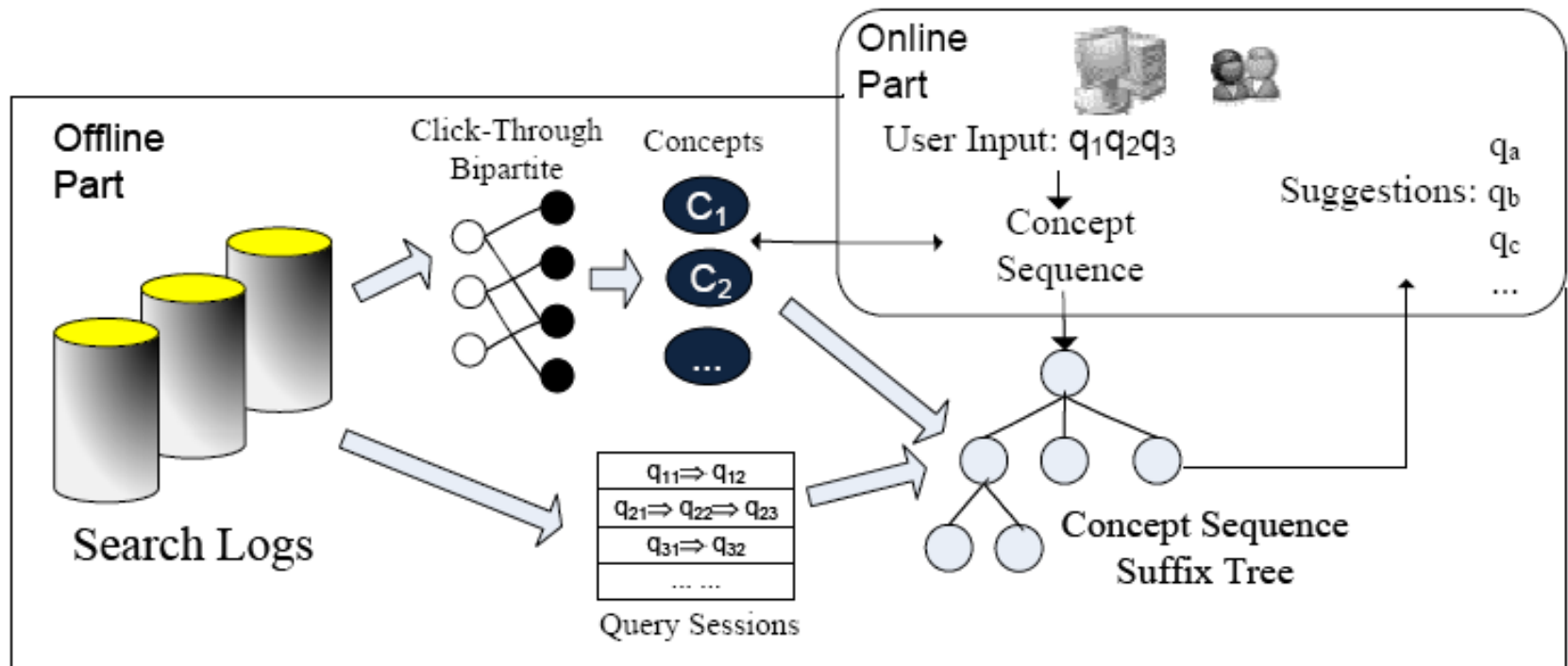
Film?

- If user raises query “*beautiful mind*” before “*gladiator*”
- Then user is likely to be interested in the film

Context-Aware Query Suggestion

- A naïve formulation
 - Given user query q_n
 - Find sequence of queries $q_1 \dots q_{n-1}$ submitted by users immediately before q_n
 - Scan log data and find out that in the same context $q_1 \dots q_{n-1}$, what queries people often ask after q_n
 - Output results as query suggestion
- Challenges: data sparseness, large scale

Method of Context-Aware Query Suggestion [Cao08]

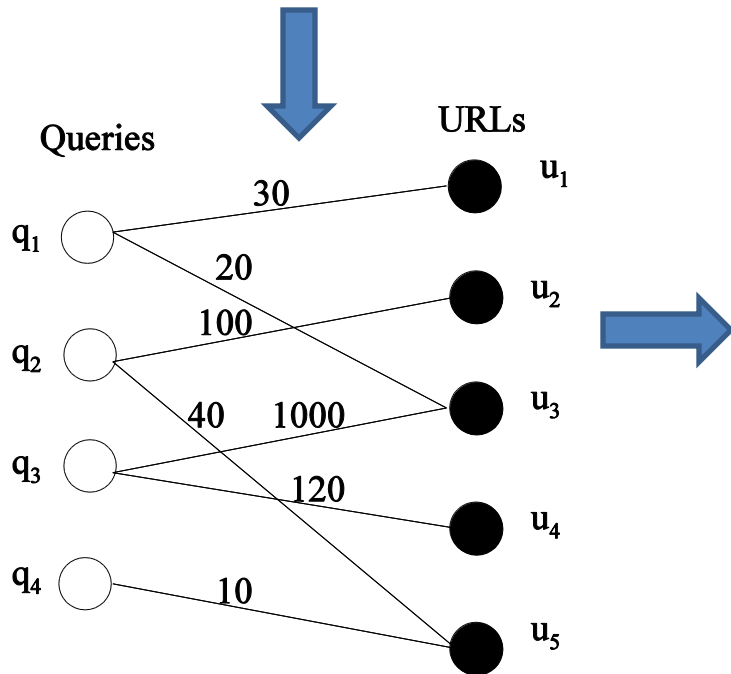


- Offline part: model learning
 - Summarizing queries into concepts by clustering queries on click-through bipartite
 - Mining frequent patterns from session data and building concept sequence suffix tree
- Online part: query suggestion

Finding Concepts from Click-Through Data

Search log

User ID	Time Stamp	Event Type	Event Value
User 1	2007-12-05 11:08:43	QUERY	KDD 2008
User 2	2007-12-05 11:08:45	CLICK	www.aaa.com
User 1	2007-12-05 11:08:48	CLICK	www.kdd2008.com
...



click-through bipartite

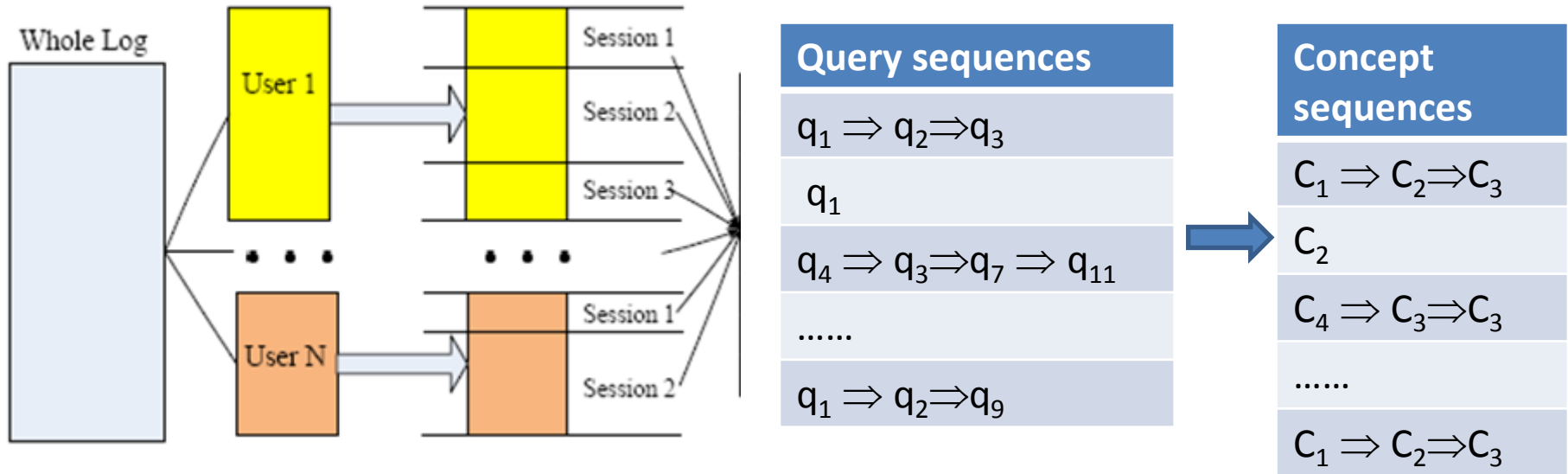
Query is represented by feature vector of URLs.

$$\vec{q}_i[j] = \begin{cases} \frac{w_{ij}}{\sqrt{\sum_{\forall e_{ik}} w_{ik}^2}} & \text{if } e_{ik} \text{ exists} \\ 0 & \text{otherwise} \end{cases}$$

Distance between two queries.

$$\text{dist}(q_i, q_j) = \sqrt{\sum_{u_k} (\vec{q}_i[k] - \vec{q}_j[k])^2}$$

Finding Concept Sequences from Session Data



Examples of query sequences

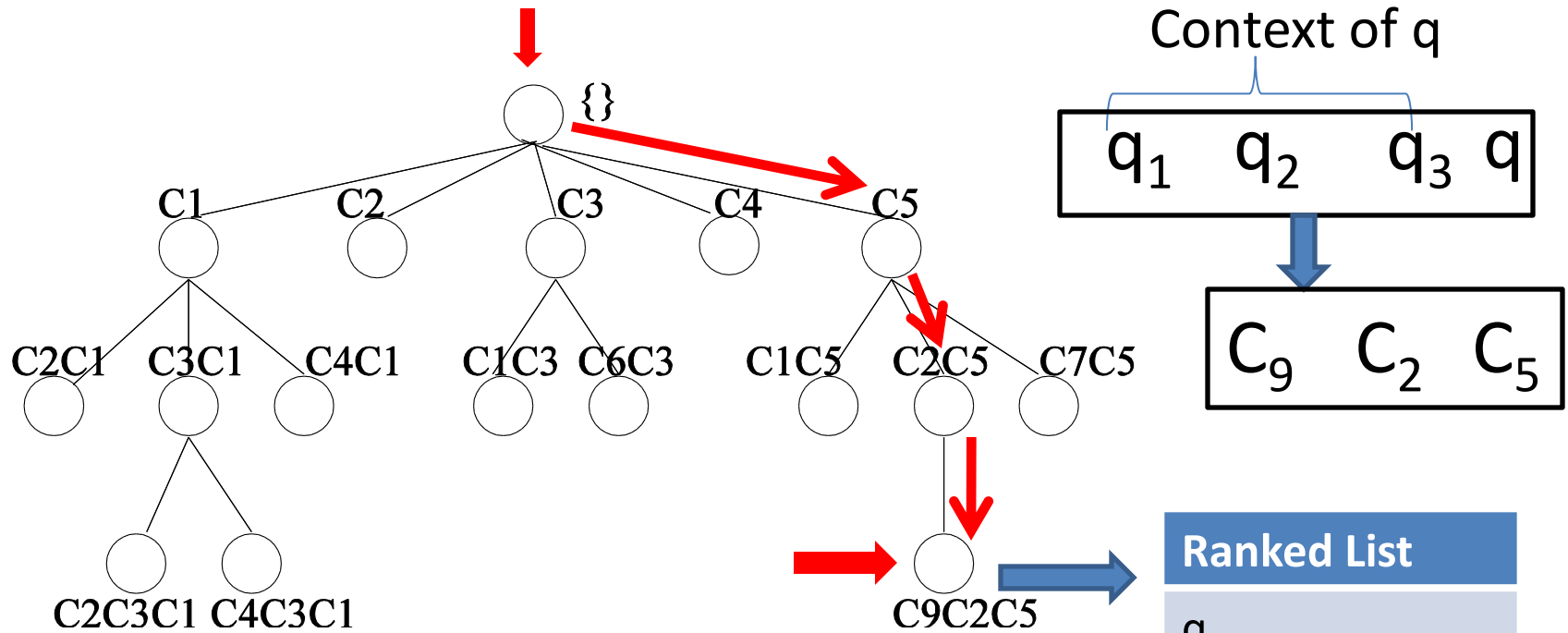
Query sequences

SMTP \Rightarrow POP3

BAMC \Rightarrow Brooke Army Medical Center

Nokia N73 \Rightarrow Nokia N73 themes \Rightarrow free themes Nokia N73

Building Concept Sequence Suffix Tree

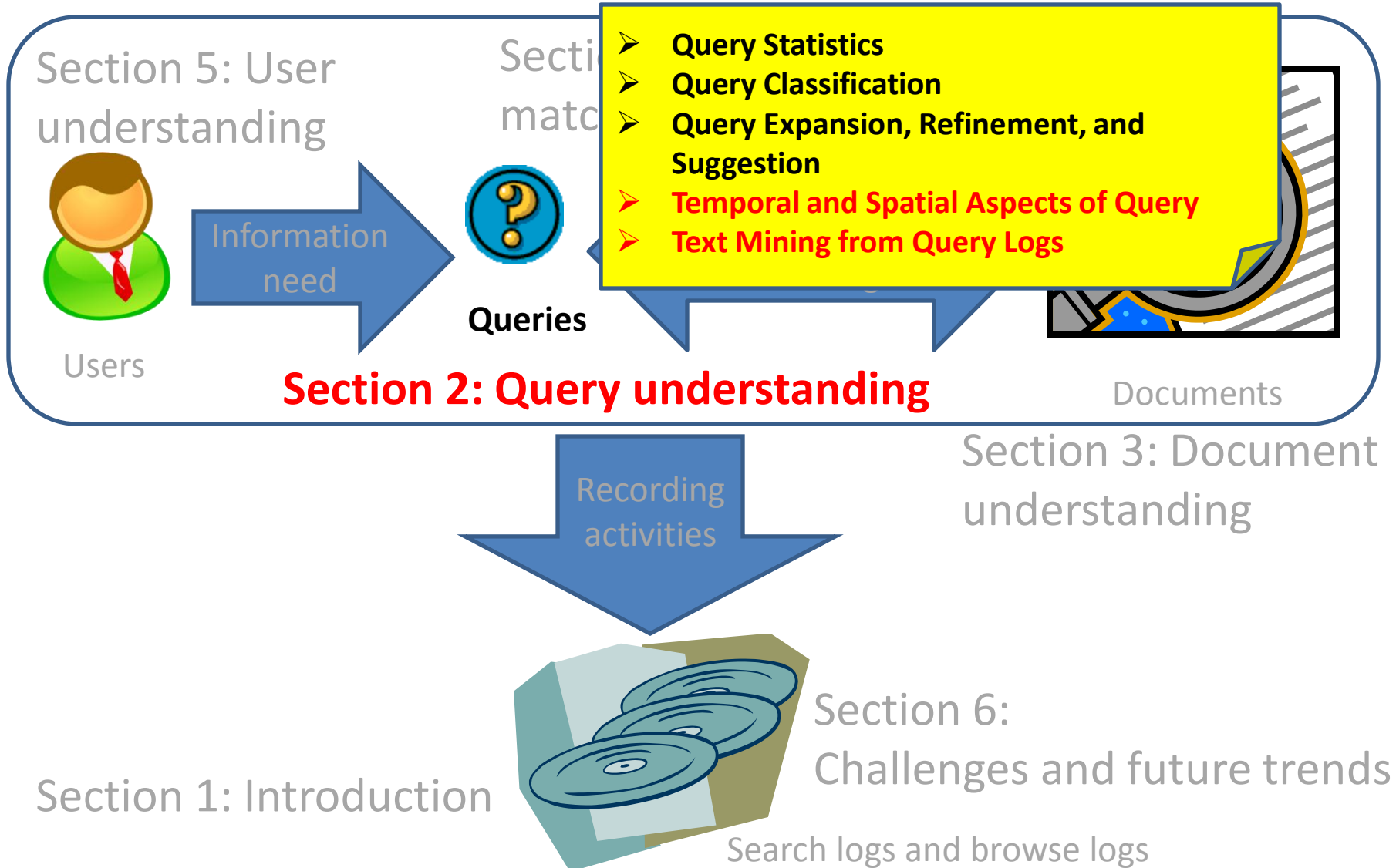


- Each node is concept sequence and associated with ranked list of query suggestions
- Each parent node is maximal suffix of its child nodes

Summary of Query Expansion, Refinement, and Suggestion

- Different methods to help users to search
 - Query expansion, refinement, and suggestion
- Click-through data and session data are useful for query expansion, refinement, and suggestion
- Using click-through data
 - Finding similar queries based on co-clicks
- Using session data
 - Finding frequent co-occurring or adjacent queries

A Road Map



Outline of Temporal and Spatial Aspects of Queries

- Overview of Temporal Aspect of Queries
- Analysis of Query Temporal Trends
- Query Temporal Models
- Summary of Temporal Aspect of Queries
- Overview of Spatial Aspect of Queries
- Summary of Spatial Aspect of Queries

Overview of Temporal Aspects of Queries

- Temporal trends of queries
 - How queries change over time
- Query temporal models
 - Periodic query identification
 - Burst query identification

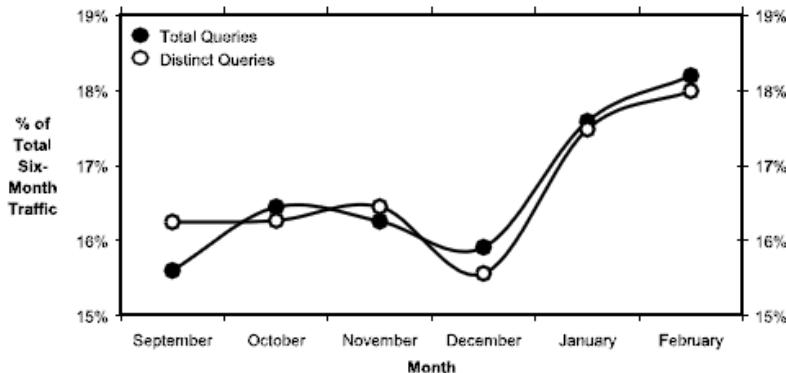
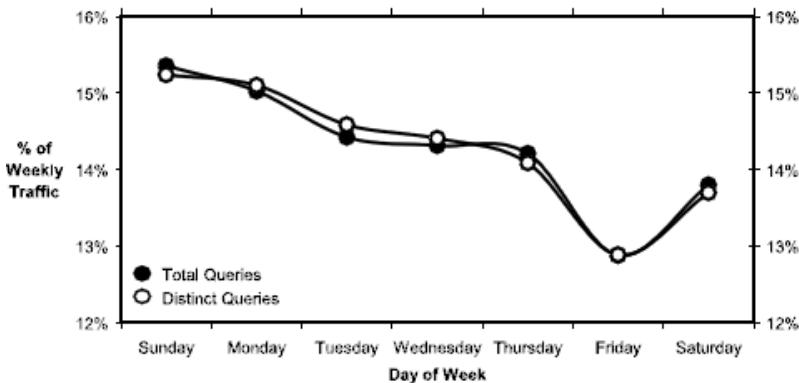
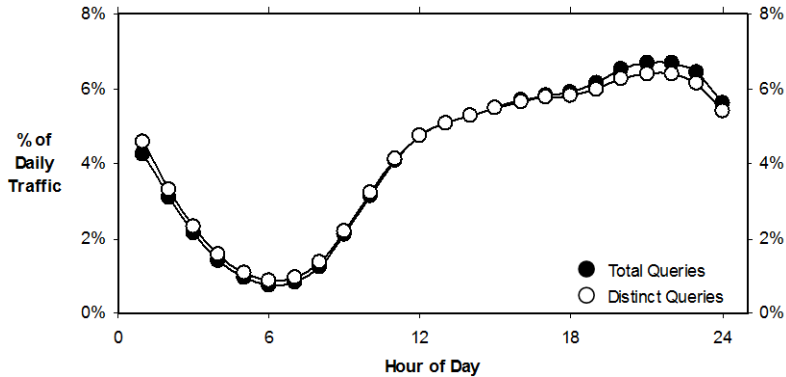
Analysis of Query Temporal Trends

- Examine several aspects of query stream over time (hourly, daily, and monthly):
 - Query volume: overall and by category
 - Query type: overall and by category
- **Result of temporal trend analysis: [Beitzel07]**
- Applications
 - Query classification
 - Caching strategy

Query Log Data

- Analyzed two AOL search logs:
 - One week of queries in December, 2003
 - Six months of queries: Sept. 2004-Feb. 2005
- Light pre-processing was done:
 - Case differences, punctuation, & special operators removed; whitespace trimmed
- Basic statistics:
 - Queries average 2.2 terms in length
 - Only one page of results is viewed 81% of the time; Two pages: 18%; Three or more: 1%
 - Consistent with previous studies

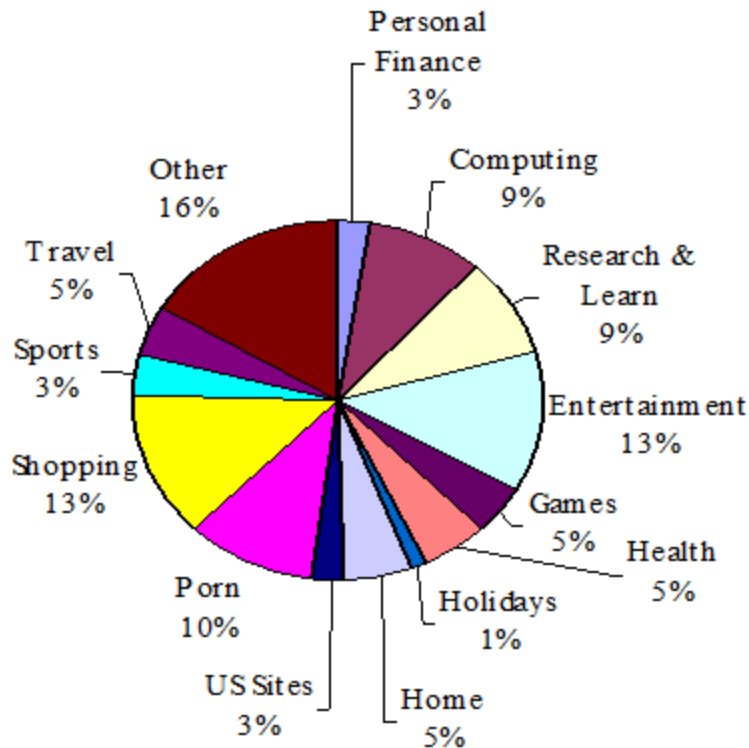
Traffic Volume Over Time



- Temporal trend of overall query volume
 - Hourly: 5-6 am lowest, 9-10 pm highest
 - Daily: drastic drop on Friday
 - Monthly: may be influenced by many other factors
- Trend of total queries matches with that of distinct queries
- Trend over months may be influenced by other factors than time

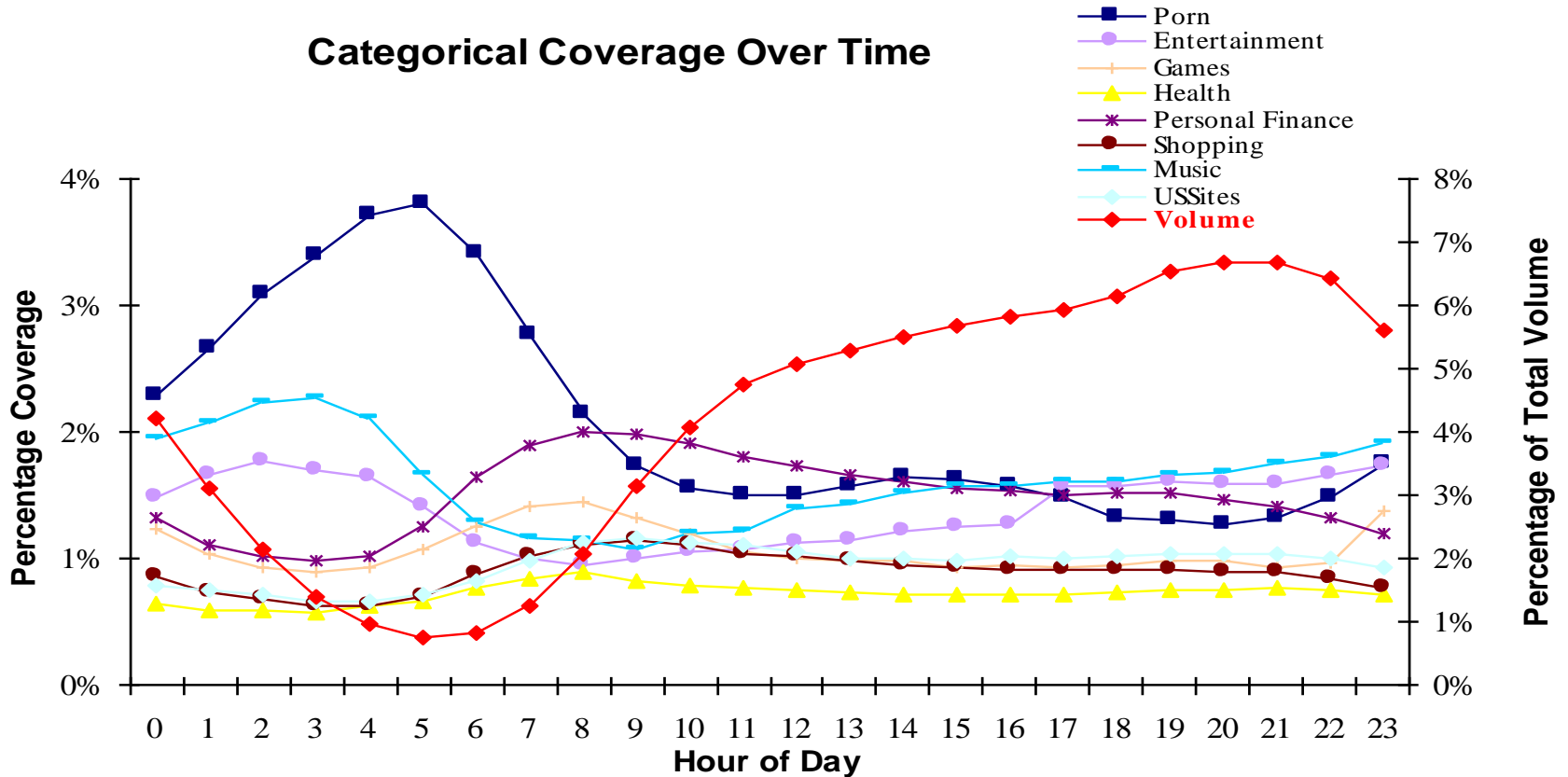
Category Breakdown

Sampled Categorized Query Stream Breakdown



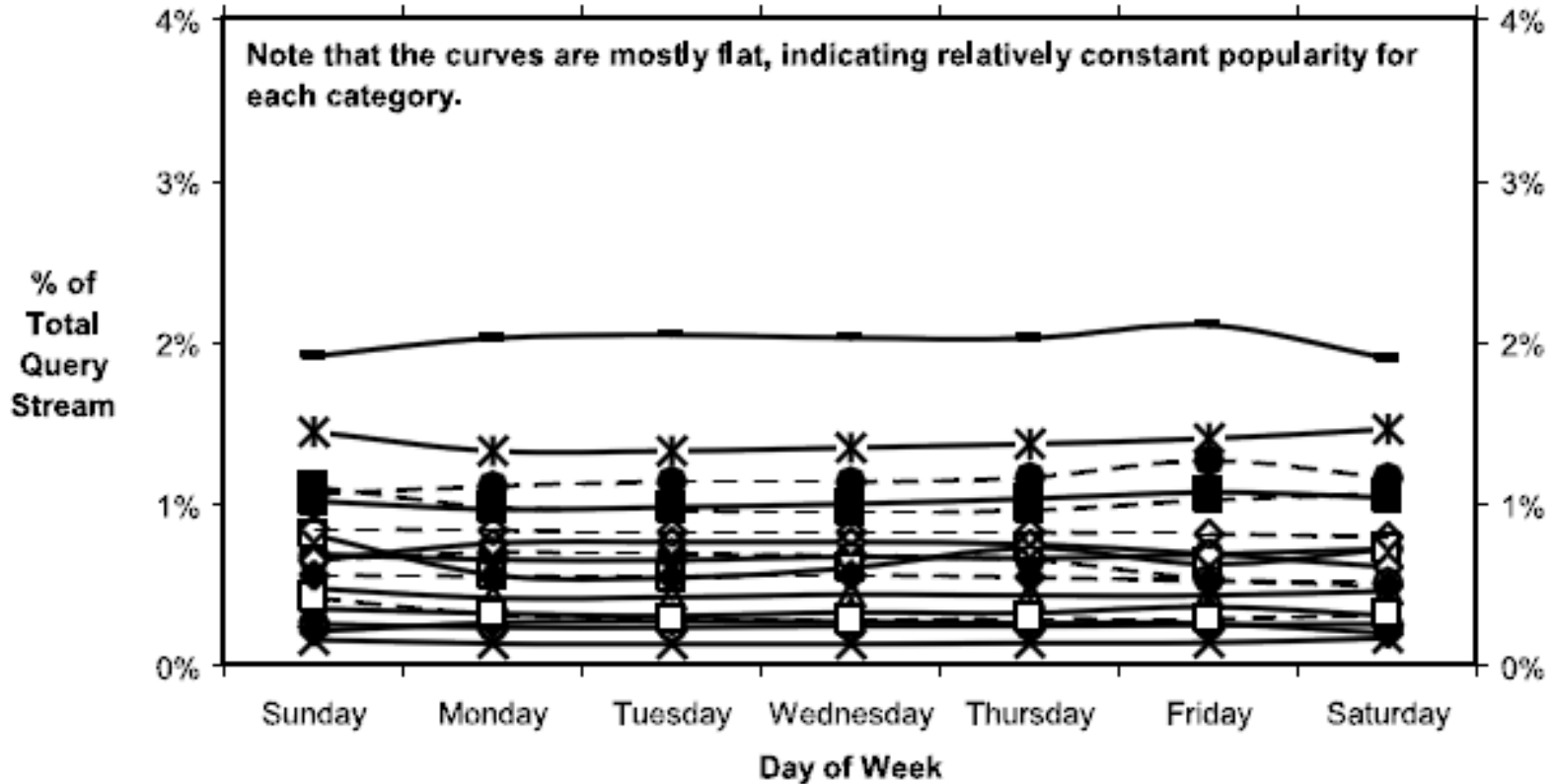
- Query lists for each category created by human editors
- Query stream classified by exactly matching each query to category lists
- Cover 13% of total query traffic

Category Popularity Over A Day



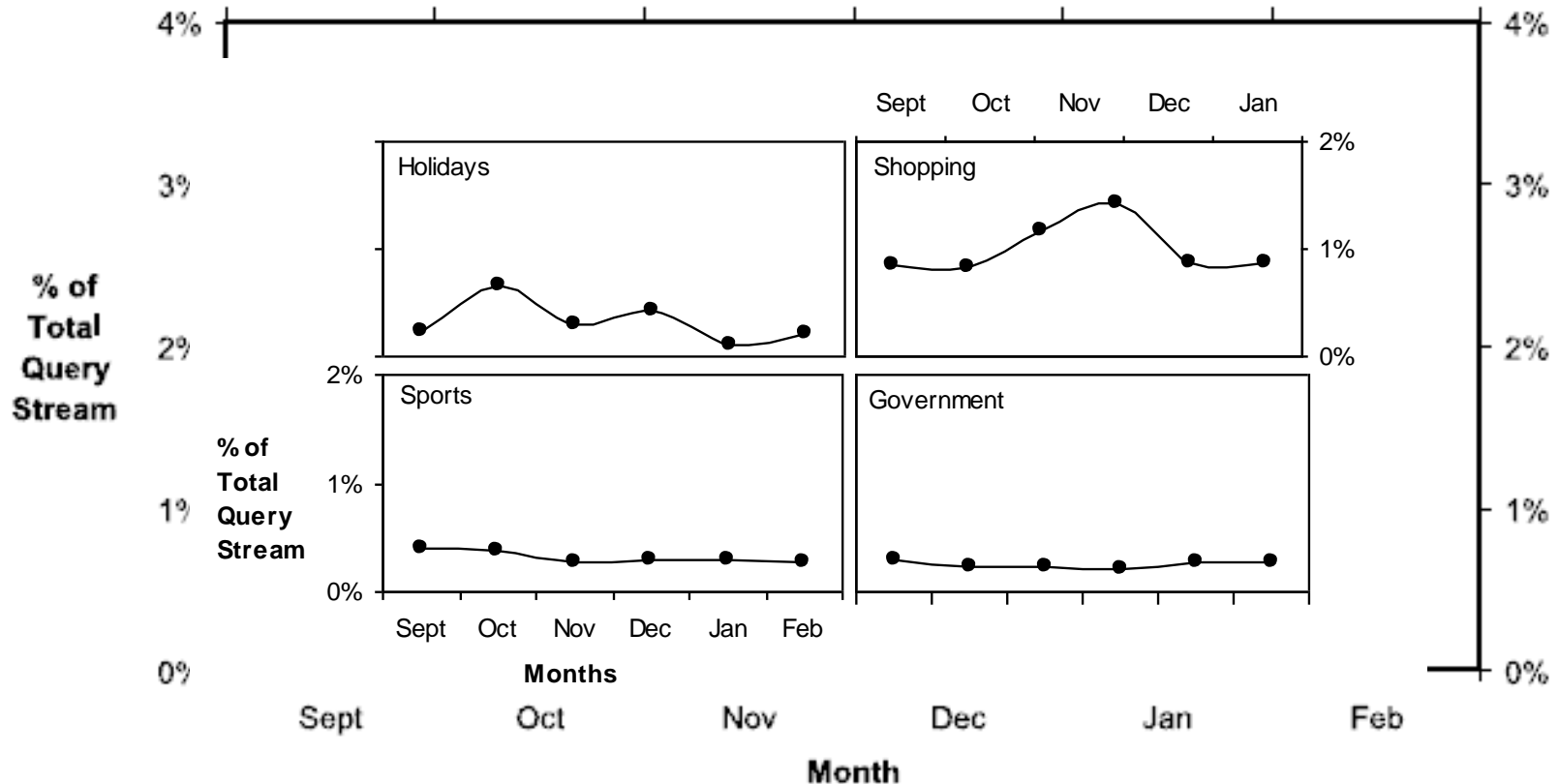
Hourly: some topical categories (e.g., entertainment) vary substantially more than others

Category Popularity Over Week



Daily: categories are relatively stable

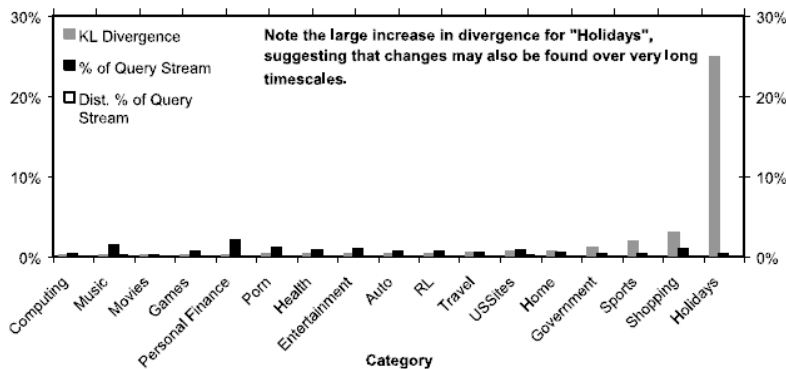
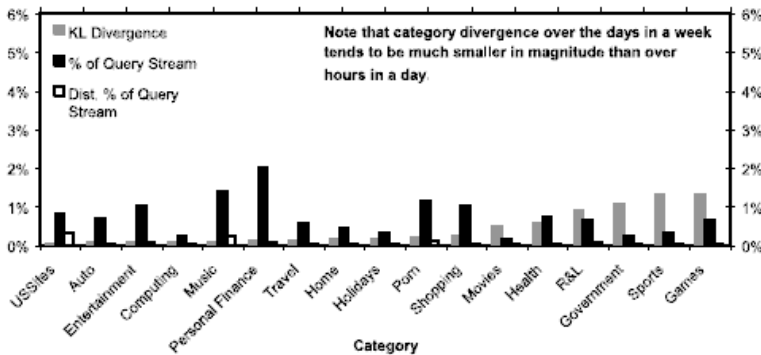
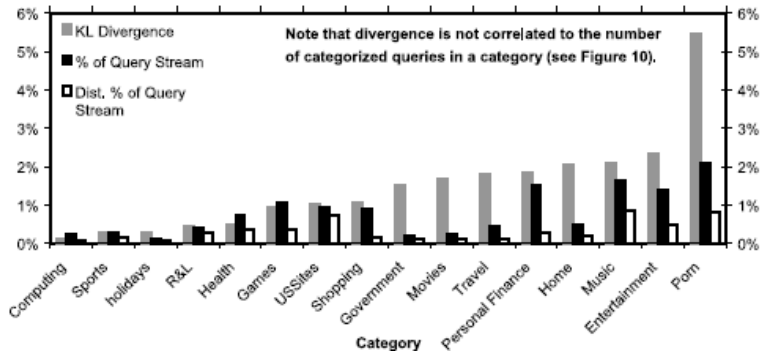
Category Popularity Over Six Months



Monthly: most categories are stable, while some others show seasonal changes (sports, holidays)

KL Divergence for Categories

- Whether distribution of queries within each category change over time



$$KL(p(q|t)||p(q|c,t)) = \sum_q p(q|t) \log \frac{p(q|t)}{p(q|c,t)}$$

- Categories with largest variance
 - Hourly: porn, entertainment, music, home
 - Daily: games, sports, government, research and learn
 - Monthly: holidays, shopping, sports, government

Query Temporal Models

- Query temporal models
 - Find periodic queries [Vlachos04]
 - Find burst queries [Dong10]
 - Find temporal relations between queries [Chien05]
- Temporal models are useful
 - Search results ranking (ranking based on recency, ranking based on periods)
 - Online advertisement

Periodic Query Identification Using Discrete Fourier Transformation [Vlachos04]

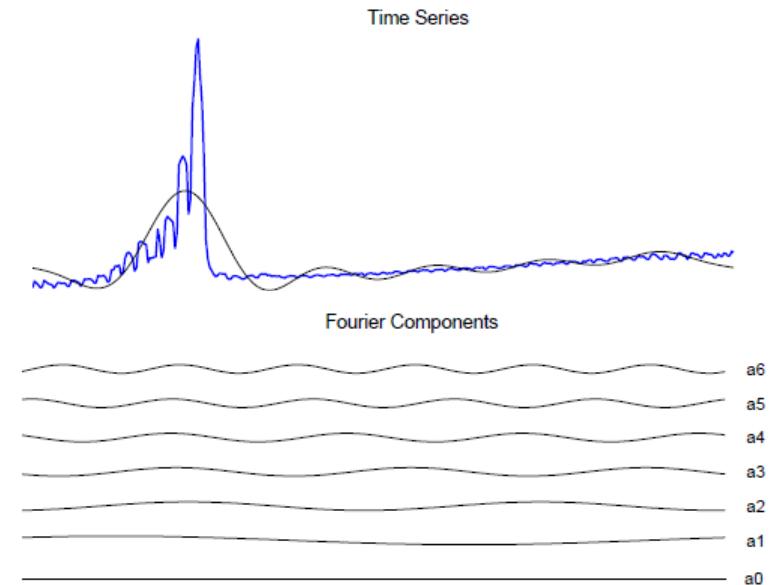
- Consider query stream as time series
- Represent time series as linear combination of complex sinusoids

$$X(f_{k/N}) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x(n) e^{-\frac{j2\pi kn}{N}}, k = 0, \dots, N - 1$$

- Find top K coefficients with highest magnitudes
- Represent power of each frequency

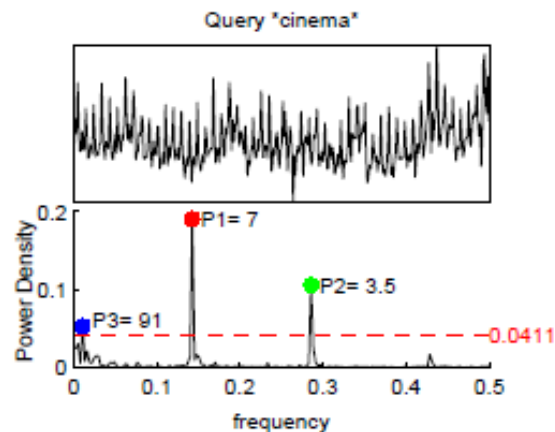
$$P(f_{k/N}) = \|X(f_{k/N})\|^2, k = 0, 1, \dots, \left\lfloor \frac{N-1}{2} \right\rfloor$$

- Power spectrum for query

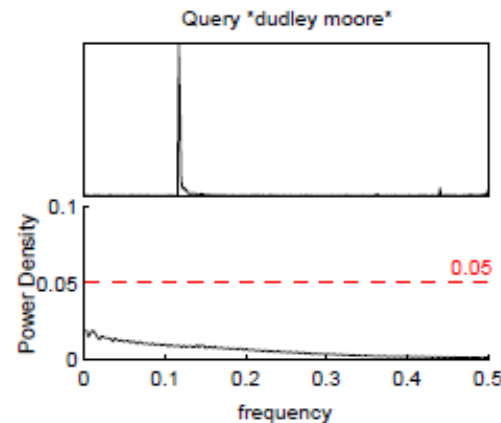


Periodic Queries vs Non-Periodic Queries

- Time periods can be found from power spectrum (period = $1 / \text{frequency}$)
- Power spectrums of random (non-periodic) queries follow exponential distribution
- Hypothesis testing: find significant periods of query using power spectrum of query



Most significant period for query "cinema" is 7



No significant period found for query "dudley moore"

Burst Query Identification Using Language Model [Dong10]

- Calculating probabilities for generating query at current time slot $P(q | M_{C,t})$ $P(q | M_{Q,t})$

- Calculating probabilities for generating query from previous time slot to current time slot

$$P(q | M_{C,t-r_i}) \quad P(q | M_{Q,t-r_i})$$

- Calculating buzziness of query from two language models and linearly combining them

$$\text{buzz}(q,t,C) = \max_i \log P(q | M_{C,t}) - \log P(q | M_{C,t-r_i})$$

$$\text{buzz}(q,t,Q) = \max_i \log P(q | M_{Q,t}) - \log P(q | M_{Q,t-r_i})$$

Summary of Temporal Aspect of Queries

- Analysis of Query Temporal Trends
- Periodic Query Identification
- Burst Query Identification

Overview Spatial Aspects of Queries

- Queries can be modeled from location perspective
- Two types
 - Local interest queries [Backstrom08]
Queries only interested by users at particular location
e.g., name of local high school, newspaper
 - Localizable queries [Welch08][Yi09]
Users at different locations may issue the same query,
but referring to different things
e.g., pizza hut, house for rent

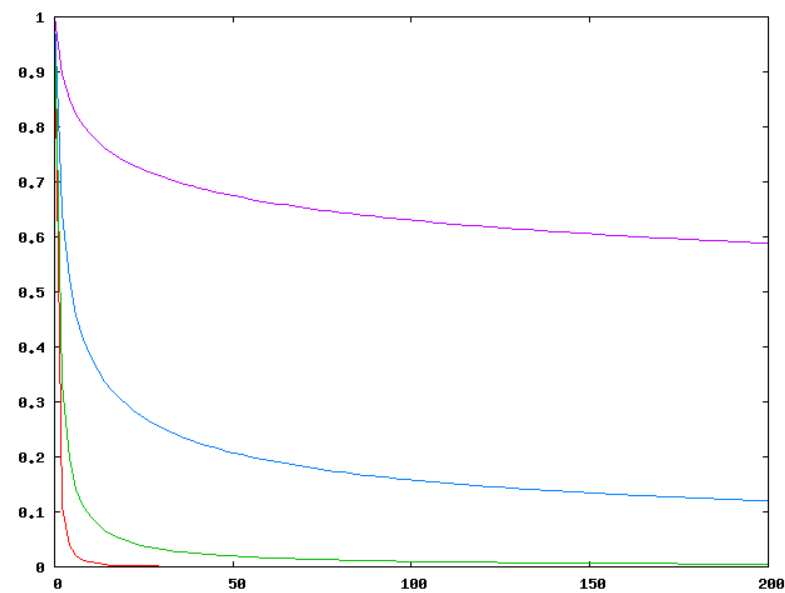
Modeling Local Interest Queries

[Backstrom08]

- Identify and characterize geo-features of topics
 - Find center of geographic focus for topic
 - Determine if topic is tightly concentrated or spread diffusely geographically
- Answer two types of question
 - Given query, what is center and dispersion?
 - Given region, what are local queries?
- Applications
 - Business intelligence
 - Re-ranking search results

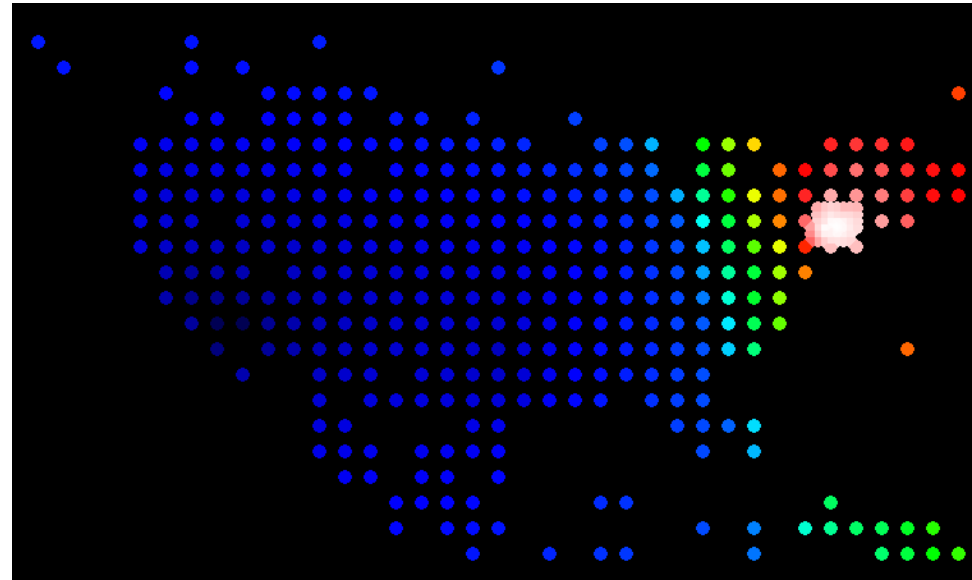
Probabilistic Model

- Consider query topic t
 - e.g. ‘red sox’
- For each location x , query coming from x has probability p_x with respect to t
- There exists center z .
 - Probability is highest at z
 - p_x is decreasing function of $||x-z||$
- Probability density function:
 - Query coming from x has probability $p_x = C d^{-\alpha}$
 - Ranges from non-local ($\alpha = 0$) to extremely local (large α)

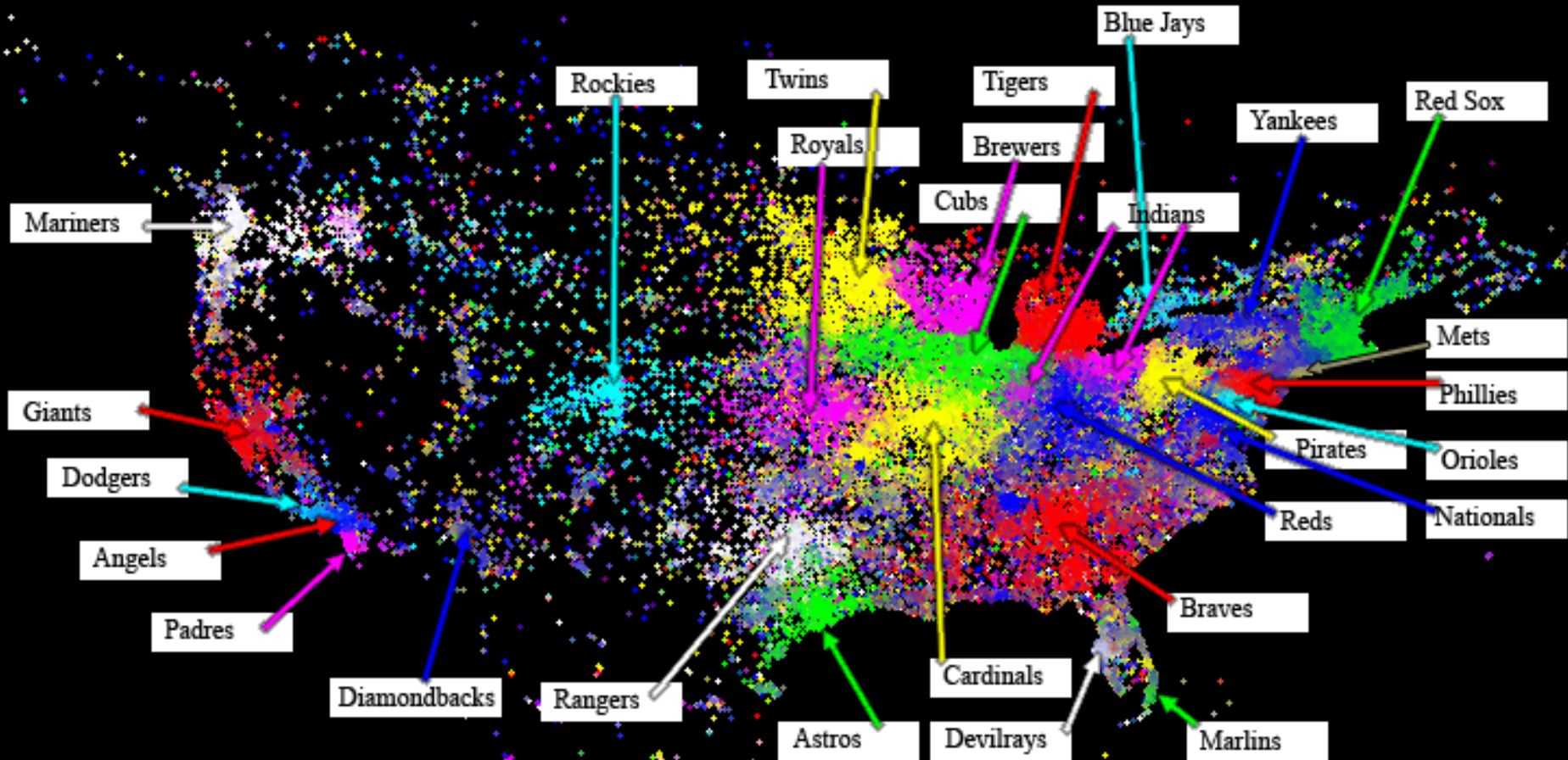


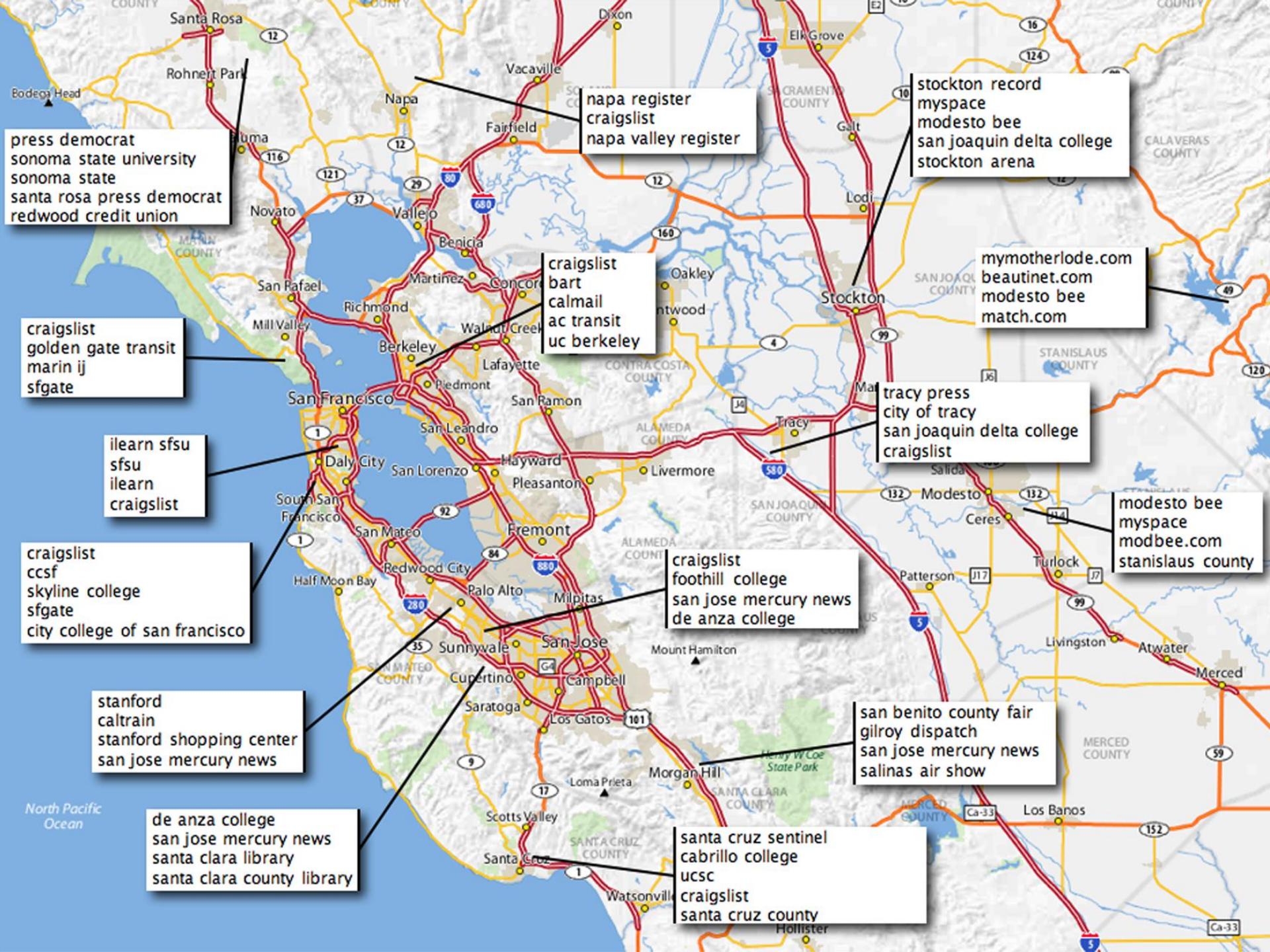
Algorithm

- Employing maximum likelihood estimation to estimate center, C and α
- Simple algorithm finds parameters which maximize likelihood
 - For given center, likelihood is unimodal and simple search algorithms find optimal C and α
 - Consider all centers on course mesh, optimize C and α for each center
 - Find best center, consider finer mesh



Baseball Team Queries





press democrat
sonoma state university
sonoma state
santa rosa press democrat
redwood credit union

napa register
craigslist
napa valley register

stockton record
myspace
modesto bee
san joaquin delta college
stockton arena

craigslist
golden gate transit
marin ij
sfgate

craigslist
bart
calmail
ac transit
uc berkeley

mymotherlode.com
beautinet.com
modesto bee
match.com

ilearn sfsu
sfsu
ilearn
craigslist

tracy press
city of tracy
san joaquin delta college
craigslist

craigslist
ccsf
skyline college
sfgate
city college of san francisco

craigslist
foothill college
san jose mercury news
de anza college

modesto bee
myspace
modbee.com
stanislaus county

stanford
caltrain
stanford shopping center
san jose mercury news

san benito county fair
gilroy dispatch
san jose mercury news
salinas air show

de anza college
san jose mercury news
santa clara library
santa clara county library

santa cruz sentinel
cabrillo college
ucsc
craigslist
santa cruz county

Identifying Localizable Queries

- Traditional work [Gravano03]
 - Using search results
 - If multiple locations distribute evenly in search result, the query is likely to be localizable query
- Recent work [Welch08][Yi09]
 - Using query log data
 - A localizable query is likely to appear as a sub query in other queries, associating with different locations
 - For example, some users may issue “car rental”, while others may issue “car rental california”, “car rental new york”, etc

Identifying Localizable Queries Using Query Log [Welch et al 08]

- An empirical study
 - Significant fraction of queries are localizable
 - Roughly 30%, but users only explicitly localize them about half of the time
 - Users exhibit consensus on which queries are localizable
- Approach
 - Identify candidate localizable queries
 - Select relevant features
 - Train and evaluate classifier

Identify Candidate Localizable Queries

- Use U.S. Census Bureau data as an address book
- For each query Q , look up the address book
 - If a match is found, the matched part is Q_l , the remaining part is Q_b
 - Q is a localized query of Q_b
- Aggregate all Q_l for each Q_b
 - The set of Q_l for each Q_b is denoted by $L(Q_b)$
 - Q_b is candidate localizable query if it often localized

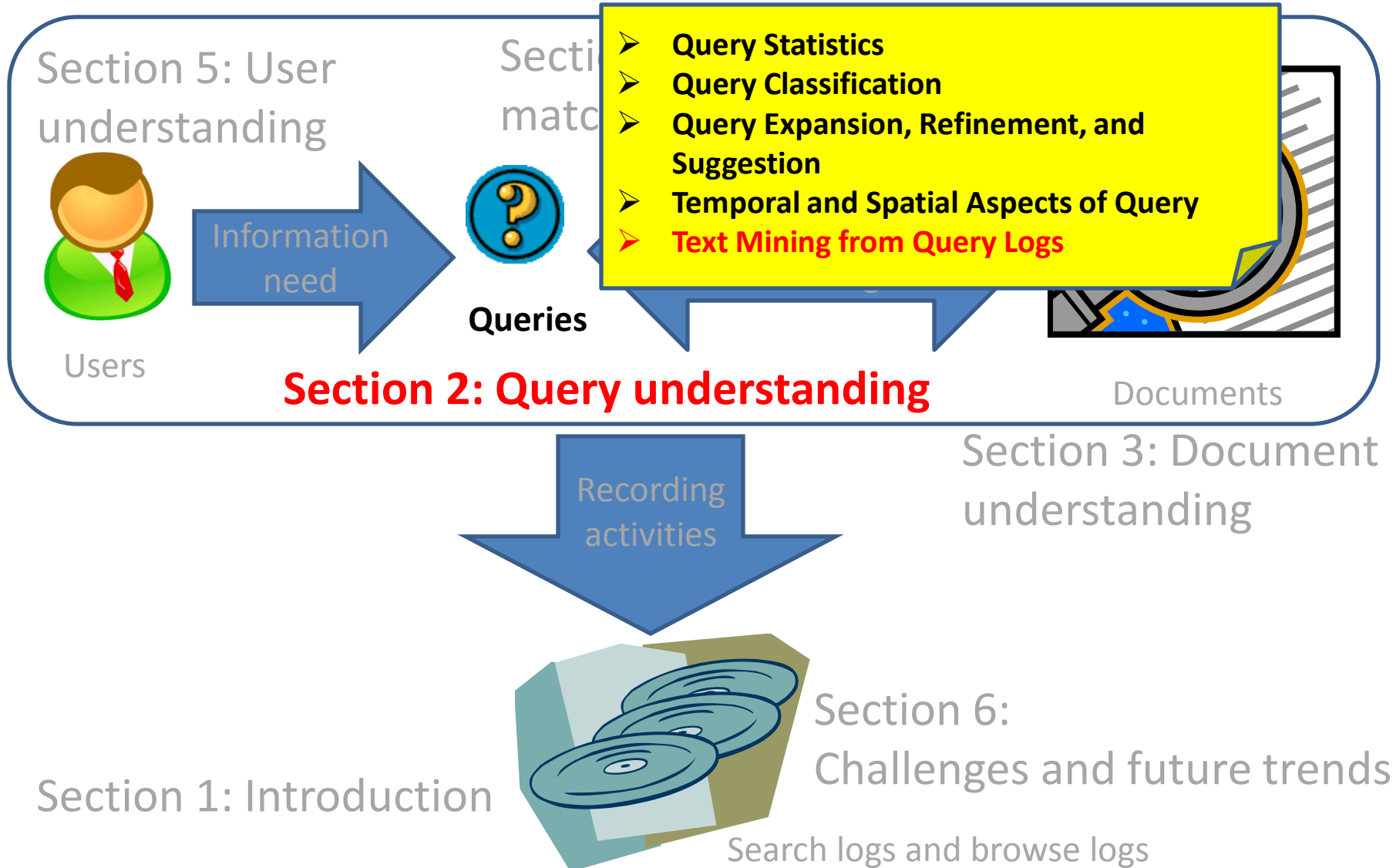
Major Features of Classifier

- Localization ratio
 - How often query is localized
- Location distribution
 - Query should be localized with evenly distributed locations
- Click-through rates
 - Click-through rate should be higher in localized query

Summary of Spatial Aspect of Queries

- Two directions to analyze queries from location perspective
- Local interest queries: identifying center and dispersion of local interest queries
- Localizable queries: identifying localizable queries using term co-occurrences

A Road Map



Outline of Text Mining from Query Logs

- Overview of Text Mining from Query Logs
- Named Entity Mining from Query Logs

Overview of Text Mining from Query Logs

- View search log data as `texts`
- Conduct text mining on log data
- Named entity mining
 - Entity extraction
 - Attribute extraction

Named Entity Mining from Query Logs

- To mine information about named entities in a class
- Examples
 - Cities: new york, los angeles, london, san francisco, dallas, boston, phoenix
 - Universities: harvard, stanford, michigan state university, oxford, mit, columbia, cambridge
- Over 70% queries contain named entities.

Methods for Named Entity Mining from Query Logs

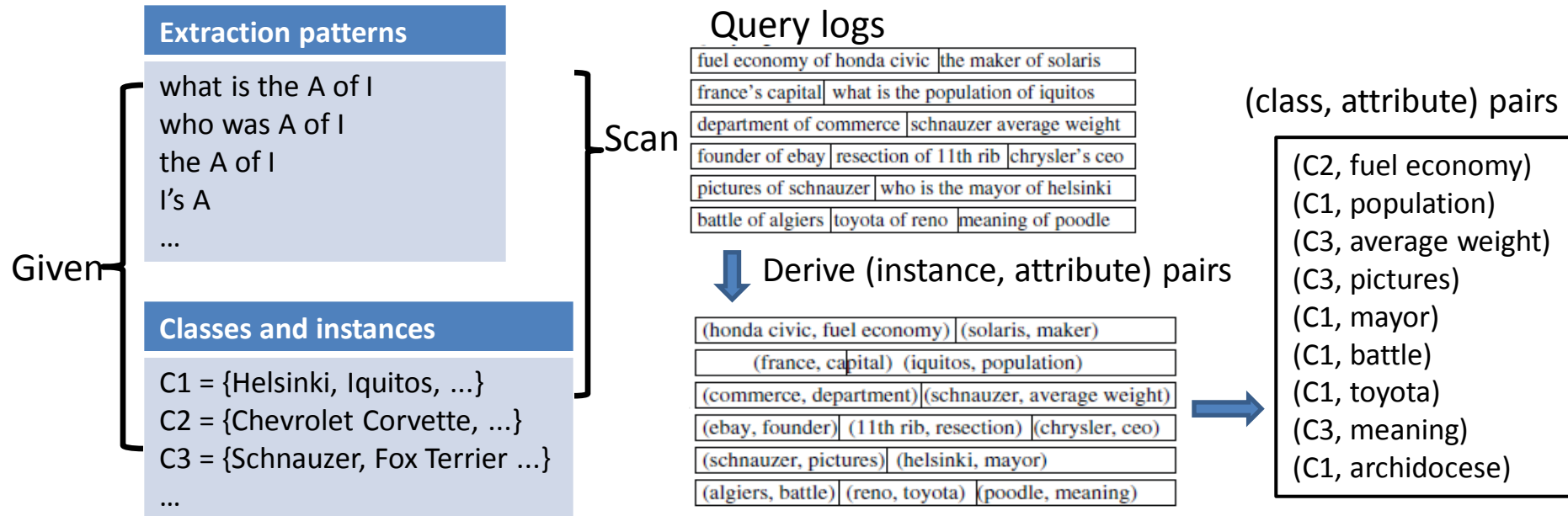
- Named entity mining using query log and weak supervision [Pasca07]
- Attribute mining using session data [Wang09]
- Named entity mining using topic model [Guo08, Xu08]

Named Entity Mining from Query Logs and Weak Supervision [Pasca07a]

- Assumption
 - Let C be a class, and $I \in C$ be instances of the class.
 - If A is a prominent attribute of C , then a fraction of queries about I are likely to ask both A and I
- Framework
 - Step 1: attribute generation
 - Step 2: attribute filtering
 - Step 3: attribute ranking

Attribute Generation

- Match queries with a set of seed attributes (patterns)
- Derive a list of (instance, attribute) pairs
- Replace (instance, attribute) pairs with (class, attribute) pairs
- Count frequency for each (class, attribute) pair



Attribute Filtering and Ranking

- Discard attributes that are proper nouns or part of proper nouns
 - With a large Web corpus
- Remove too generic attributes
 - e.g., meaning, story, summary, pictures
- Remove near-duplicate attributes
 - Two attributes are considered redundant if they have small edit distance
- Ranking attributes based on their frequencies

Named Entity Mining Using Query Log Data and Topic Model [Guo09]

- Using Query Log Data (or Click-through Data)
- Using Topic Model
- Weakly Supervised Latent Dirichlet Allocation
- vs Pasca's work (named entity mining from log data, deterministic approach)

Offline Step 1: Seed and Query Log

final fantasy

Movie Game

gone with the wind

Movie Book

harry potter

Movie Book Game

Named entity can belong to several classes

→ probabilistic approach

```
final fantasy 300
final fantasy movie 120
final fantasy wallpaper 50
gone with the wind movie 120
gone with the wind review 10
gone with the wind photos 10
harry potter 1000
harry potter book 650
gone with the wind book 80
gone with the wind summary 20
harry potter cheats 300
harry potter pics 200
harry potter summary 100
final fantasy xbox 10
final fantasy soundtrack 10
gone with the wind 250
harry potter movie 800
.....
```

Named Entity, Context, Class, and Frequency

final fantasy	<code>\#</code>	<i>Movie, Game</i>	300
final fantasy	<code>\# movie</code>	<i>Movie, Game</i>	120
final fantasy	<code>\# wallpaper</code>	<i>Movie, Game</i>	50
final fantasy	<code>\# xbox</code>	<i>Movie, Game</i>	10
final fantasy	<code>\# soundtrack</code>	<i>Movie, Game</i>	10
gone with the wind	<code>\#</code>	<i>Movie, Book</i>	250
gone with the wind	<code>\# movie</code>	<i>Movie , Book</i>	120
gone with the wind	<code>\# book</code>	<i>Movie , Book</i>	80
gone with the wind	<code>\# summary</code>	<i>Movie ,Book</i>	20
gone with the wind	<code>\# review</code>	<i>Movie , Book</i>	10
gone with the wind	<code>\# photos</code>	<i>Movie , Book</i>	10
harry potter	<code>\#</code>	<i>Movie, Book, Game</i>	1000
harry potter	<code>\# movie</code>	<i>Movie, Book, Game</i>	800
harry potter	<code>\# book</code>	<i>Movie, Book, Game</i>	650
harry potter	<code>\# cheats</code>	<i>Movie, Book, Game</i>	300
harry potter	<code>\# pics</code>	<i>Movie, Book, Game</i>	200
harry potter	<code>\# summary</code>	<i>Movie, Book, Game</i>	100

Pseudo Documents of Named Entities

final fantasy

\#	300
\# movie	120
\# wallpaper	50
\# xbox	10
\# soundtrack	10

Movie, Game



Labels of document:
Topics

gone with the wind

\#	250
\# movie	120
\# book	80
\# summary	20
\# review	10
\# photos	10

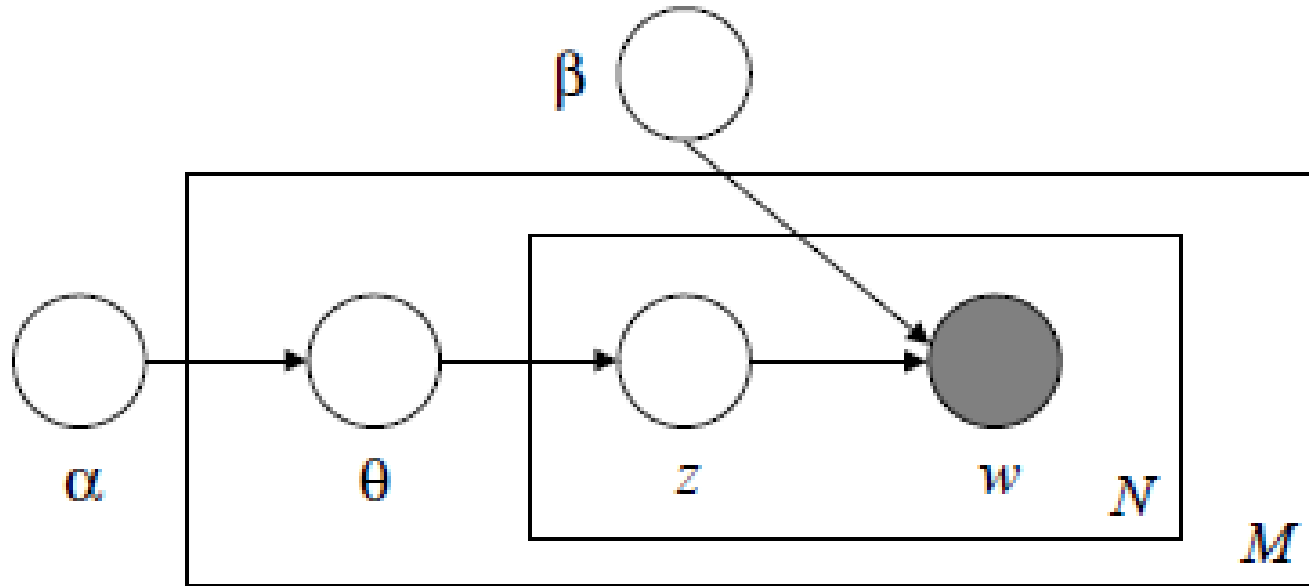
Movie, Book

harry potter

\#	1000
\# movie	800
\# book	650
\# cheats	300
\# pics	200
\# summary	100

Movie, Book, Game

Offline Step 2: Building Latent Dirichlet Allocation Model



z : *Movie, Book, Game*

w : \#, \# movie, \# book,

θ : distribution of classes for named entity

β : distribution of contexts for class

Offline: Weakly Supervised Latent Dirichlet Allocation

$$p(\mathcal{D}|\Theta) = \prod_{d=1}^M \int p(\theta_d|\alpha) \left(\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn}|\theta_d) p(w_{dn}|z_{dn}, \beta) \right) d\theta_d$$

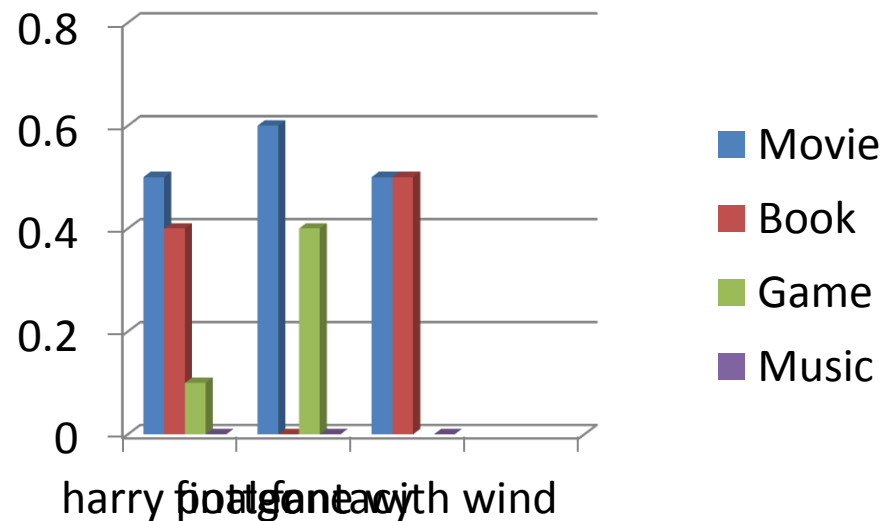
$$\log p(\mathcal{D}|\Theta) + \lambda C(\Theta, y)$$

$$= \sum_{d=1}^M \log \int p(\theta_d|\alpha) \left(\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn}|\theta_d) p(w_{dn}|z_{dn}, \beta) \right) d\theta_d$$

$$+ \sum_{d=1}^M \lambda \sum_{i=1}^K y_{di} \bar{z}_{di}$$

$$\bar{z}_{di} = \frac{1}{N} \sum_{n=1}^N z_{ni}^d$$

constraints



Learned Probabilities

$$P(w | z, \beta)$$

$$P(z | \theta)$$

\#	0.5
\# movie	0.2
\# review	0.1
\# wallpaper	0.1
\# photos	0.1

Movie

final fantasy

Movie 0.5

Game 0.5

gone with the wind

Movie 0.6

Book 0.4

harry potter

Movie 0.6

Book 0.3

Game 0.1

\#	0.8
\# book	0.1
\# summary	0.05
\# review	0.05

Book

\#	0.6
\# pics	0.2
\# cheats	0.1
\# xbox	0.05
\# soundtrack	0.05

Game

Online: Inference

kung fu panda

\#	250
\# movie	100
\# wallpaper	20
\# walkthrough	10
\# review	10

Movie, Game?

beautiful mind

\#	200
\# movie	150
\# summary	60
\# review	40
\# book	80

Movie, Book?

kung fu panda

Movie 0.9

Game 0.1

beautiful mind

Movie 0.7

Book 0.3

Summary of Text Mining from Query Logs

- Named entity mining using query log and patterns
- Attribute mining using session data
- Named entity mining using topic model

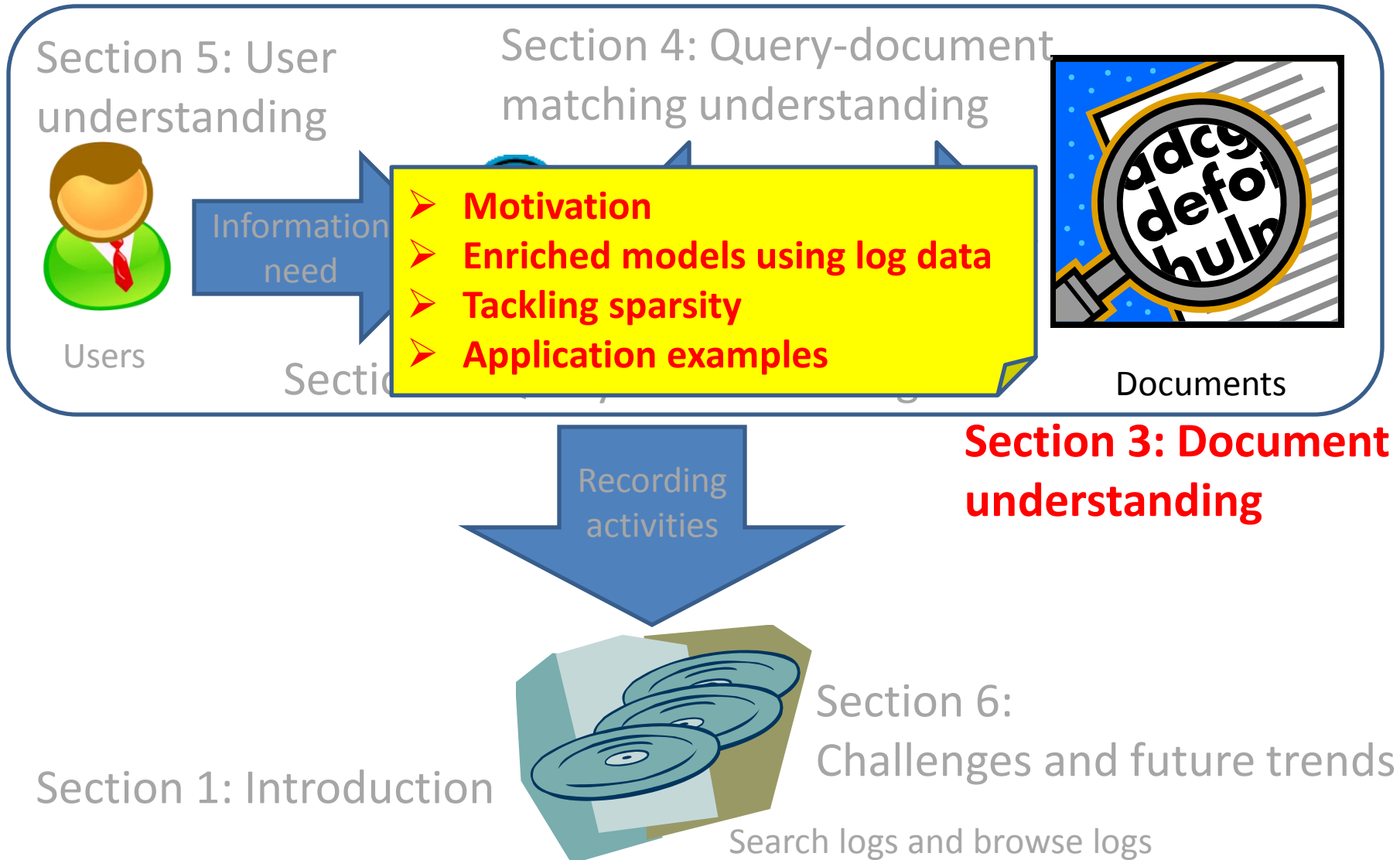
Summary of Log Mining for Query Understanding

- Query Statistics
- Query Classification
- Query Expansion, Refinement, and Suggestion
- Temporal and Spatial Aspects of Queries
- Text Mining from Query Logs

Summary of Query Log Mining for Query Understanding

- At macro level
 - Query Statistics
 - Query Temporal Trend Analysis
- At micro level
 - Query Classification (Intent Understanding)
 - Task, Topic, Entity & Attribute, Time, Location
 - Offline: Large Scale Log Mining
 - Online: Query Classification, Query Expansion, Refinement, Suggestion
- Click-through data and session data are very useful

A Road Map



Modeling Documents

- Traditionally, a document is modeled as a bag of words
- Vector model
 - $V = \{v_1, \dots, v_n\}$, the set of terms
 - A document $d = (w_1, \dots, w_n)$, where w_i is the importance of term v_i in d
 - Importance can be measured by, for example, TFIDF
 - $TF(v, d) = \#$ of times term v appears in d
 - $IDF(v) = \log(N / \#$ of documents in the corpus containing v)
 - $TFIDF(v, d) = TF(v, d) * IDF(v)$
- A vector model tries to capture what the author of a document wants to express using the terms in the document

Web Documents and Links

- A Web page may be referred (pointed to) by other Web pages
 - A link to the target page
 - Anchor text: a short annotation on the intension of reference
- A page having many incoming links tends to be important (well explored by link-based ranking methods, e.g., PageRank)
- What does anchor text tell us? – what others on the Web think about the target page

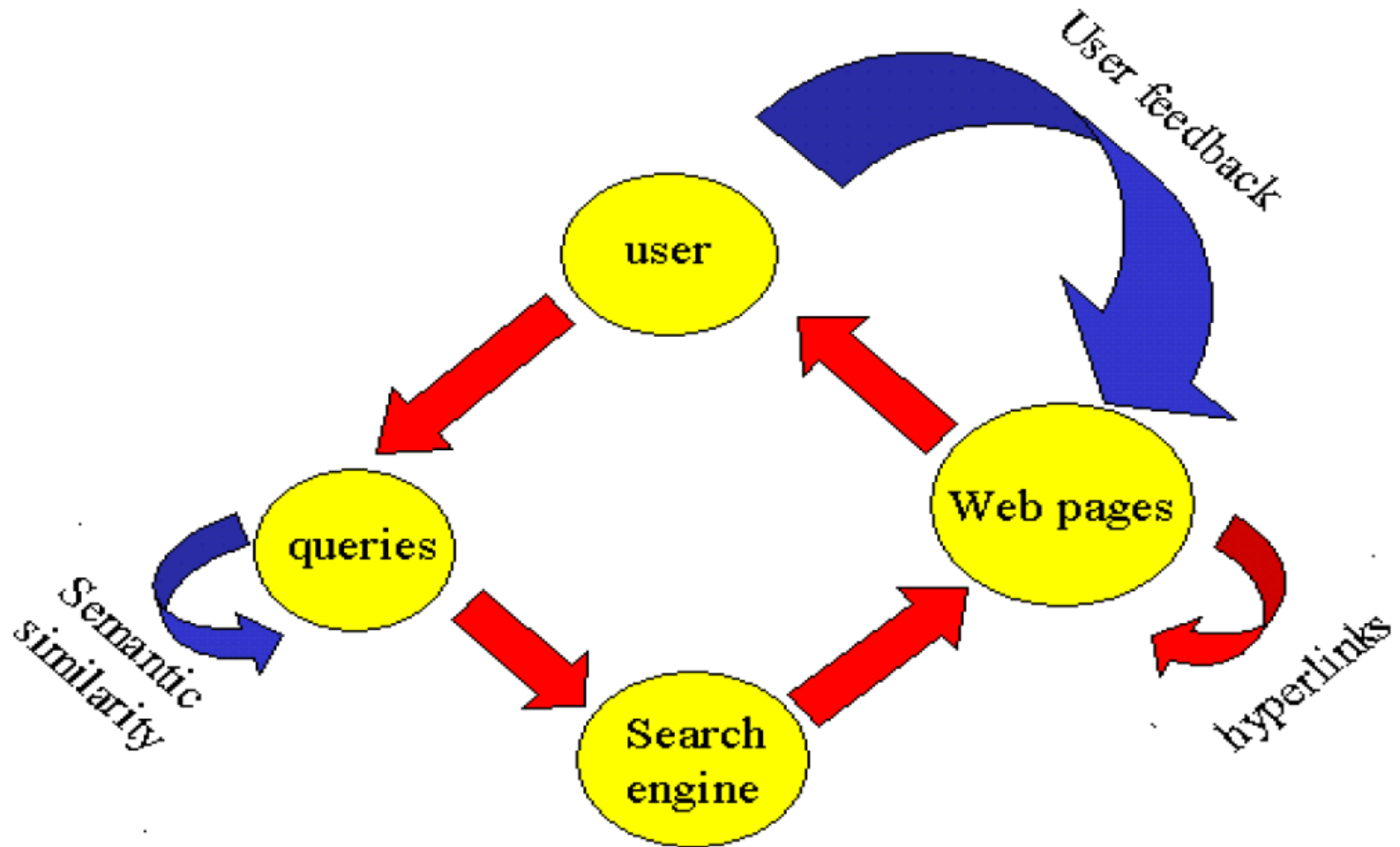
Anchor Text

- Anchor text may not be consistent with the vector model of the target page
 - Example: Anchor text “my publications” → DBLP bibliography server
- Anchor text can be used to complement the vector model of a target Web page
 - What the author writes + what some others read
- Anchor text tends to be bias and static
 - Old pages may receive more references, and thus more anchor text annotations
 - Once a link is made, more often than not, it will not be updated (at least in a long time)

Search Logs as User Annotations

- User queries → search results → user clickthroughs
 - If a user asks a query Q and clicks a page P, likely P is related to Q – Q can be used as an annotation of why a user wants to read P
- User clickthroughs can be used as dynamic, continuously updated, more accurate (after aggregation) annotation of Web pages

A Bigger Picture



Ricardo Baeza-Yates, Carlos Hurtado, and Marcelo Mendoza. Query Clustering for Boosting Web Page Ranking AWIC, 2004.

A User Study

- For a query Q
 - Query suggestion generated by frequently asked queries containing Q as a substring or following Q in sessions
 - Query destination – also show Web pages that most clicked by the users asking similar queries
- Two types of tasks
 - Known-item task: “find three tropical storms that have caused property damage and/of loss of life”
 - Exploratory task: “learn about VoIP technology and service providers, select the provider and telephone that best suits you”
- Query suggestion and destination improve user search experience substantially
 - Query destination works particularly well for exploratory tasks
- [R. W. White, M. Bilenko, and S. Cucerzan. Studying the use of popular destinations to enhance web search interaction. SIGIR'07.]

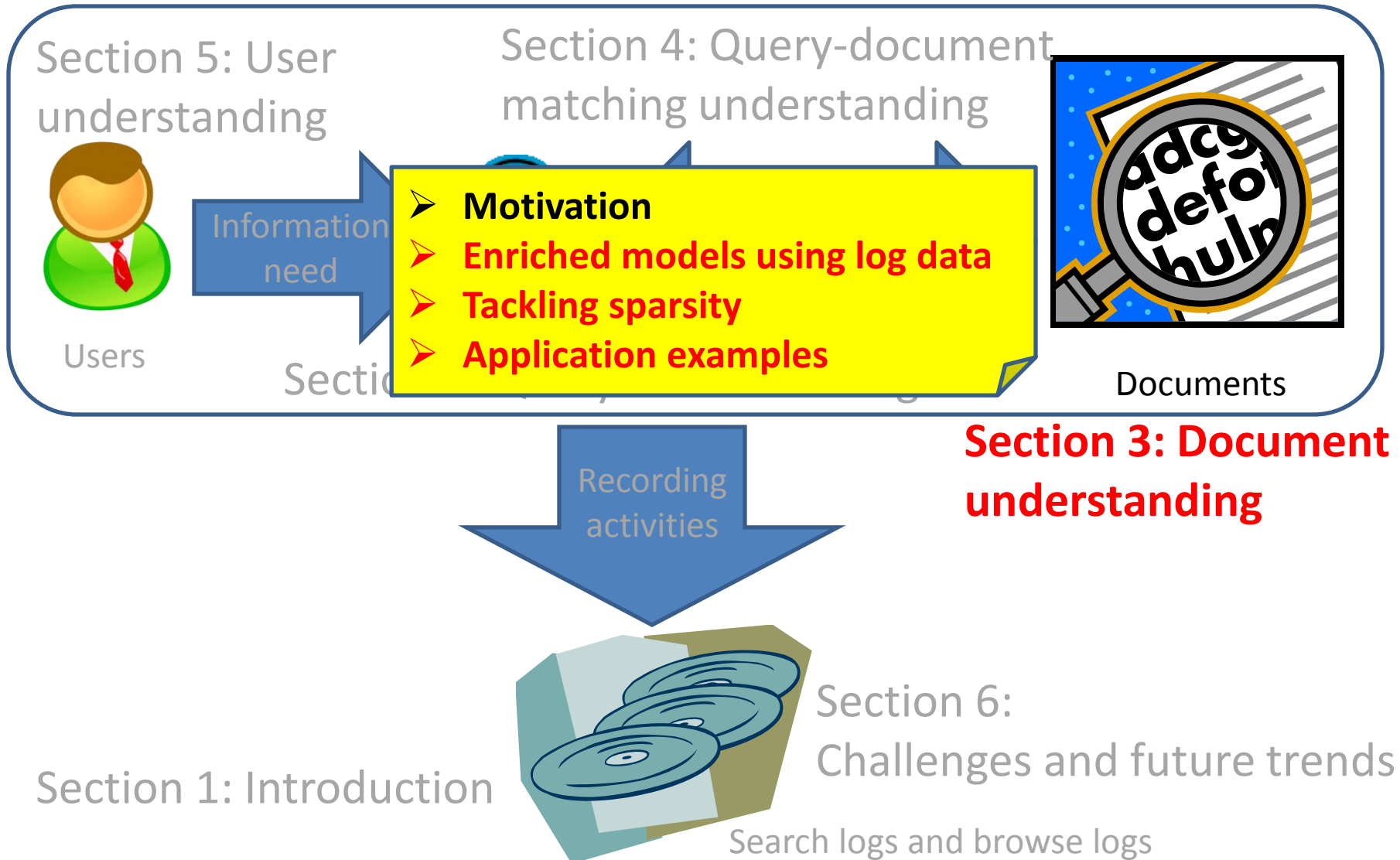
Why Can We Learn?

- In query destination, Web pages in the destination recommendation part are selected based on their “query annotation” instead of their content vector model
- Queries as annotation can improve the accuracy of matching user information needs and documents

Challenges

- How to model “query annotations”?
- Search log data is sparse, how to handle documents that have very few or even no clicks?
 - A small number of queries are frequently asked, many queries are rarely asked
 - A small number of Web pages are heavily clicked, many Web pages have very few or even no clicks
- How to use “query annotations”?

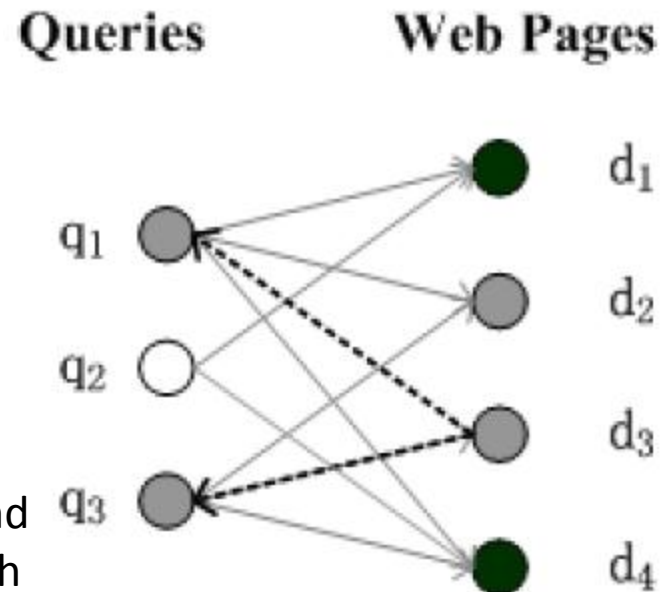
A Road Map



Using Queries as Features

- Queries can be used as features to model documents
- Two documents are similar if they are clicked in the same set of queries

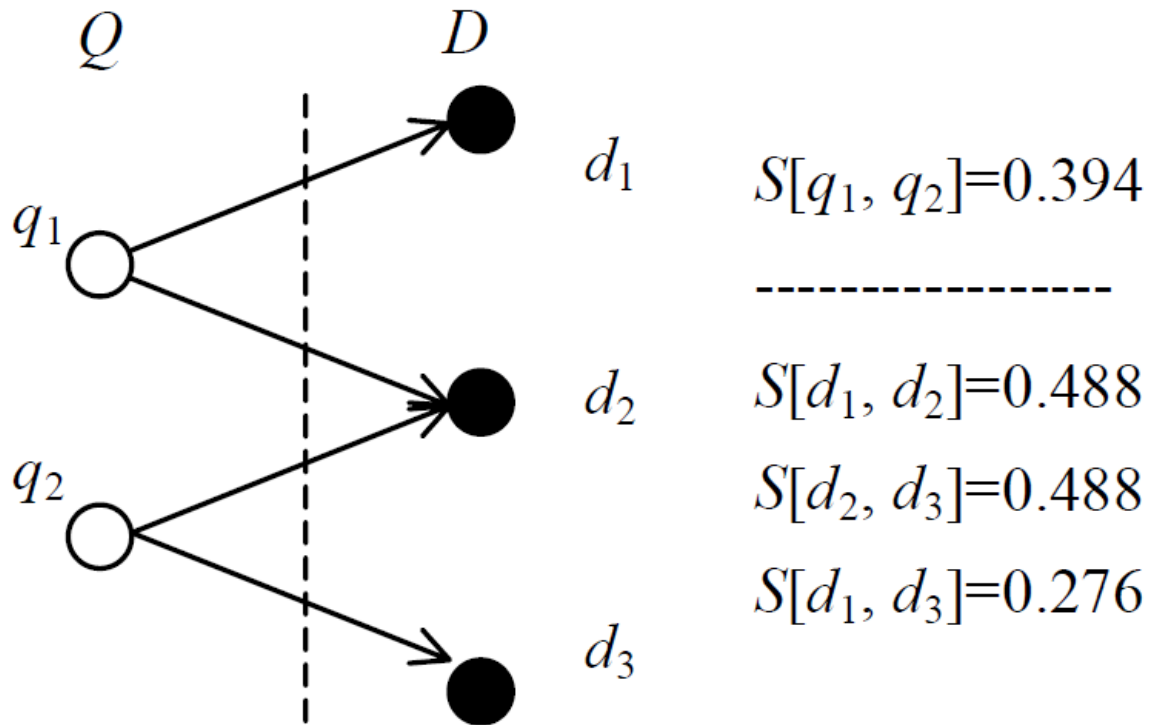
Using queries as “bridges”, similar documents d_2 and d_3 can be captured



G.-R. Xue, H.-J. Zeng, Z. Chen, Y. Yu, W.-Y. Ma, W. Xi, and W. Fan. Optimizing web search using web click-through data. CIKM '04.

Two-Way Annotation

- Can we use documents as features of queries?



G.-R. Xue, H.-J. Zeng, Z. Chen, Y. Yu, W.-Y. Ma, W. Xi, and W. Fan. Optimizing web search using web click-through data. CIKM '04.

Query-Document Model

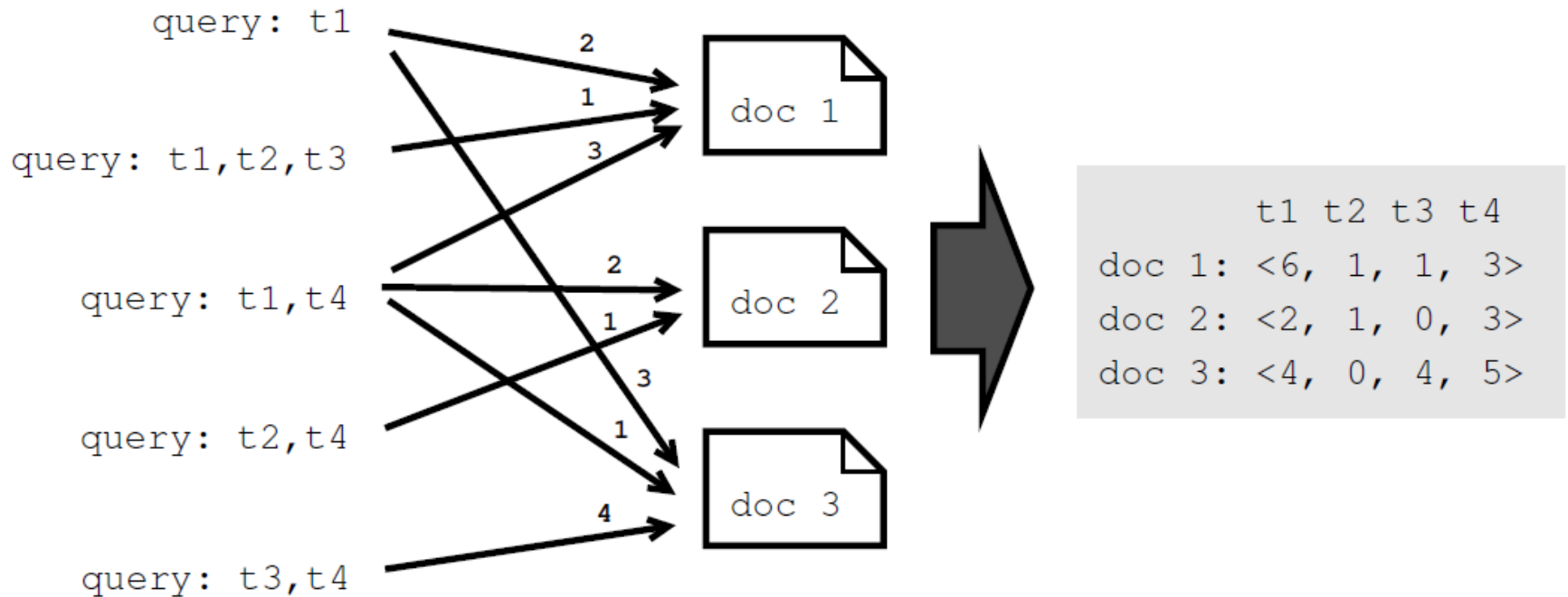
- Let $V = \{t_1, \dots, t_m\}$ be the vocabulary of all queries in the access log L , where t_1, \dots, t_m are the terms in V
- Let $Q(d)$ be the set of all queries in L from which users clicked at least one time on d
- Let the frequency of t in $Q(d)$ be the total number of times that queries that contained t were used to visit d

$$\vec{d} = \langle C_1, \dots, C_m \rangle$$

– Where $C_i = \text{TFIDF}(t_i, Q(d))$

- [Barbara Poblete, Ricardo Baeza-Yates, Query-sets: using implicit feedback and query patterns to organize web documents. WWW'08.]

Example



Barbara Poblete, Ricardo Baeza-Yates, Query-sets: using implicit feedback and query patterns to organize web documents. WWW'08.

Query-Set Document Model

- Query-document model considers terms in queries independently even if some of them co-occur frequently
 - “Apple” and “Apple phone” carry very different meanings
- Query-set document model includes frequent term combinations as features for documents
- [Barbara Poblete, Ricardo Baeza-Yates, Query-sets: using implicit feedback and query patterns to organize web documents. WWW'08.]

Example

t1
t1,t2,t3
t1,t4
t2,t4
t3,t4
t1,t2,t4
t1,t3,t4
t3
t2,t3
t1,t3

queries



freq.	support	set
6	60%	t3
6	60%	t1
5	50%	t4
4	40%	t2
3	30%	t1 t4
3	30%	t1 t3
2	20%	t2 t4
2	20%	t2 t3
2	20%	t1 t2
2	20%	t3 t4
1	10%	t1 t2 t4
1	10%	t1 t2 t3
1	10%	t1 t3 t4

term sets

Barbara Poblete, Ricardo Baeza-Yates, Query-sets: using implicit feedback and query patterns to organize web documents. WWW'08.

Case Study

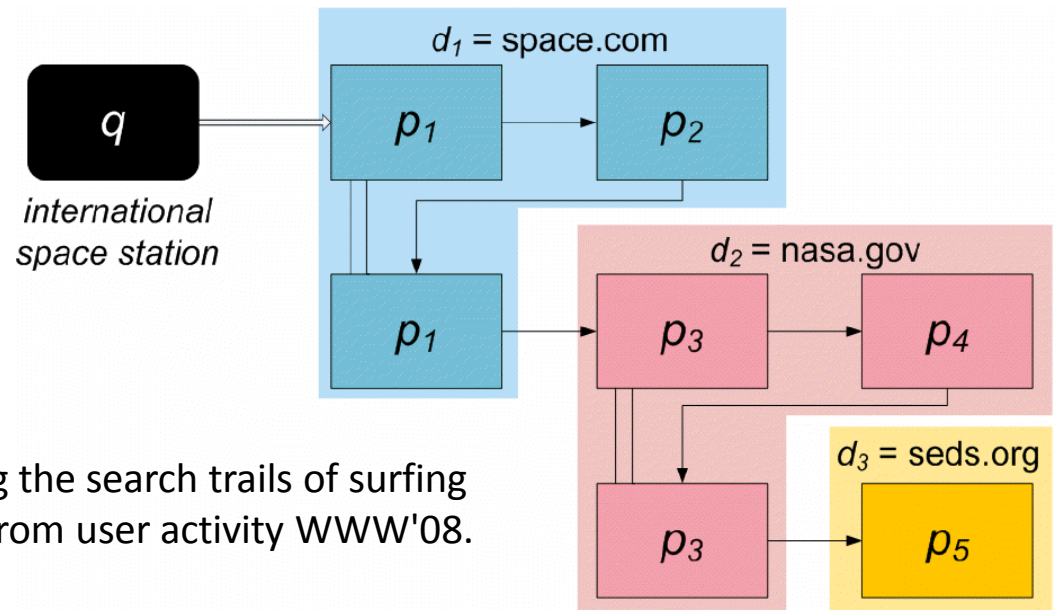
DocId	Vector Space	Query	Query-Set
58	download, test, file, 2007, guide, publication	official, test, social, publication, module, science, guides	physics, geometry, physics topics, topics, admission topics
74	able, Europe, world, kingdom, MBA, Asia, library	degree, search, graduate, certificate, advanced, diploma, simulation	university scholarship, universities, university ranking, best universities
47	scholarship, application, loan, benefit, fill, form	dates, free, vocational, on-line, scholarship, loan	loan scholarship loan cosigner loan application
80	vitae, curriculum, presentation, job, letter, interview, experience, highlight	CV, letter, resume, recommendation, presentation, example	CV, write CV, curriculum vitae, CV example, write curriculum vitae

<i>Model</i>	<i>Quality</i>	<i>Dimensions</i>	<i>Agreement</i>
Vector-Space	40%	8,910	69%
Query	57%	7,718	67%
Query-Set	77%	564	81%

Barbara Poblete, Ricardo Baeza-Yates, Query-sets: using implicit feedback and query patterns to organize web documents. WWW'08.

Browse Logs and Search Trails

- Browse logs may contain more information than search log
 - Search trails record other browsing activities in addition to queries



Mikhail Bilenko, Ryen W. White. Mining the search trails of surfing crowds: identifying relevant websites from user activity WWW'08.

A Generative Model

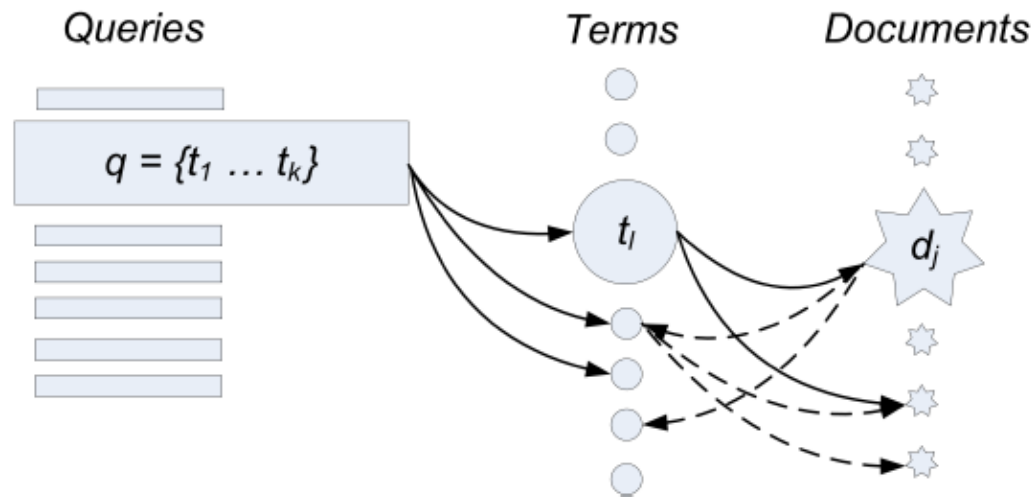
- A set of search trails $D = \{q \rightarrow (d_1, \dots, d_m)\}$, where d_1, \dots, d_m are documents
- Assuming every query q instantiates a multinomial distribution over its terms

$$\text{Rel}_p(d, \hat{q}) = p(d | \hat{q}) = \sum_{\hat{t} \in q} p(\hat{t} | \hat{q}) p(d | \hat{t})$$

- [Mikhail Bilenko, Ryan W. White. Mining the search trails of surfing crowds: identifying relevant websites from user activity WWW'08.]

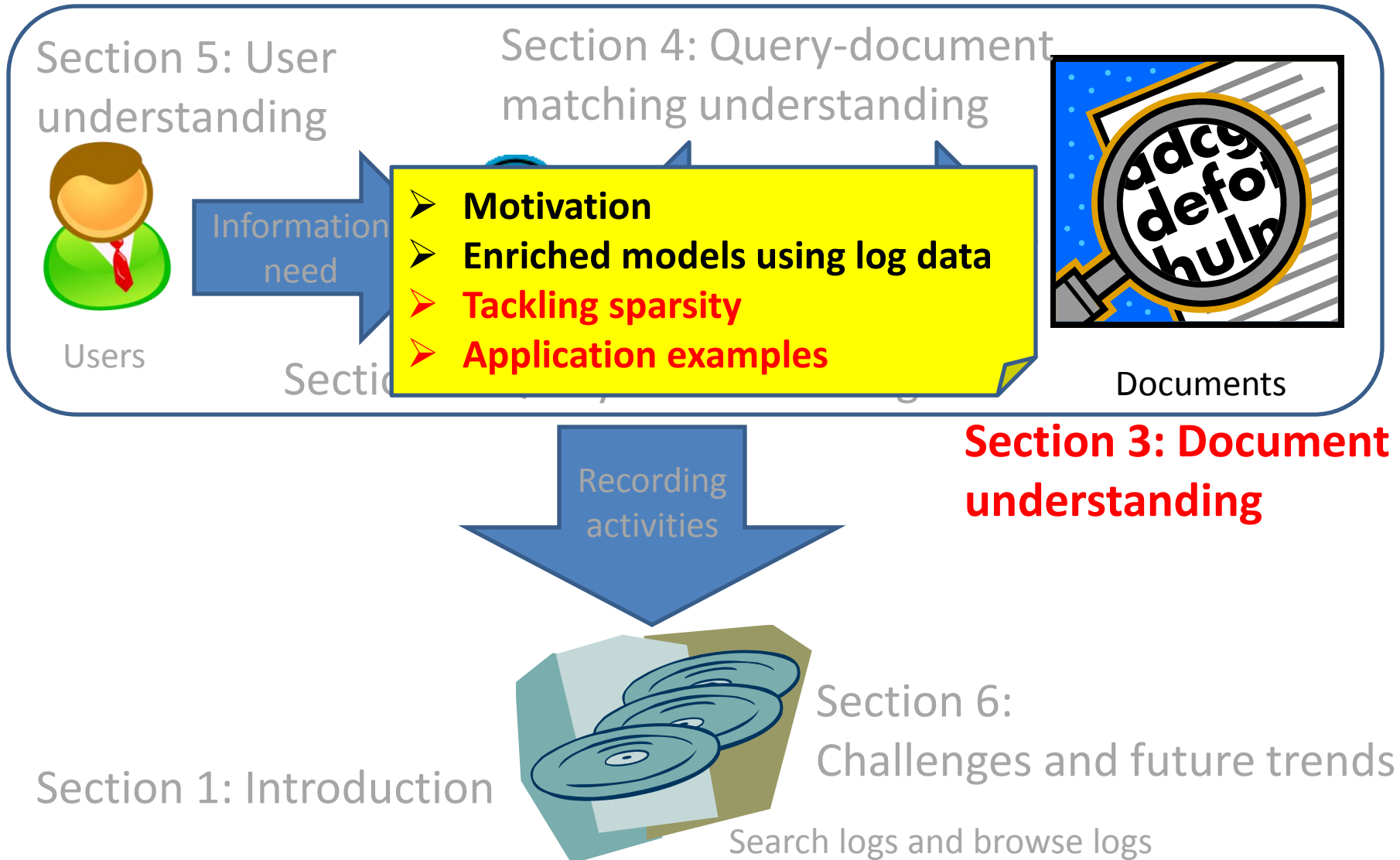
A Random Walk Extension

- The probability of reaching a document starting from a given query is the likelihood of hitting the document node via the two-step random walk that originates at the query node and proceeds via the term nodes



- [Mikhail Bilenko, Ryen W. White. Mining the search trails of surfing crowds: identifying relevant websites from user activity WWW'08.]

A Road Map



Tackling Query Sparsity

- Many queries are rarely asked
- Idea: clustering similar queries to identify groups of user information needs of significant sizes → reliable annotations on Web pages clicked
- A two phase algorithm
 - Preprocessing phase
 - Online searching phase
- [Ricardo Baeza-Yates, Carlos Hurtado, and Marcelo Mendoza. Query Clustering for Boosting Web Page Ranking AWIC, 2004.]

Preprocessing Phase

- At periodical and regular intervals
- Extract queries and clicked URLs from the Web log, and cluster them using the text of all the clicked URLs (by k-means)
- For each cluster C_i , compute and store
 - A list Q_i containing queries in the cluster
 - A list U_i containing the k-most popular URLs along with their popularity
- [Ricardo Baeza-Yates, Carlos Hurtado, and Marcelo Mendoza. Query Clustering for Boosting Web Page Ranking AWIC, 2004.]

Online Searching Phase

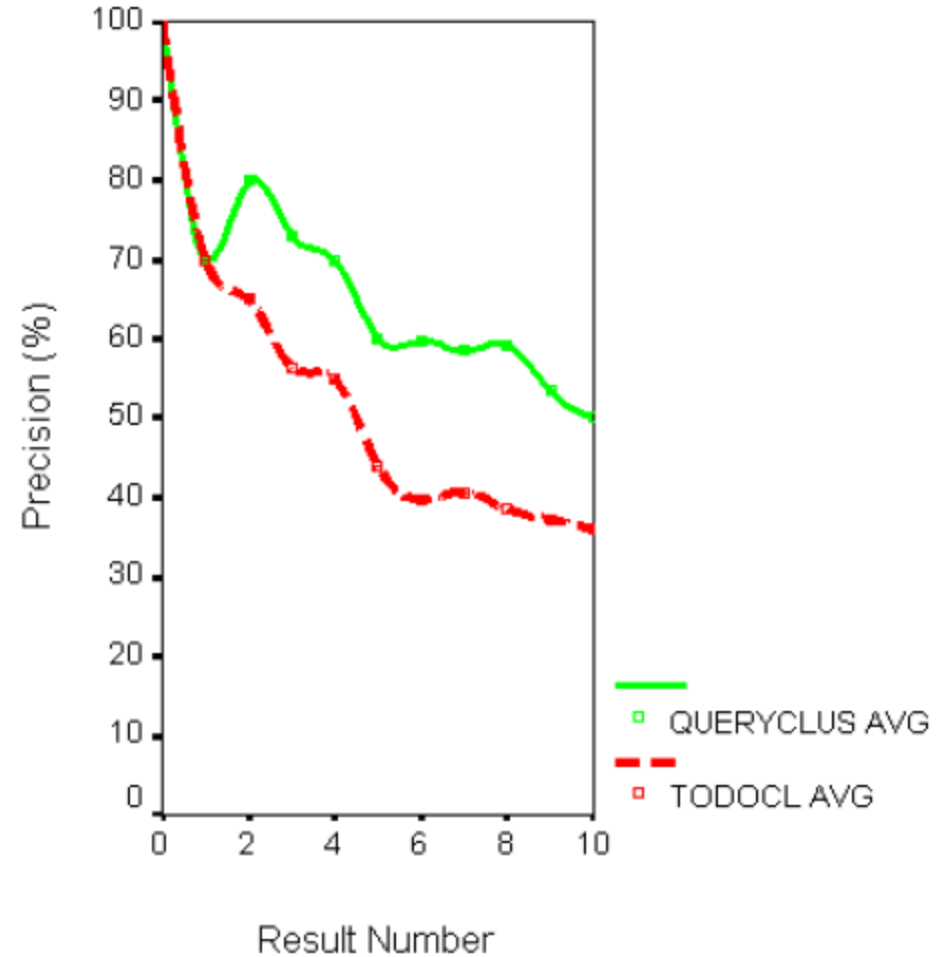
- Input: a query q
- If q appears in the stored clusters, find the corresponding cluster C_i containing q , use U_i to boost the search engine ranking algorithm by

$$\mathit{NewRank}(u) = \beta \times \mathit{OrigRank}(u) + (1 - \beta) \times \mathit{Rank}(u)$$

- [Ricardo Baeza-Yates, Carlos Hurtado, and Marcelo Mendoza. Query Clustering for Boosting Web Page Ranking AWIC, 2004.]

Examples & Effectiveness

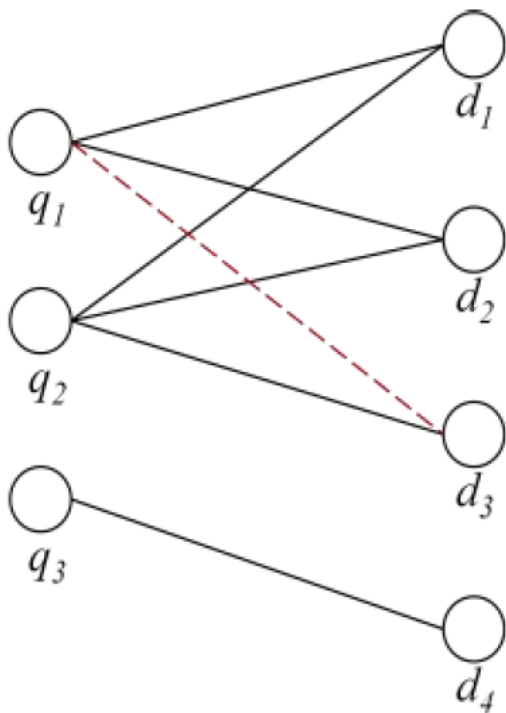
Query	Other Queries in Cluster.
dress bride	house of bride dress wedding dress bridegroom wedding cake wedding rings
free internet	phone company free internet connection free ads <i>cibercafe santiago</i> free text messages free email
yoga	tai chi exercises astral letter reiki birth register
soccer leagues	<i>ivan zamorano</i> soccer leagues chile soccer teams chile <i>marcelo salas</i>



Documents Not Clicked

- Many documents may have very few or even no clicks
 - 75% of a sample of 2.62 million Web pages do not have any click in a real case study
- Idea: use smoothing techniques
 - Random walk
 - Discounting
- [Gao, J., et al. Smoothing clickthrough data for web search ranking. SIGIR'09.]

Random Walk



Construct matrix $A_{ij} = P(d_i | q_j)$ and matrix $B_{ij} = P(q_i | d_j)$

Random walk using the probabilities

Before expansion, document d_3 has a clickthrough stream of q_2 only; after a random walk expansion, the click-through stream is augmented with query q_1 , which has a similar click pattern as q_2

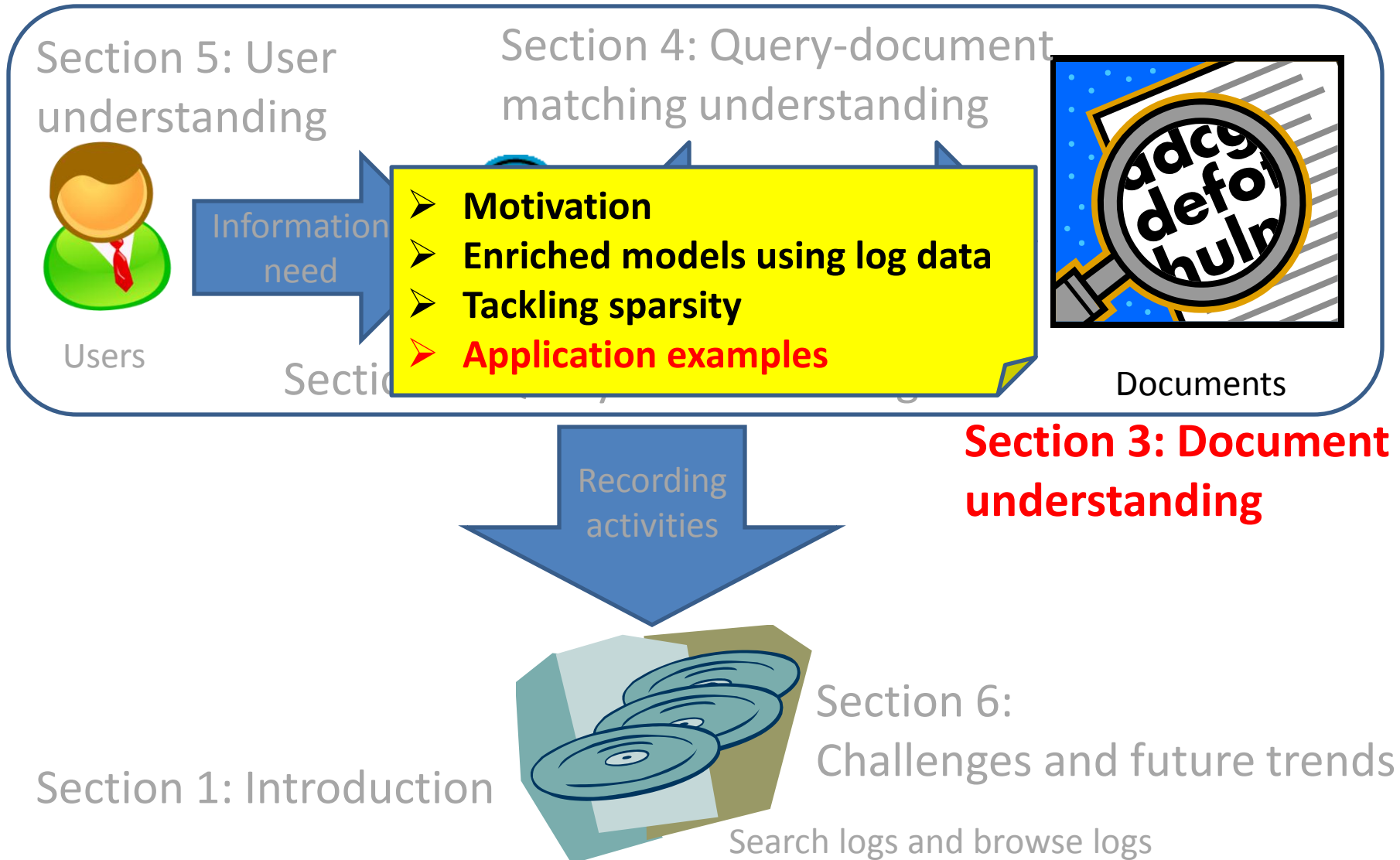
Good-Turing Estimator

- Let N be the size of a sample text, n_r be the number of words which occur in the text exactly r times $N = \sum_r r n_r$
- Estimate P_{GT} for a probability of a word that occurred in the sample r times as $P_{GT} = \frac{r^*}{N}$, where $r^* = (r+1) \frac{n_{r+1}}{n_r}$
- Heuristic: not discounting high values of counts, i.e., for $r > k$ (typically $k = 5$), $r^* = r$

Discounting

- Applying Good-Turing estimate on raw clickthrough data does not work – all not-clicked words take the same free ride
 - Those features are meaningless
- Idea: discounting the clickthrough feature values
 - Details in [Gao, J., et al. Smoothing clickthrough data for web search ranking. SIGIR'09.]
- The discounting method works very well in the empirical studies

A Road Map



Using Logs and Query Annotations

- Generating keyword
- Learning document importance
- Organizing search results
- ...

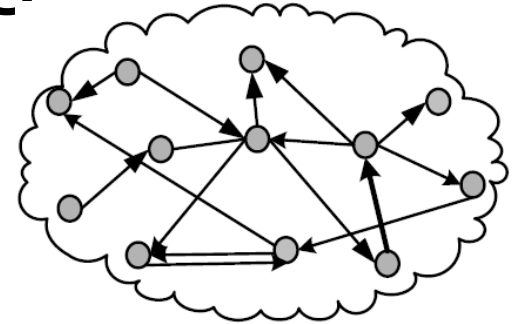
Keyword Generation

- What are the keywords that describe the concept “shoes”?
 - Starting point: shoes.com is about shoes
 - A user asked “running shoes” and clicked shoes.com → “running shoes” is about shoes
 - The user also clicked runningshoes.com → runningshoes.com is also about shoes
 - Queries “reebok shoes” and “rebok shoes” led to clicks on runningshoes.com → those keywords are also about shoes
- Given a concept (e.g., shoes), a set of elements representing the concept (e.g., a set of URLs), and the relationship between the documents and the queries, find a set of keywords capturing the concept best
- [Ariel Fuxman, Panayiotis Tsaparas, Kannan Achan, Rakesh Agrawal Using the wisdom of the crowds for keyword generation WWW'08.]

A Semi-Supervised Version

- Input
 - A set of labeled objects about a concept (e.g., URLs)
 - A set of unlabeled objects (the remaining URLs and the queries in the log)
 - A set of constraints between labeled and unlabeled objects (the click log)
- Task: label some of the unlabeled elements in a meaningful way
- Idea: use Markov random fields to model the query click graph
- [Ariel Fuxman, Panayiotis Tsaparas, Kannan Achan, Rakesh Agrawal Using the wisdom of the crowds for keyword generation WWW'08.]

Modeling User Browsing Behavior



- User browsing graph
 - Vertices representing pages
 - Directed edges representing transitions between pages in browsing history
 - Lengths of staying time are included
- Using the continuous-time Markov process
 - The stationary probability distribution of the process is the importance of a page
- [Yuting Liu, Bin Gao, Tie-Yan Liu, Ying Zhang, Zhiming Ma, Shuyuan He, Hang Li. BrowseRank: letting web users vote for page importance. SIGIR'08.]

ClickRank

- A session is modeled as a logical sequence of hops through the Web graph according to the user's retrieval intention
 - Temporal attributes (e.g., dwell time) reflects user's interest on a page
- For a session s , the local ClickRank defines a random variable associated with all pages on the Web graph reflecting how important a page is to the user's retrieval intention in this session
- [Zhu, G, Mishne, G. Mining rich session context to improve web search. KDD'09.]

Organizing Search Results

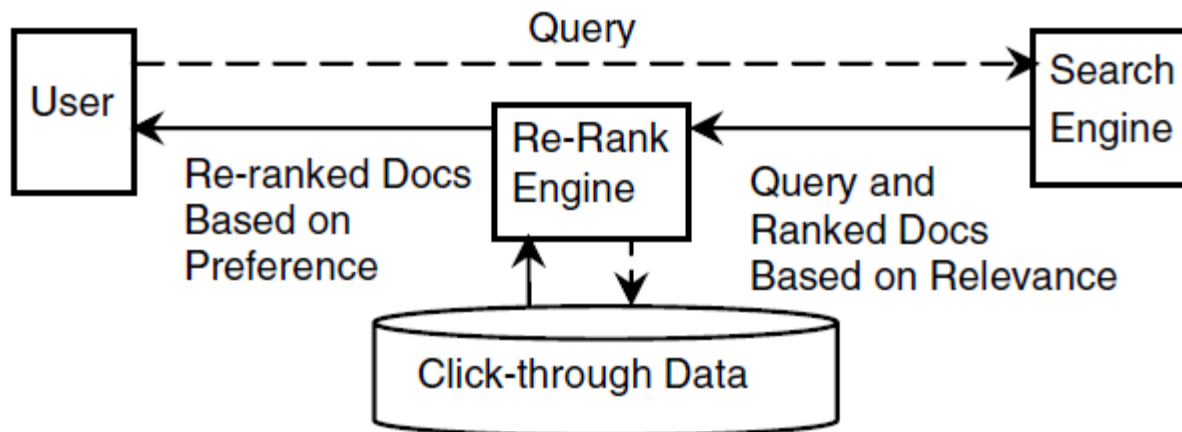
- Query “jaguar” is ambiguous: car, animal, software, or a sport team?
 - Instead of presenting a mixed list of results, a user may prefer clusters of results according to the senses
- Challenges in clustering results
 - Clusters may not necessarily correspond to the interesting aspects of a topic from the user’s perspective
 - Cluster labels may not be informative
- [X. Wang and C. Zhai. Learn from web search logs to organize search results. SIGIR'07.]

Clustering Using Search Logs

- What kinds of pages viewed by users in the results of a query?
 - Finding aspects interesting to users by mining user clickthrough data
- Generate meaningful cluster labels using query words entered by users
- [X. Wang and C. Zhai. Learn from web search logs to organize search results. SIGIR'07.]

Using Search Logs in Re-Ranking

- Search engine → candidate answers and baseline ranking
- Click-through data → learning user preference for re-ranking (more on next section)



Min Zhao, Hang Li, Adwait Ratnaparkhi, Hsiao-Wuen Hon, and Jue Wang. Adapting document ranking to users' preferences using click-through data, AIRS'06.

More Examples ...

- Considering clickthrough data in page summarization
 - Category maintenance
 - ...
-
- [J.-T. Sun, D. Shen, H.-J. Zeng, Q. Yang, Y. Lu, and Z. Chen, Web-page summarization using clickthrough data, SIGIR '05.]
 - [A. Cid, C. Hurtado, and M. Mendoza, Automatic maintenance of web directories using click-through data, in ICDEW '06.]

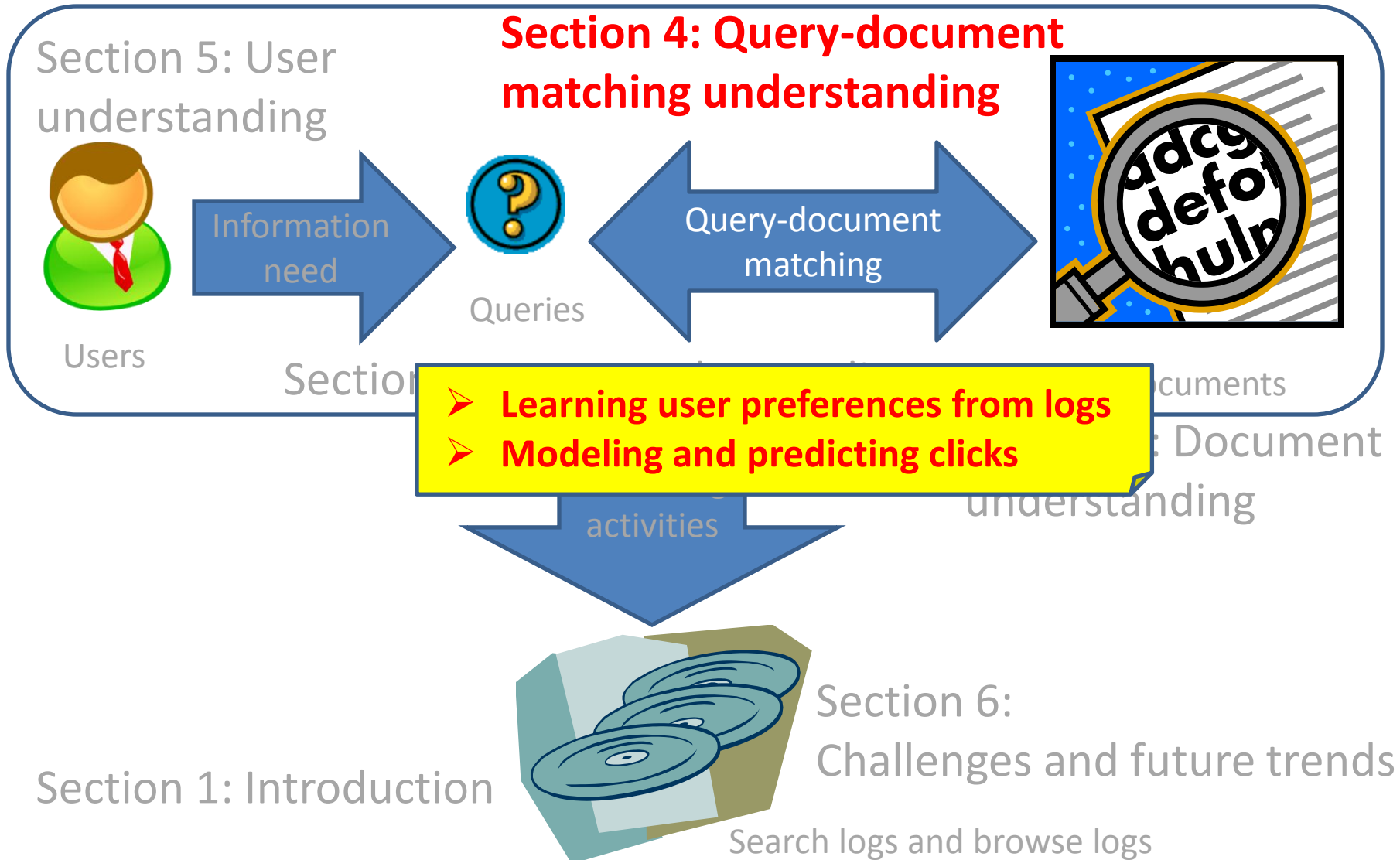
Summary

- Search logs and browse logs can be used to improve document search
 - Verified by user studies
- Enriched models of documents considering log data
 - Central idea: using query terms and segments as features
- Tackling sparsity of log data
 - Clustering similar queries
 - Smoothing
- Many applications: generating keywords, computing importance of documents, organizing search results, considering clickthrough data in page summarization, and category maintenance

Challenges

- From document modeling to document cluster/site modeling
 - Several Web pages are often visited together
- Modeling temporal characteristics of search activities
 - Detecting bursts of new interests
- Many applications can be improved by using search/browse log data

A Road Map



Clicks and Preferences

- A user asks a query, a search engine shows a list of results
- Why does a user click on a result?
 - The result looks interesting, probably hinted by the snippet information
- Why does a user click on another result?
 - Possibly, the previous result clicked does not satisfy the user's information need
- User clickthrough data provides implicit feedback and hints about user preference on search results

Learning Preferences from Clicks

- Pair-wise versus list-wise preferences
 - Pair-wise: between pages a and b, which one is more preferable?
 - List-wise: given a set of Web pages, sort them in preference order
- Clickthrough information used in learning
 - What does a click tell us?
 - What do a series of clicks tell us?
 - What do a series queries and the corresponding clickthrough information tell us?
- Preference functions: binary, scoring function, categorical/discrete
- Applications: organic search and sponsored search

A Naïve Method

- A clicked answer is more preferable than a non-clicked answer ranked at a lower place
- For a ranking of results (d_1, \dots, d_n) and a set C of clicked results, extract a preference relation
$$d_i < d_j$$
for $1 \leq j < i, i \in C, \text{ and } j \notin C$
- Drawbacks: much information has not been used
 - No comparison between clicked answers
 - No comparison between non-clicked answers

What Do User Clicks Mean?

- For a ranking of results (d_1, \dots, d_n) and a set C of the clicked results
- (Click > Skip above) for all pairs $1 \leq j < i, i \in C$, and $j \notin C, R(d_i, d_j)$
 - (Last click > Skip above) let $i \in C$ be the rank of the link that was clicked temporally last, for all pairs $1 \leq j < i, j \notin C, R(d_i, d_j)$ [more accurate empirically]
- (Last click > No-click next) for all pairs $i \in C$ and $i + 1 \notin C, R(d_i, d_{i+1})$
- T. Joachims, L. Granka, B. Pan, H. Hembrooke, and G. Gay. Accurately interpreting clickthrough data as implicit feedback. SIGIR'05.

Kendall's τ

- How can we compare two rankings of a set of m documents?
- For two preference relations R and R' , let P be the number of concordant pairs (a, b) such that $R(a, b) = R'(a, b)$, and Q be the number of discordant pairs (a, b) such that $R(a, b) \neq R'(a, b)$

$$\tau(R, R') = \frac{P - Q}{P + Q} = 1 - \frac{2Q}{\binom{m}{2}}$$

$$- P + Q = m$$

How Good Is a Preference Relation?

- For a preference relation R , the average precision of R is bounded by

$$\text{Avg Prec}(R) \geq \frac{1}{l} \left[Q + \binom{l+1}{2} \right]^{-1} \left(\sum_{i=1}^l \sqrt{i} \right)^2$$

where l is the number of relevant documents

- Learn a preference relation R maximizing

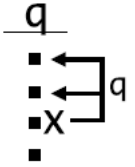
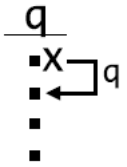
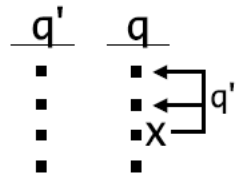
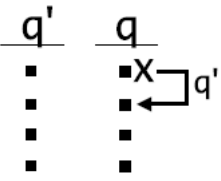
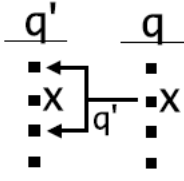
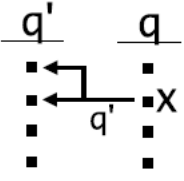
$$\int \tau(R_q, R^*) d \Pr(q, R^*)$$

- However, the ideal preference is unknown ...
 - An SVM algorithm
- T. Joachims, Optimizing search engines using clickthrough data. KDD '02.

Query Chains

- Users often reformulate their queries to approach a good representation of their information needs (for the target search engine)
 - “Lexis **Nexis**” → “Lexis Nexus”
- Query chain: a sequence of reformulated queries asked by a user
 - How can we use query chains to learn preferences?
- Filip Radlinski and Thorsten Joachims. Query chains: learning to rank from implicit feedback. KDD'05.

Feedback Strategies

<p>Click $>_q$ Skip Above</p> 	<p>Click First $>_q$ No-Click Second</p> 
<p>Click $>_{q'}$ Skip Above</p> 	<p>Click First $>_{q'}$ No-Click Second</p> 
<p>Click $>_{q'}$ Skip Earlier Query</p> 	<p>Click $>_{q'}$ Top Two Earlier Query</p> 

Filip Radlinski and Thorsten Joachims. Query chains: learning to rank from implicit feedback. KDD'05.

Example

$\frac{q1}{d1}$	$\frac{q2}{d4}$ x
d2 x	d5
d3	d6

$d_2 >_{q1} d_1$	$d_4 >_{q2} d_5$	$d_4 >_{q1} d_5$
$d_4 >_{q1} d_1$	$d_4 >_{q1} d_3$	

Filip Radlinski and Thorsten Joachims. Query chains: learning to rank from implicit feedback. KDD'05.

Using Aggregated Clickthrough Data

- The preferences learned from individual user clickthrough data may not be highly reliable
- Using intelligence of crowd – aggregating clickthrough data from many users
 - Let $\text{click}(q, d)$ be the corresponding aggregate click frequency of document d with respect to query q
 - Let $\text{cdif}(q, d_i, d_j) = \text{click}(q, d_i) - \text{click}(q, d_j)$
- If $\text{cdif}(q, d_i, d_j) > 0$, $d_i >_q d_j$
- Z. Dou, R. Song, X. Yuan, J-R Wen. Are click-through data adequate for learning web search rankings? CIKM'08.

Presentation Bias

- A user is more likely to click on documents presented higher in the result set irrespective of relevance
 - T. Joachims, L. Granka, B. Pan, H. Hembrooke, and G. Gay. Accurately interpreting clickthrough data as implicit feedback. SIGIR'05.
- A simple FairPairs algorithm
 - Let $R = (d_1, \dots, d_n)$ be the results for some query
 - Randomly choose $k \in \{0, 1\}$ with uniform probability
 - If $k = 0$ ($k = 1$), for all odd (even) numbers i , swap d_i and d_{i+1} with probability 0.5
 - Present R to the user, recording clicks on results
 - Every time the lower result in a pair that was considered for flipping is clicked, record this as a preference for that result over the one above it
- Filip Radlinski and Thorsten Joachims. Minimally invasive randomization for collecting unbiased preferences from click-through data. AAAI'08.

Why FairPairs Works?

- Let c_{ij} be the number of times a user clicks on d_i when d_j is presented just above d_i
- FairPairs designs the experiment such that c_{ij} is the number of votes for $(d_i > d_j)$ and c_{ji} is the number of votes for $(d_j > d_i)$
 - The votes are counted only if the results are presented in equivalent ways
- Both sets of votes are affected by presentation bias in the same way
- Filip Radlinski and Thorsten Joachims. Minimally invasive randomization for collecting unbiased preferences from click-through data. AAAI'08.

Passive Learning

- A user often considers only the top-ranked answers, and rarely evaluates results beyond the first page
 - The clickthrough data collected passively is strongly biased toward documents already ranked highly
- Highly relevant results not initially ranked highly may never be observed and evaluated
- F. Radlinski and T. Joachims. Active exploration for learning rankings from clickthrough data. KDD'07.

Active Exploration for Learning

- Idea: presenting to users a ranking optimized to obtain useful feedback
- A naïve method: intentionally present unevaluated results in the top few positions
 - May hurt user satisfaction
- A principled approach: using a Bayesian approach
- F. Radlinski and T. Joachims. Active exploration for learning rankings from clickthrough data. KDD'07.

Clickthrough for Sponsored Search

- Preference learning problem also exists for sponsored search
 - Which ads are more likely to be clicked by a user with respect to a query?
- Machine learning approaches can be used
 - How to use click data for training and evaluation?
 - Which learning framework is more suitable for the task?
 - Which features are useful for existing methods?
- Ciaramita, M., et al. Online learning from click data for sponsored search. In WWW'08, 2008.

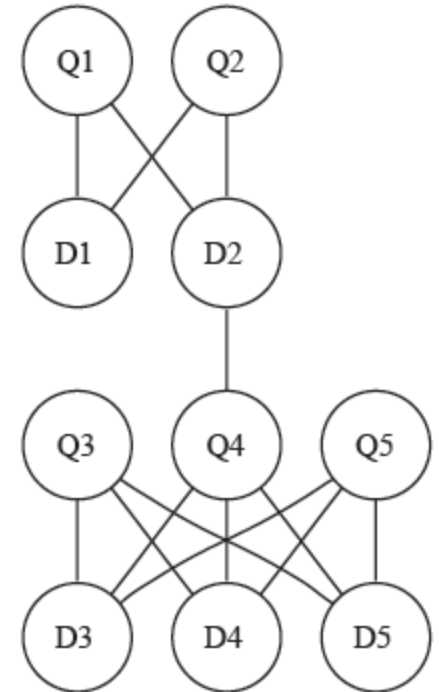
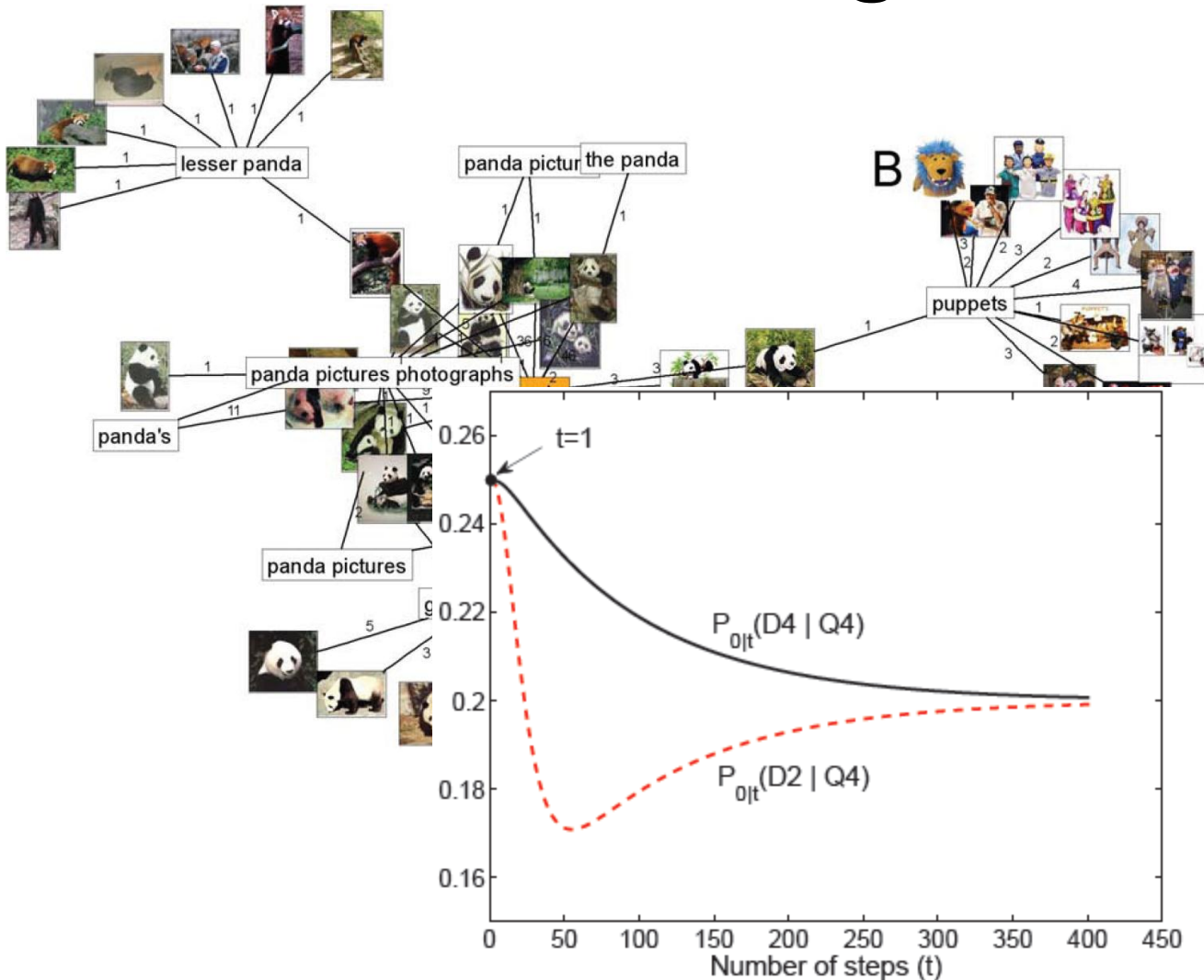
Learning Listwise Preferences

- Pairwise preferences are easy to learn, but may not generate a ranked list
 - Given $a > b$, $b > c$, and $c > a$, no ranking can be generated
- Learning listwise preferences: for a given query, produce a ranking of documents
 - Using listwise preferences a search engine can retrieve relevant documents that have not yet been clicked for that query, and rank those documents effectively

A Markov Random Walk Method

- Query-document bipartite graph
- The random walk process
 - A user imagines a single document to represent the user's information need, and thinks of a query associated with the document, and issues the query
 - Alternatively, the query makes the user imagine another document, and that document makes the user imagine another query
- The model produces a probabilistic ranking of documents for a query
- N. Craswell and M. Szummer. Random walks on the click graph. SIGIR'07.

Clustering Effect

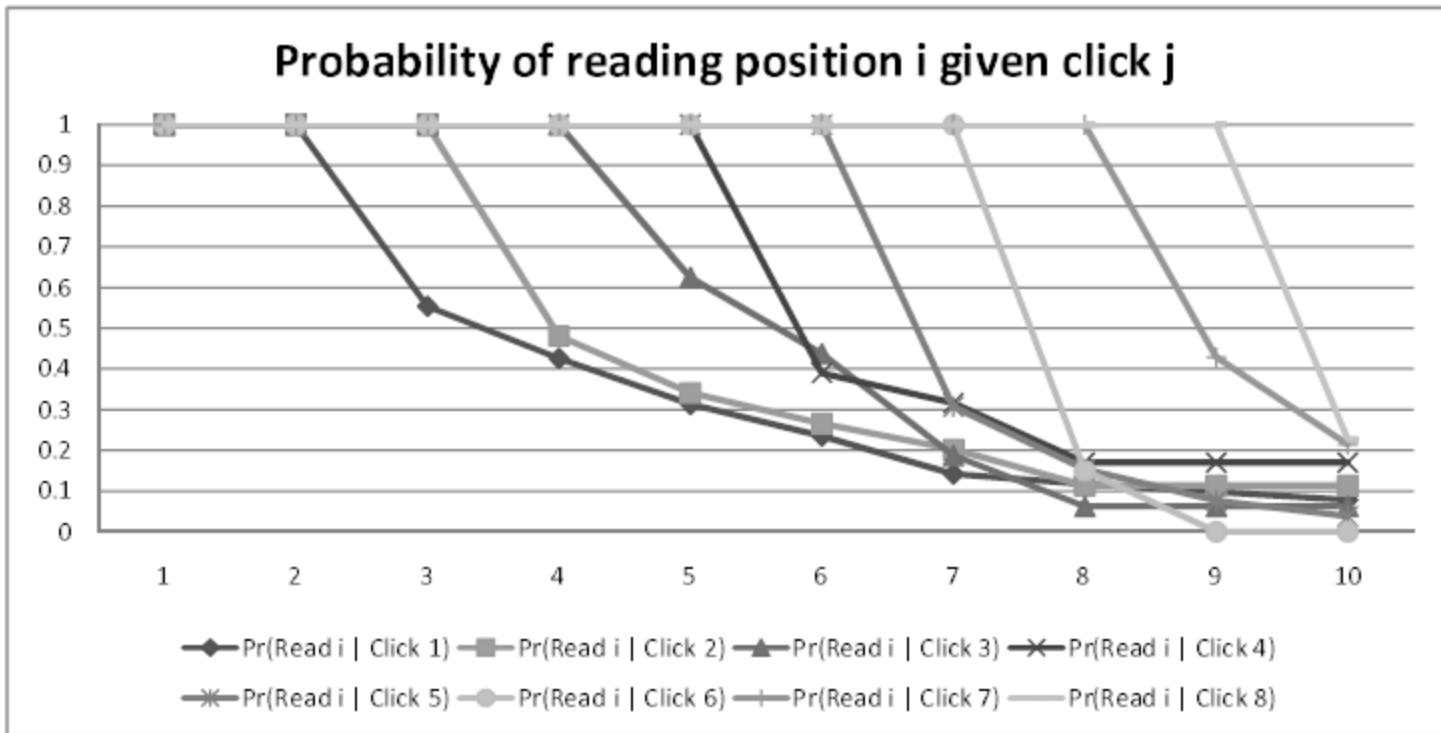


N. Craswell and M. Szummer. Random walks on the click graph. SIGIR'07.

Learning from Labeled Data

- In search engines, a ranking function is learned from labeled training data
 - A training example is a (query, URL) pair labeled by a human judge who assigns a score of “perfect”, “excellent”, etc.
- Clickthrough data can be used to generate good labels automatically
 - Generate preferences between URLs for a given query with probability proportional to the probability a user reads position i given that the user clicks on position j
 - Create a per query preference graph: vertices are URLs, and a directed edge $u \rightarrow v$ indicates the number of users who read u and v , clicked u and skipped v
- R. Agrawal et al. Generating labels from clicks. WSDM'09.

Click-Read Probability



The probability a user reads position i given that the user clicks on position j

R. Agrawal et al. Generating labels from clicks. WSDM'09.

Computing Labels

- Using pairwise preferences
- Given a directed graph $G(V, E)$, and an ordered set A of K labels, find a labeling L such that the net agreement weight is maximized

$$A_G(L) = \sum_{u \rightarrow v} w_{u \rightarrow v} - \sum_{u \rightarrow v} w_{v \rightarrow u}$$

- NP-hard in general
 - Can be solved in time $O(|E|)$ when $K = 2$
-
- R. Agrawal et al. Generating labels from clicks. WSDM'09.

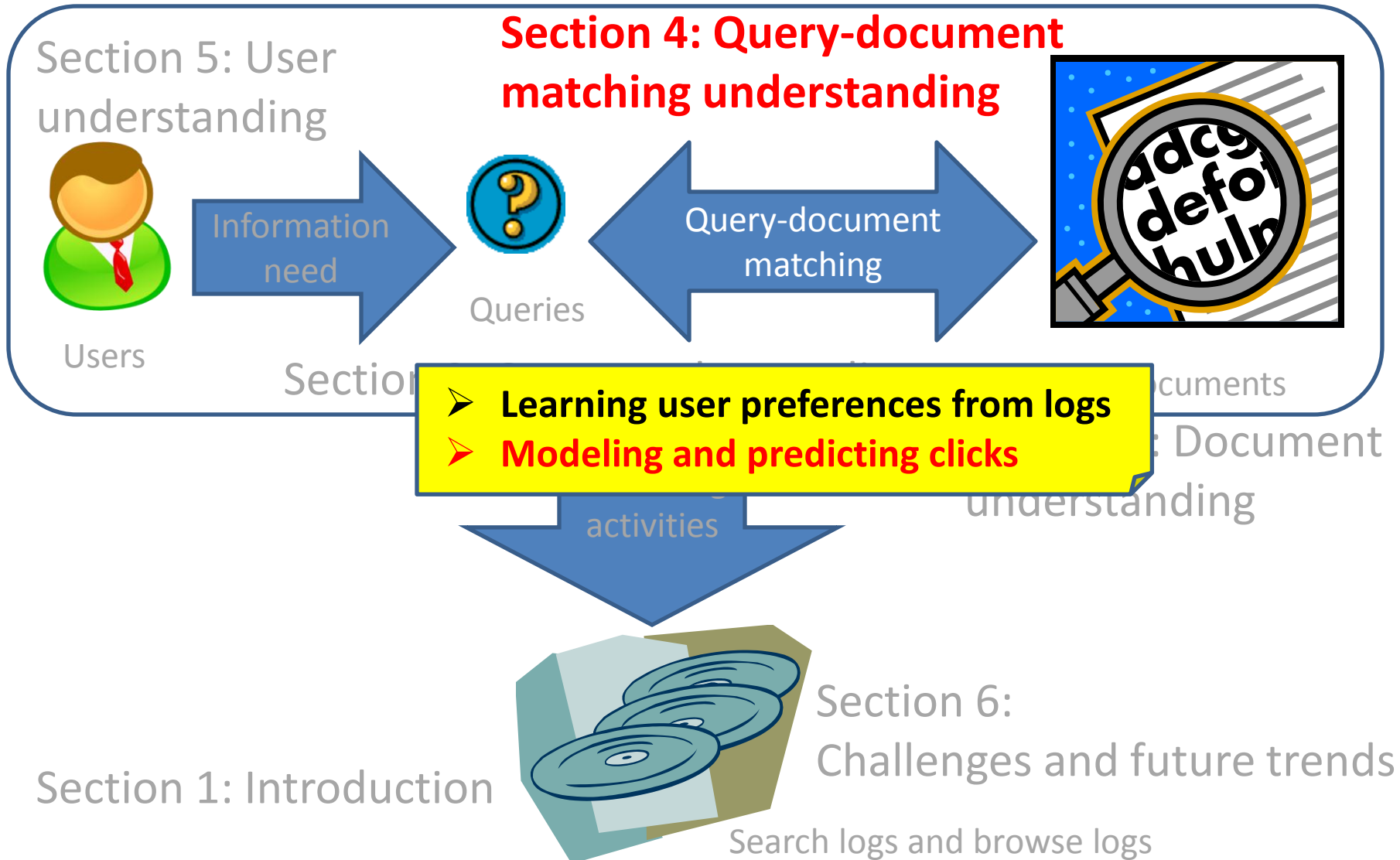
Summary

- User clickthrough data provides implicit feedback and hints about user preference on search results
- Pair-wise versus list-wise preferences
- Clickthrough information used in learning
 - A click \rightarrow a series of clicks \rightarrow a series queries and the corresponding clickthrough
- Preference functions: binary, scoring function, categorical/discrete
- Applications: organic search and sponsored search

Challenges

- There are still many problems remained open
- How to learn preferences effectively about rare queries and documents?
- Context-aware preference learning
 - Query “digital camera”
 - About Cannon versus Nikon, different users may have different preferences – how can we detect the preferences?
- Temporal and burst sensitive preferences
 - Query “Obama”
 - More recent events may be more preferable
 - Some milestone events (e.g., medical insurance bill) may be more preferable
 - How to model, learn, and apply such preferences?

A Road Map



Click Bias on Presentation Order

- The probability of click is influenced by the position of a document in the results page
- Click bias modeling: how probability of click depends on positions
 - Probability $P(c | r, u, q)$ that a document u presented at position r is clicked by a user who issued a query q
- A related problem CTR modeling/prediction
 - CTR: number of clicks per display
 - CTR can be used to select the best document in some applications such as the Today module on Yahoo!
Front page

Baseline/Examination Hypotheses

- Baseline: no bias associated to the document positions
 - $P(c|r, u, q) = P(a|u, q)$, where $P(a|u, q)$ is the attractiveness of document u as a result of query q
- The examination/separability hypothesis: users are less likely to look at results at lower ranks – each rank has a certain probability $P(e|r)$ of being examined
 - $P(c|r, u, q) = P(e|r)P(a|u, q)$
 - When $P(e|r) = 1$, we obtain the baseline
- Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. An experimental comparison of click position-bias models. WSDM'08.

The Cascade Model

- Users view search results from top to bottom, deciding whether to click each result before moving to the next
 - Each document is either clicked with a probability $P(a | u, q)$ or skipped with a probability $1 - P(a | u, q)$
 - A user clicks never comes back; a user skips always continues

$$P(c | r, u, q) = P(a | u, q) \prod_{i=1}^{r-1} (1 - P(a | u_i, q))$$

- Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. An experimental comparison of click position-bias models. WSDM'08.

Empirical Study

- The cascade model performs significantly better than the other models for clicks at higher ranks, but slightly worse than the other models for clicks at lower ranks
- What does the cascade model capture?
 - Users examine all documents sequentially until they find a relevant document and then abandon the search
- Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. An experimental comparison of click position-bias models. WSDM'08.

What Are Not Modeled Yet?

- What is the possibility that a user skips a document without examining it
- In informational queries, a user may examine documents after the first click – what is the possibility?
 - In navigational queries, a user tends to stop after the first relevant document is obtained
- We need a user browsing model
- Georges E. Dupret, Benjamin Piwowarski. A user browsing model to predict search engine click data from past observations. SIGIR'08.

The Single Browsing Model

- The probability that a user examines a document depends on the distance from the document to the last click
 - Rationale: a user tends to abandon the search after seeing a long sequence of unattractive snippets
- Assuming both attractiveness and examination be Bernoulli variables
- Georges E. Dupret, Benjamin Piwowarski. A user browsing model to predict search engine click data from past observations. SIGIR'08.

The Single Browsing Model

- Assuming both attractiveness and examination be Bernoulli variables
 - $P(a|u, q) = \alpha_{uq}^a (1 - \alpha_{uq})^{1-a}$
 - $P(e|r, d) = \gamma_{rd}^e (1 - \gamma_{rd})^{1-e}$
 - α_{uq} is the probability of attractiveness of snippet u if presented to a user who issued query q
 - γ_{rd} is the probability of examination at distance d and position r
- The full model: $P(c, a, e|u, q, d, r) = P(c|a, e)P(e|d, r) P(a|u, q) = P(c|a, e) \gamma_{rd}^e (1 - \gamma_{rd})^{1-e} \alpha_{uq}^a (1 - \alpha_{uq})^{1-a}$
- Georges E. Dupret, Benjamin Piwowarski. A user browsing model to predict search engine click data from past observations. SIGIR'08.

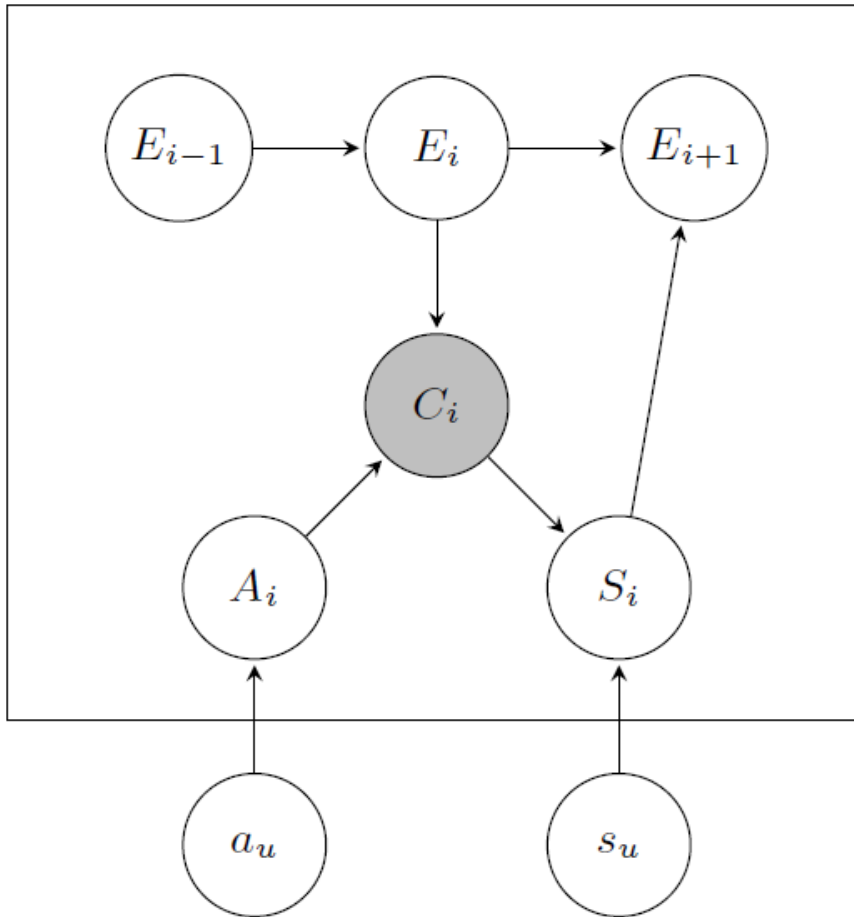
Multiple Browsing Model

- Navigational versus informational queries
 - In general, there may be a variety of many kinds of user behaviors
- Build a mixture of single browsing models, and use a latent variable m to indicate which is used for a particular query q
 - $P(e|r, d, m) = \gamma_{rdm}^e (1 - \gamma_{rdm})^{1-e}$
- Georges E. Dupret, Benjamin Piwowarski. A user browsing model to predict search engine click data from past observations. SIGIR'08.

Logistic Model

- Model the logarithm of the odds of a click
 - Odds = $P(c=1 | r, d, u, q) / (1 - P(c=1 | r, d, u, q))$
- The logarithms of the odds are regressed against the explanatory variable
 - $\ln \text{odds} = \beta_{uq} + \beta_{rd}$
 - Odds = $\exp(\beta_{uq}) + \exp(\beta_{rd})$
- Georges E. Dupret, Benjamin Piwowarski. A user browsing model to predict search engine click data from past observations. SIGIR'08.

A Dynamic Bayesian Network Model



For a given position i , C_i is the only observed variable indicating whether there was a click or not at this position. E_i , A_i , and S_i are hidden binary variables modeling whether the user examined the URL, the user was attracted by the URL, and the user was satisfied by the landing page, respectively

Chapelle, O. and Zhang, Y. A Dynamic Bayesian Network Click Model for Web Search Ranking. WWW'09.

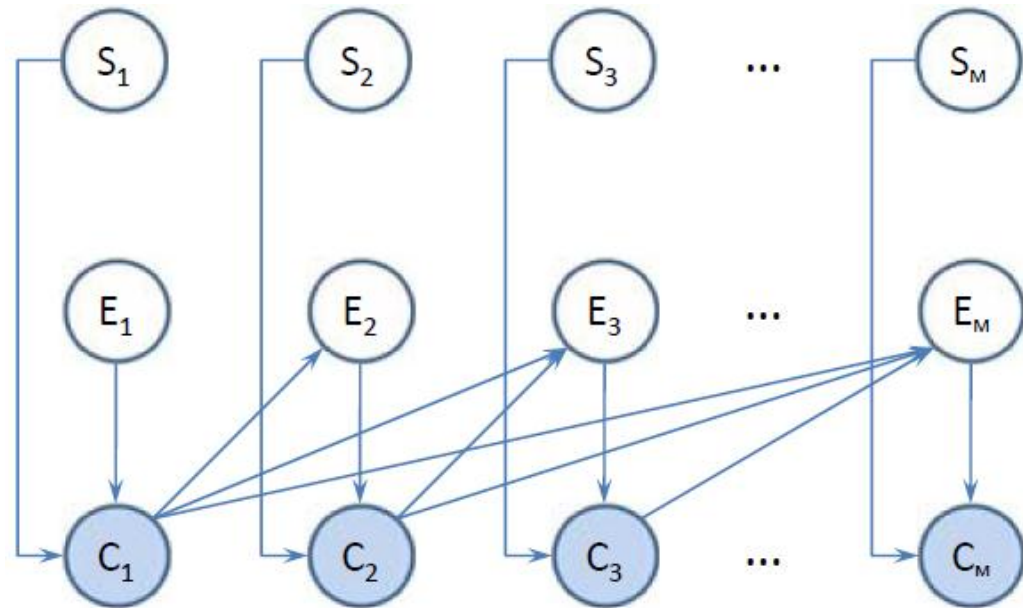
Handling Huge Amounts of Data

- Scalability: how to handle terabyte- or even petabyte-scale data
- Parallelizability: can a model be implemented in a parallelizable way?
- Incremental updatability: can it be single-pass computable?

- Chao Liu, Fan Guo, Christos Faloutsos. BBM: bayesian browsing model from petabyte-scale data. KDD'09.

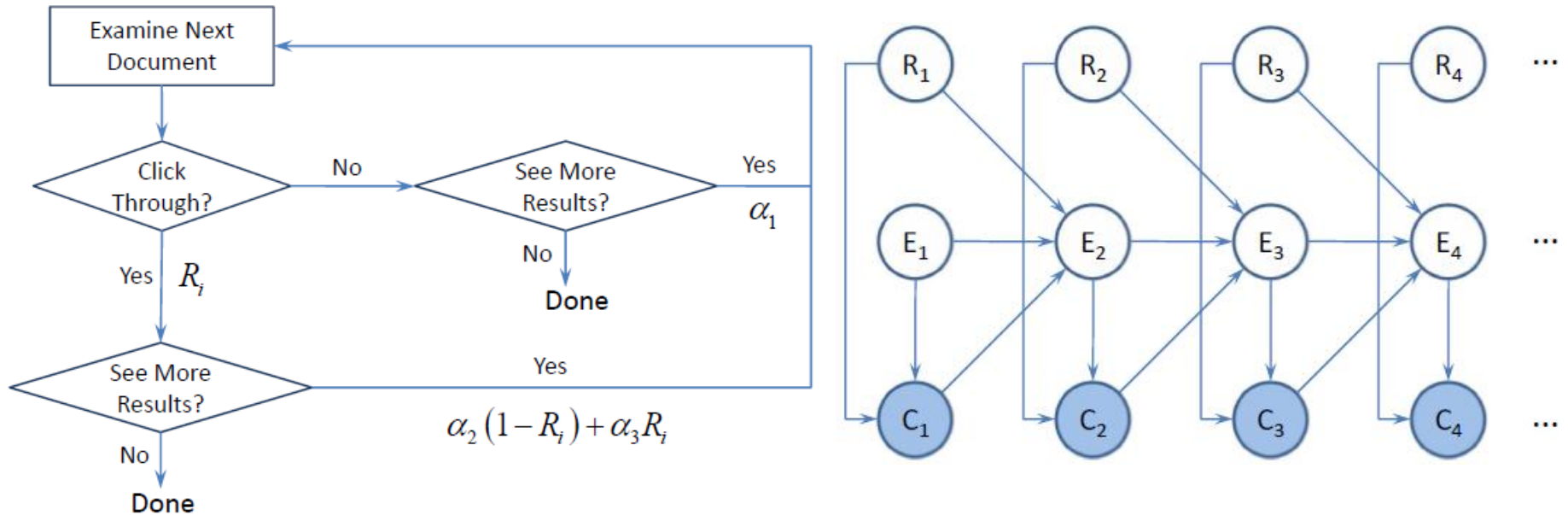
BBM: A Bayesian Browsing Model

- A single pass suffices for computing global parameters and inferring document relevance
- Exact posterior for document relevance can be derived in closed form



Chao Liu, Fan Guo, Christos Faloutsos. BBM: bayesian browsing model from petabyte-scale data. KDD'09.

Click Chain Model



The generative process

The graphical model representation

R_i is the relevance variable of d_i at position i , and α 's form the set of user behavior parameters

Fan Guo, et al. Click chain model in web search. WWW'09.

CTR Modeling/Prediction for Ads

- $CTR = P(\text{click} | \text{ad}, \text{pos}) = P(\text{click} | \text{ad}, \text{seen})P(\text{seen} | \text{pos})$
- Using logistic regression, we have
$$CTR = \frac{1}{1 + e^{-\sum_i w_i f_i(ad)}}$$
 - $f_i(ad)$ is the value of the i -th feature for the ad, and w_i is the learned weight for that feature
- Features
 - Term CTR: the CTR of other ads that have the same bid terms
 - Related term CTR: the CTR of the ads bidding on “buy red shoes” is related to the CTR of the ads bidding on “red shoes”
 - ...
- Matthew Richardson, Ewa Dominowska, Robert Ragno. Predicting clicks: estimating the click-through rate for new Ads. WWW'07.

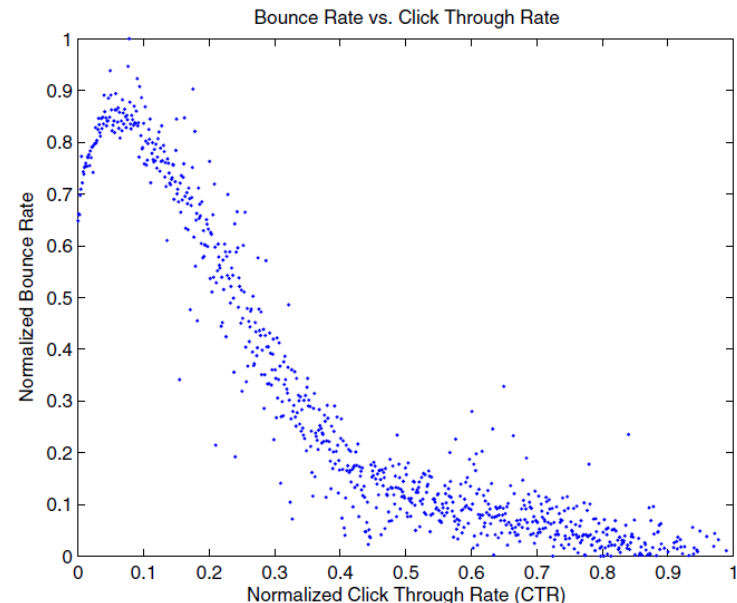
Spatio-Temporal Models

- For a fixed location over time, use a dynamic Gamma-Poisson model
- Combine information from correlated locations through dynamic linear regressions
- Deepak Agarwal, Bee-Chung Chen, Pradheep Elango. Spatio-temporal models for estimating click-through rate. WWW'09.

Bounce Rates

- For an ad, the bounce rate is the fraction of users who click on the ad but almost immediately move on to other tasks
 - A poor bounce rate leads to poor advertiser return on investment and poor search engine user experience following the click

Sculley, D., et al. Predicting bounce rates in sponsored search advertisement. KDD'09.



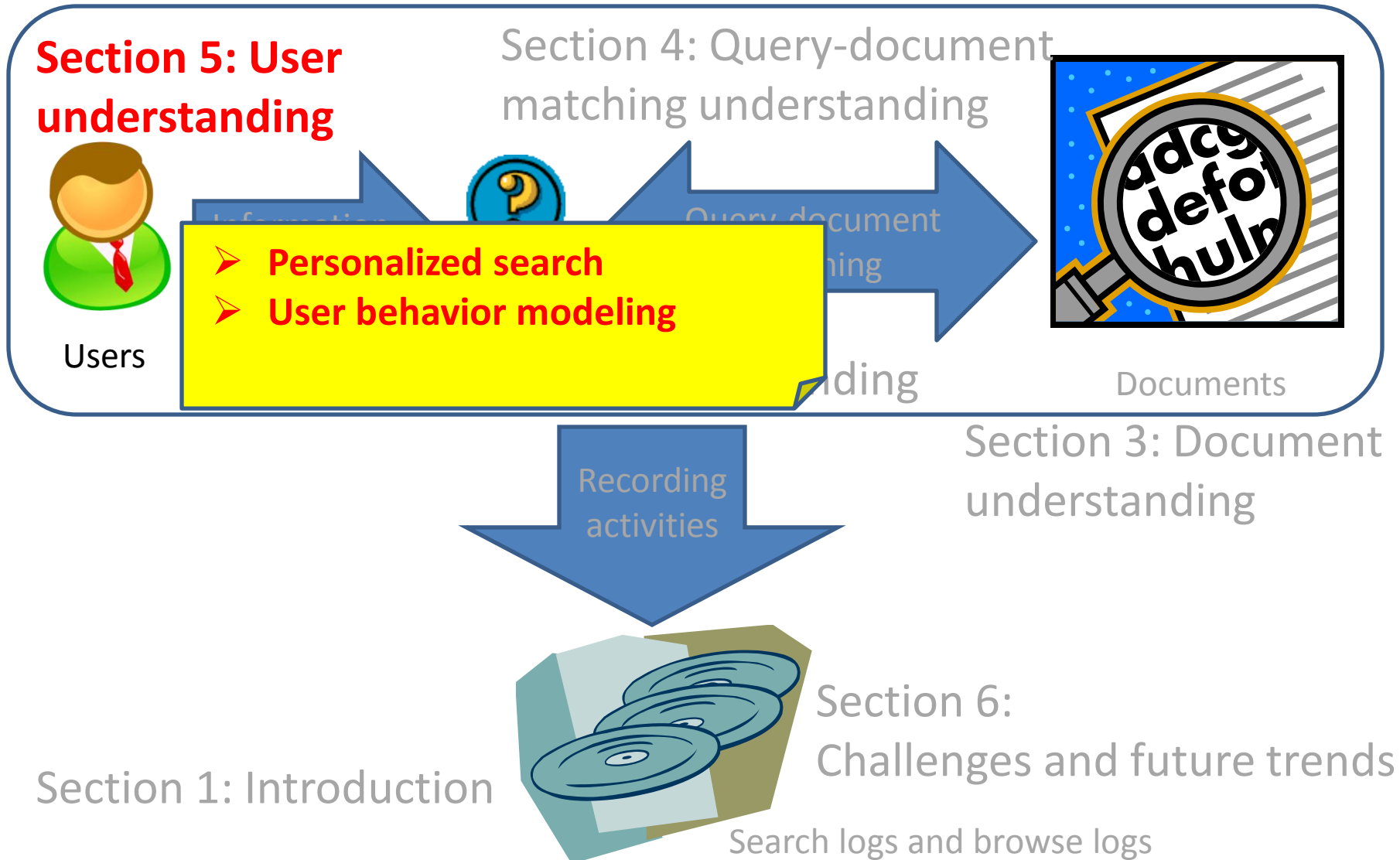
Bounce Rate Prediction

- Features
 - Parsed terms, extracted from content, scored using TF-IDF
 - Related terms, derived from the parsed terms using a transformation similar to term expansion via latent semantic analysis (LSA)
 - Cluster/category membership, the strength of similarity of a given piece of content to a set of topical clusters and a semi-automatically constructed hierarchical taxonomy
 - Shannon Redundancy, how focused a piece of content is
 - Binary cosine similarity between content groups
 - Binary KL-divergence for term-based relevance
- Using a logistic regression approach
- Sculley, D., et al. Predicting bounce rates in sponsored search advertisement. KDD'09.

Summary

- Click-bias on presentation order
 - Click (bias) modeling and CTR prediction
- Click models
 - Examination hypothesis
 - Cascade model
 - Single/multiple browsing models, logistic model
 - Dynamic Bayesian Network model
 - BBM/Click chain model: scalability, exact inference-ability, and parallelizability
- CTR prediction
 - Spatio-temporal models
 - Bounce rate prediction

A Road Map



User Understanding

What users searched or browsed in the past

How users searched or browsed in the past

Personalization

User Behavior Modeling

- Deriving behavioral features
- Designing sequential models

Predicting users' preferences
Recommending queries and URLs

...

...

Predicting users' search satisfaction
Predicting users' next activity

...

...

Outline for Personalized Search

- Introduction to personalized search
- Three questions for personalized search
 - Which personalization methods work better?
 - How much is personalization feasible in Web search?
 - When is personalization effective in Web search?

Personalized Search

- Different users may have different intents behind the same query
 - Example “GMC”
 - The mix-and-for-all method may not be an optimal solution



General
Medical
Council

Regulating doctors
Ensuring good medical practice

Contextualization and Individualization

- Personalized search and context-aware search
- Following Pitkow, et al., personalization includes *individualization* and *contextualization*
 - Individualization: the totality of characteristics that distinguishes an individual
 - Often creates a profile for each user from long history
 - Contextualization: the interrelated conditions that occur within an activity
 - Often leverages the user's previous search/browse information within the same session

How Personalization Helps Search

- Consider the query “GMC”
 - Individualization: if the user profile shows that the user often raises medical-related queries or browses medical-related Web pages, it is more likely the user is searching for General Medical Council
 - Contextualization: if the user inputs query “Honda” and “Nissan” before “GMC” in the same session, it is more likely the user is searching for GMC cars

Approaches to Personalized Search

- Individualization
 - Create a profile for each user from a long history
 - Topic-based profile, e.g., [Pretschner99][Liu02][Pitkow02][Speretta05][Qiu06]
 - Term-based profile, e.g., [Teevan05][Tan06]
 - Click-based profile, e.g., [Teevan07]
- Contextualization
 - Use previous search/browse info in the same session
 - Use previous queries/clicks in the same session, e.g., [Shen05][Cao09]
 - Use previous browsed pages in the same session, e.g., [White09]

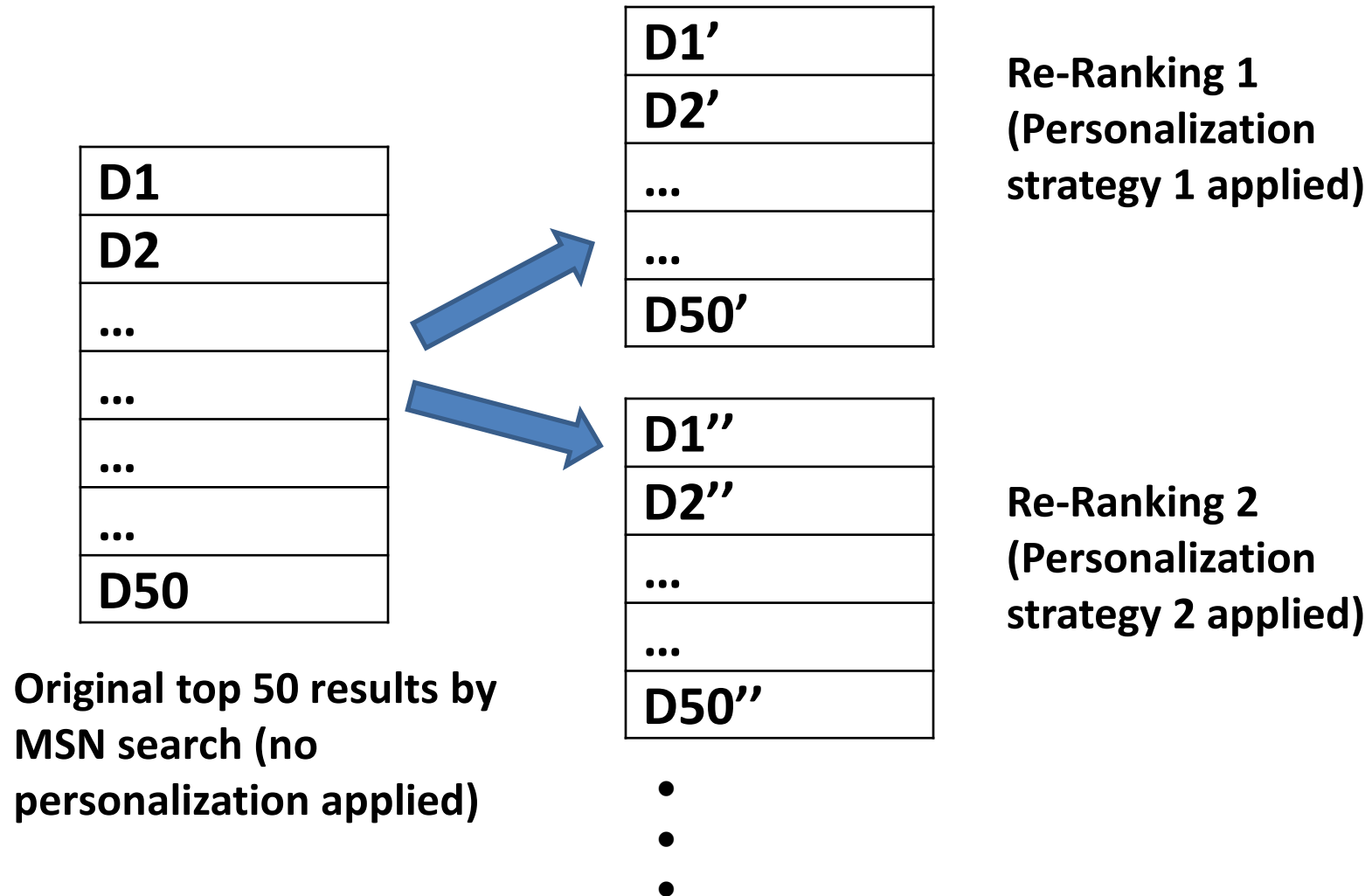
Outline for Personalized Search

- Introduction to personalized search
- Three questions for personalized search
 - Which personalization methods work better?
 - How much is personalization feasible in Web search?
 - When is personalization effective in Web search?

A Large-scale Evaluation and Analysis of Personalized Search Strategies

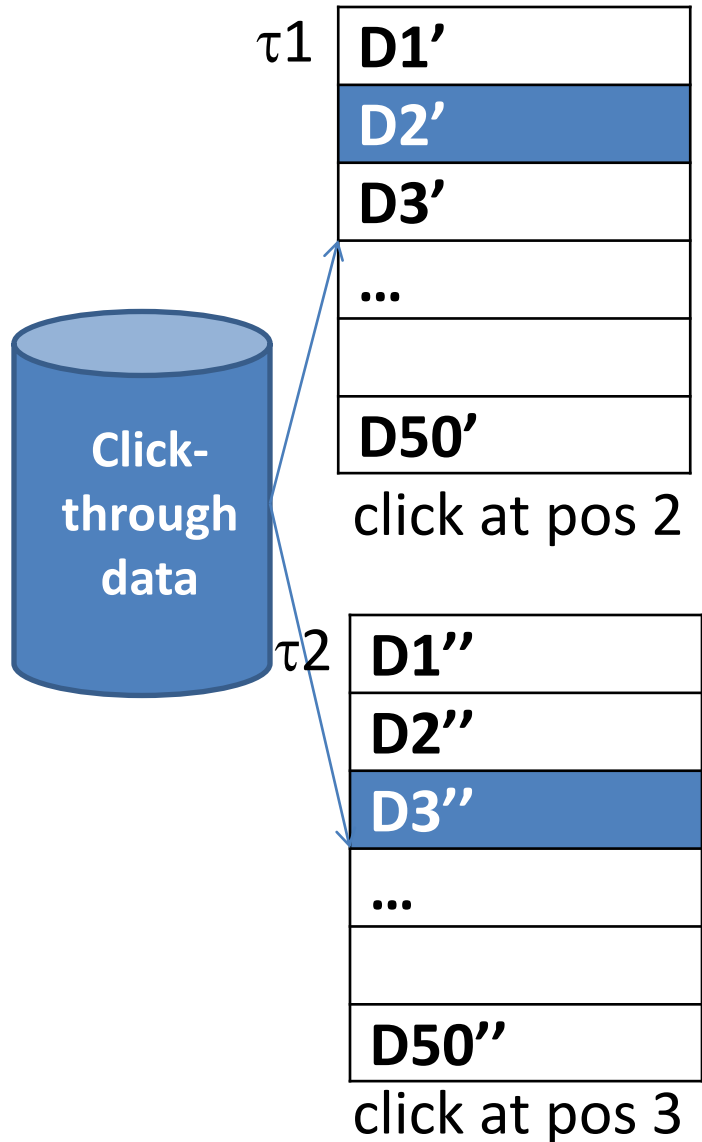
- Zhicheng Dou, Ruihua Song and Ji-Rong Wen, WWW' 2007
- 12 days MSN search logs
 - 11 days for training and 1 day for testing
- 10,000 randomly sampled users
- 4,639 test queries

Methodology (Step 1): Re-Ranking



Dou, Z. et al. A large-scale evaluation and analysis of personalized search strategies. WWW' 2007.

Methodology (Step 2): Evaluation



- Evaluation Measurements
 - Given two re-ranked lists τ_1 and τ_2 , if the clicked documents are ranked higher in τ_1 than in τ_2 , then τ_1 is better than τ_2
 - Metrics
 - Rank scoring (the larger the better)
 - Average Rank (the smaller the better)

Dou, Z. et al. A large-scale evaluation and analysis of personalized search strategies. WWW' 2007.

Five Strategies

- Topic-based strategies
 - Strategy 1: Topic-based individualization
 - Strategy 2: Topic-based contextualization
 - Strategy 3: A combination of topic-based individualization and contextualization
- Click-based strategies
 - Strategy 4: Click-based individualization
 - Strategy 5: A smoothing method of click-based individualization

Topic-Based Strategies

- Individualization

- Create a topic profile $c_l(u)$ for each user u



Probability of user interested in a topic

- The profile is aggregated from all visited pages of u

- For each result p of query q , create a topic vector $c(p)$



Probability of page belonging to a topic

- Personalized score: $S^L(q, p, u) = \frac{c_l(u) \cdot c(p)}{\|c_l(u)\| \|c(p)\|}$

- Contextualization

- Replace the topic profile with topic context

- The context is aggregated from the visited pages of u only within the current session

- Combination $S^{LS}(q, p, u) = \theta S^L(q, p, u) + (1 - \theta) S^S(q, p, u)$

Click-Based Strategies

- Users tend to click on the results they clicked before
 - Personalized score $s^C(q, p, u) = \frac{|Clicks(q, p, u)|}{|Clicks(q, :, u)| + \beta}$
 - A user's history could be sparse
- Users tend to click on the results which were clicked by similar users before
 - Personalized score $s^{GC}(q, p, u) = \frac{\sum_{u'} Sim(u, u') |Clicks(q, p, u')|}{\beta + \sum_{u'} Sim(u, u') |Clicks(q, :, u')|}$
 - $Sim(u, u')$ is the similarity between the topic profiles of u and u'

Experimental Results

	Method	Ranking Score	Average Rank
Baseline	MSN search	69.4669	3.9240
Click-based	Click-based	70.4350	3.7338
	Group click-based	70.4168	3.7361
Topic-based	Long history	66.7378	4.5466
	Short history	66.7822	4.4244
	Combined profile	68.5958	4.1322

- Click-based strategies have better performance
- A combined approach is better than purely individualization or contextualization
- Topic-based strategies do not perform well
 - Possible reasons: simple implementation, simple user profiles, insufficient search histories, noises in user search histories

Remaining Questions

- With better implementation, how will the topic-based strategies perform?
- Why a combination of long and short history works better?
- How is the coverage for different strategies?
- How is the correlation between the implicit measures from click-through data and the explicit user labeling?

Need more works comparing different personalization strategies.

Outline for Personalized Search

- Introduction to personalized search
- Three questions for personalized search
 - Which personalization methods work better?
 - How much is personalization feasible in Web search?
 - When is personalization effective in Web search?

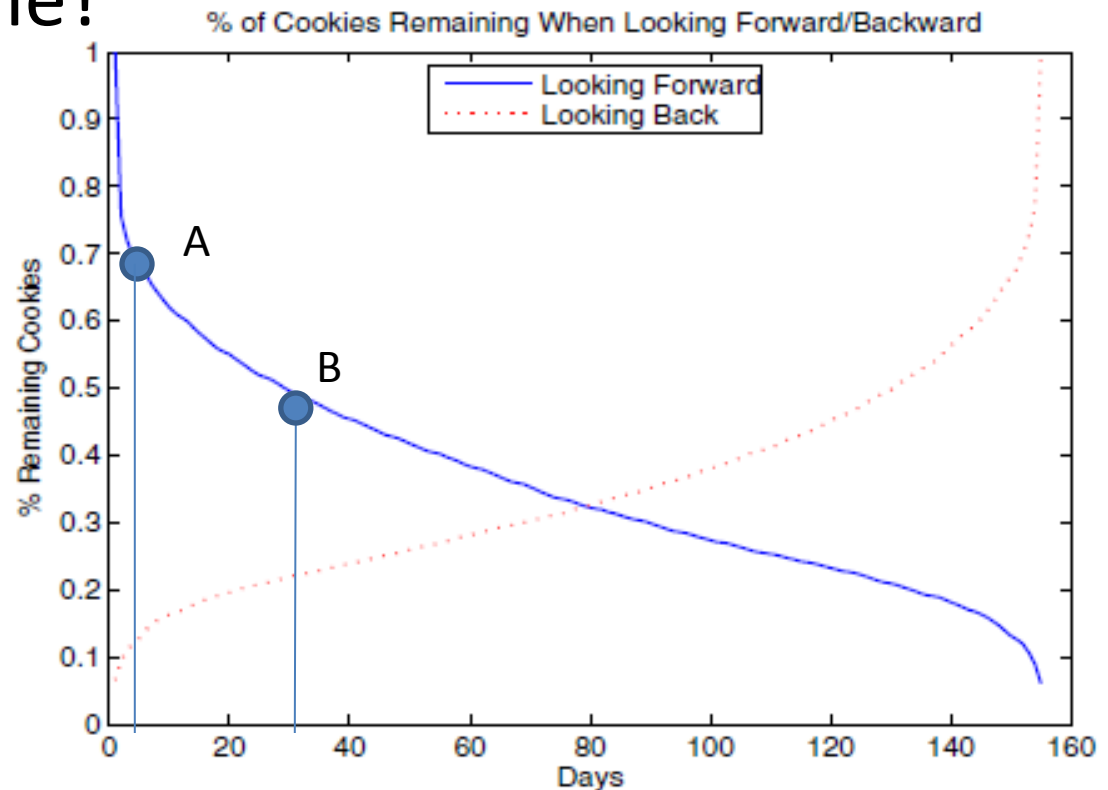
Assumptions in Personalized Search

- Individualization
 - Each user has sufficiently long history to create user profile
 - Each user has consistent interests over time, while different users' interests vary
- Contextualization
 - A short history of a user is available as context information
 - A user's information need does not change within a session

User Persistence

- Each user has sufficiently long history to create user profile?

Although 30% cookies expire after the first day (point A) over 40% cookies persist for at least a month (point B)



User Topic Interests

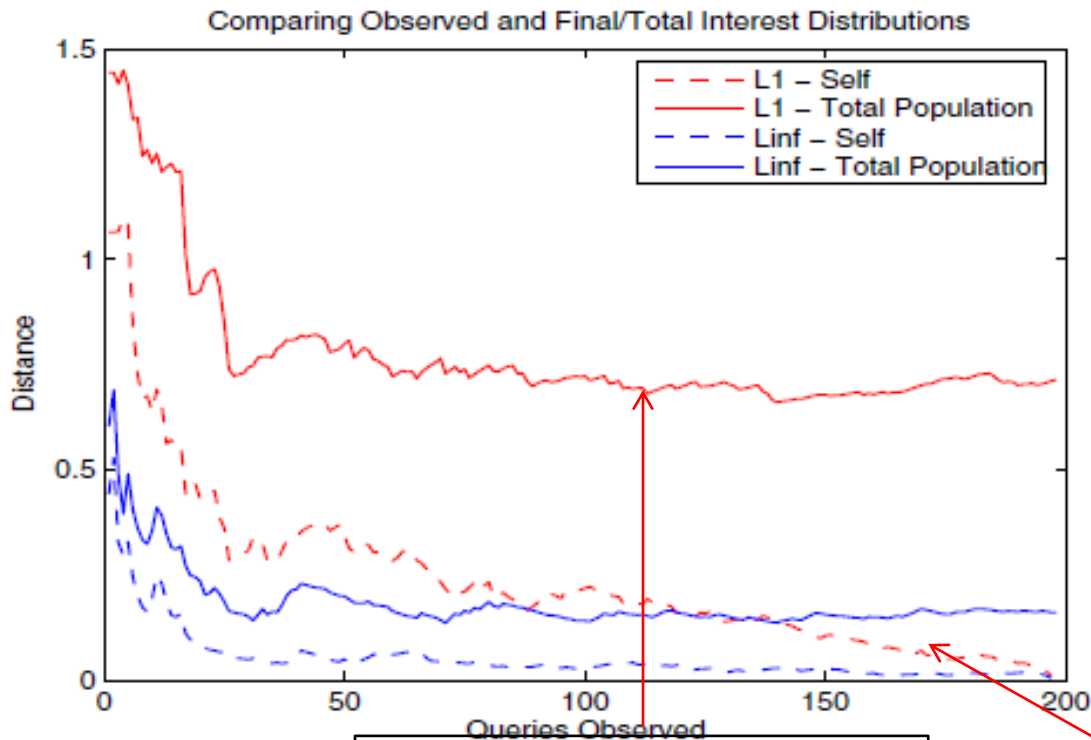
- Each user has consistent interests over time, while different users' interests vary?
 - 22 general topics (“Travel”, “Computing” ...)
 - Create a topic vector for each query q



- User distribution: F
 - Aggregate from all queries of a user
- Cumulative distribution: $C(K)$
 - Aggregate from the first K queries of a user
- Global distribution: G
 - Aggregate from all queries of all users

Probability of q
belonging to a
topic

Consistence and Distinctiveness



Queries issued by different users have different topics

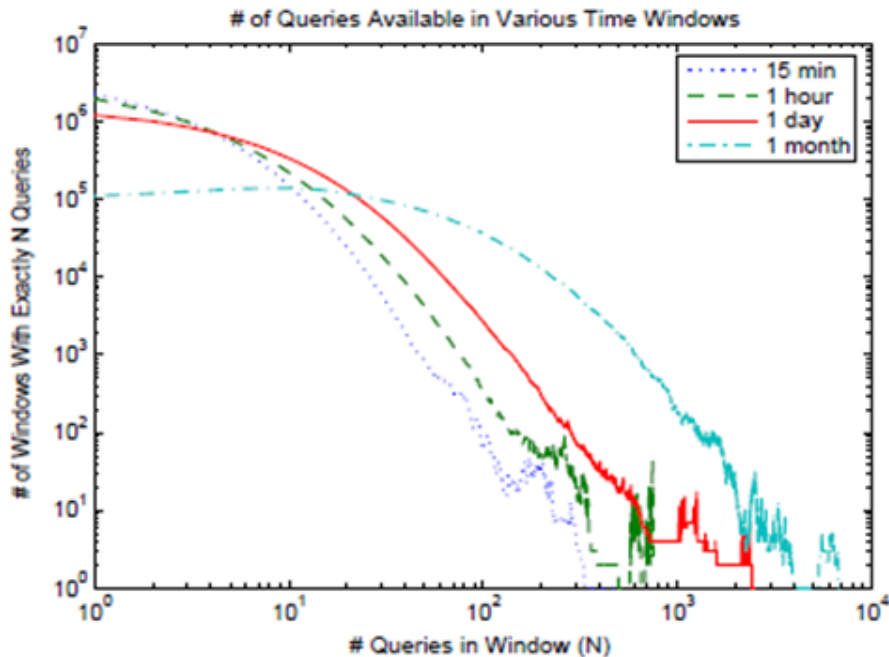
Queries issued by the same user have consistent topics

- The red dotted curve: how the L1 distance between F and C(K) changes with K
- The red solid curve: how the L1 distance between G and C(K) changes with K

Wedig S. and Madani, O. A large-scale analysis of query logs for assessing personalization opportunities. KDD' 06.

Context Availability

- A short history of a user is available as context information?



- 60% of 15-minute time window contains at least two queries [Wedig06]
- About 50% of sessions (30 minutes time out) have at least 2 queries [Cao09]
- Average session length is 1.6-3.0 [see “Query Statistics” part of this tutorial]

30%~60% of queries have previous searches in the same session as context information

Topic Consistency in Sessions

- A user's information need does not change within a session?
- Depending on how sessions are derived
 - Simple timeout threshold: ~70% accuracy
 - More complex features: >90% accuracy

Jones, R. and Klinkner K.L. Beyond the session timeout: Automatic hierarchical segmentation of search topics in query logs. CIKM'08.

Assumptions in Personalized Search

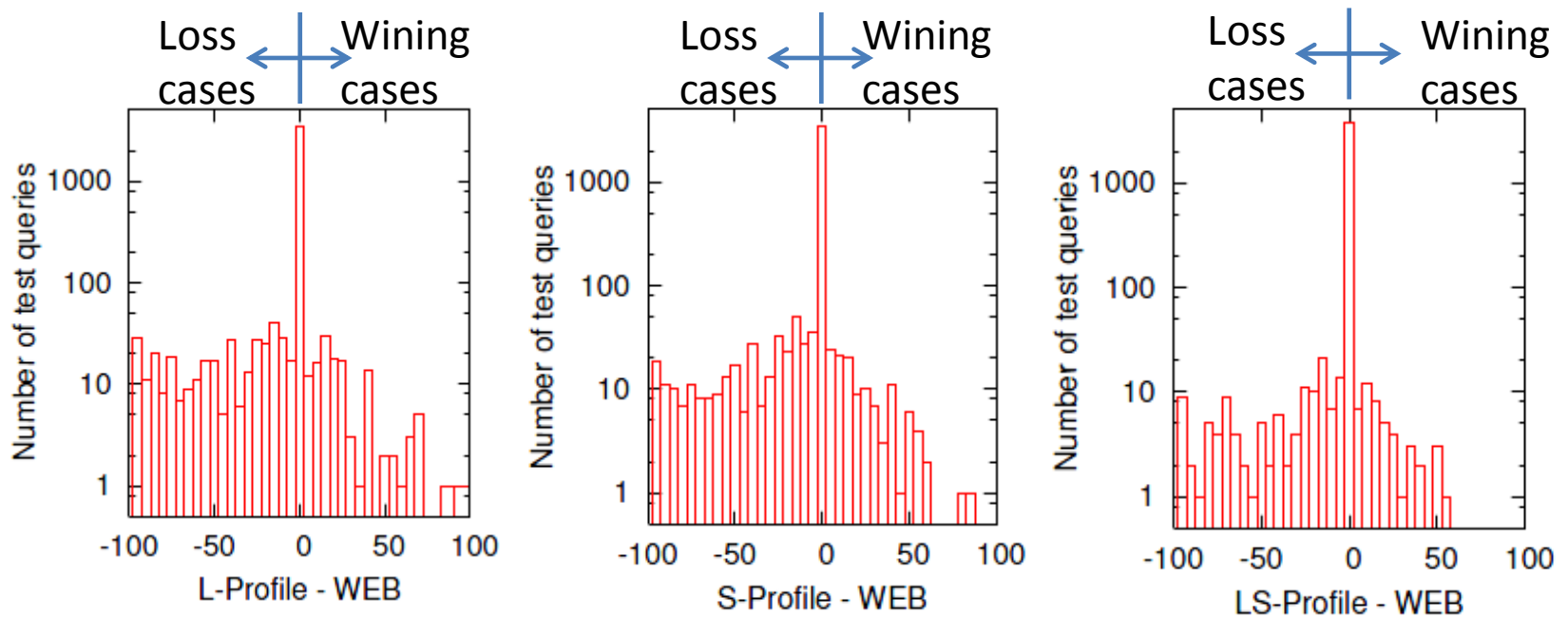
- Individualization
 - Each user has sufficiently long history to create user profile (>40% [Wedig06])
 - Each user has consistent interests over time, while different users' interests vary (Yes [Wedig06])
- Contextualization
 - A short history of a user is available as context information (30%-60% [Wedig06][Cao09])
 - A user's information need does not change within a session (depending on session segmentation: 70%~90% [Jones08])

Outline for Personalized Search

- Introduction to personalized search
- Three questions for personalized search
 - Which personalization methods work better?
 - How much is personalization feasible in Web search?
 - **When is personalization effective in Web search?**

Case Studies on Personalization Strategies

- Each personalization strategy benefits some queries (wining cases), but harms others (loss cases)
- Can we automatically recognize the wining cases and apply personalization only to those cases?



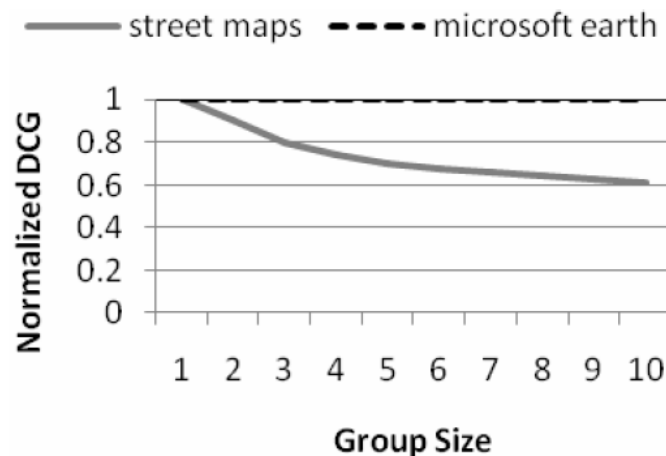
Click Entropy

- Major observation from the case studies
 - Personalization only works well on ambiguous queries such as “GMC”
- How to recognize ambiguous queries?
 - Idea: Ambiguous query \Rightarrow different intents \Rightarrow click on different search results
 - Click entropy: indicate click diversity of a query

$$H(q) = \sum_{p \in Clicks(q)} -P(p | q) \log_2 P(p | q) , \text{ where } P(p | q) = \frac{|Clicks(q, p, \bullet)|}{|Click(q, \bullet, \bullet)|}$$

Building a Predictive Model

- Major features to identify ambiguous queries
 - Click entropy
 - Click-based potential for personalization

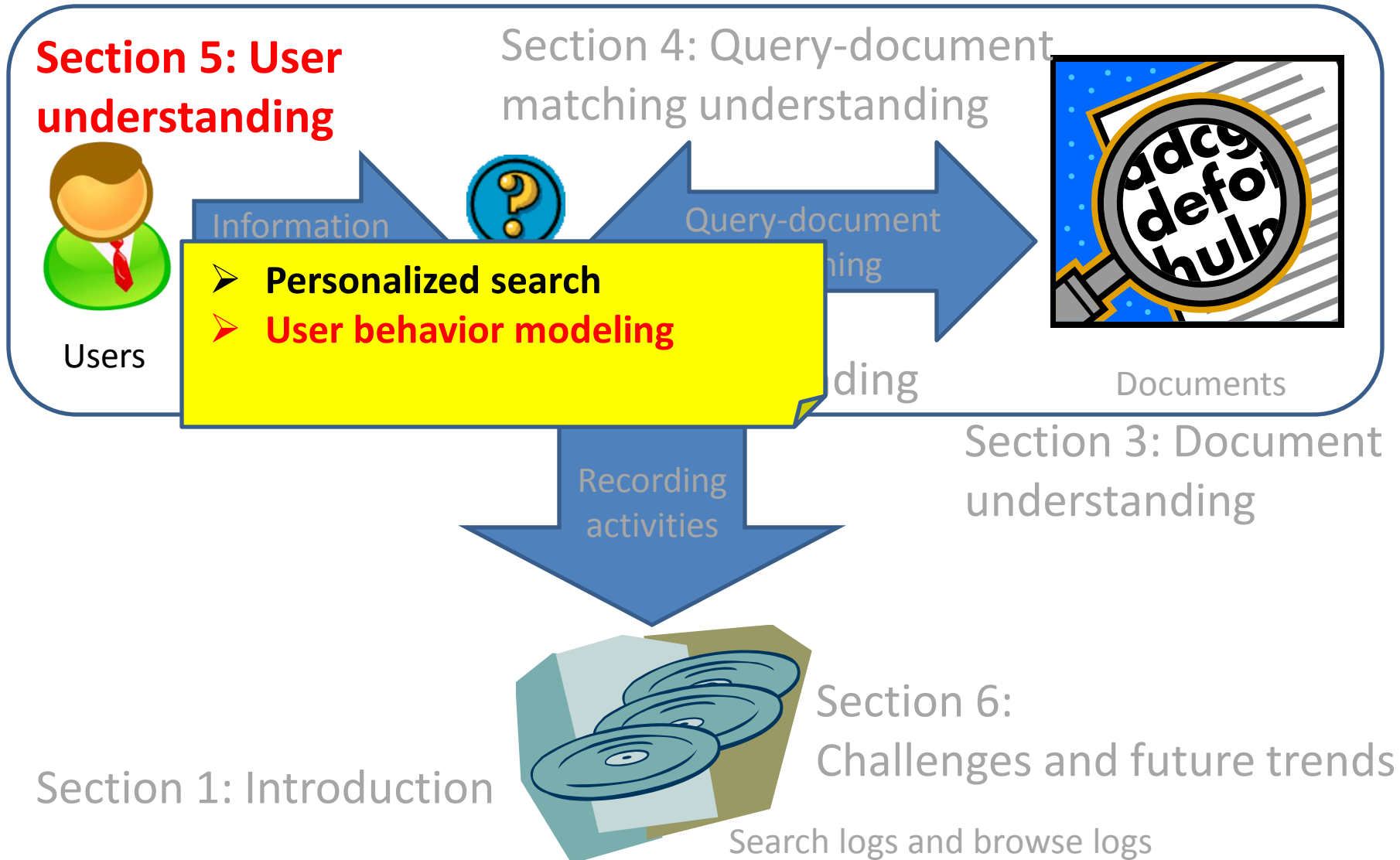


Given the click-through information of several users

- Specify a random user's clicks as the "ground truth"
- Calculate the average NDCG of other users' clicks

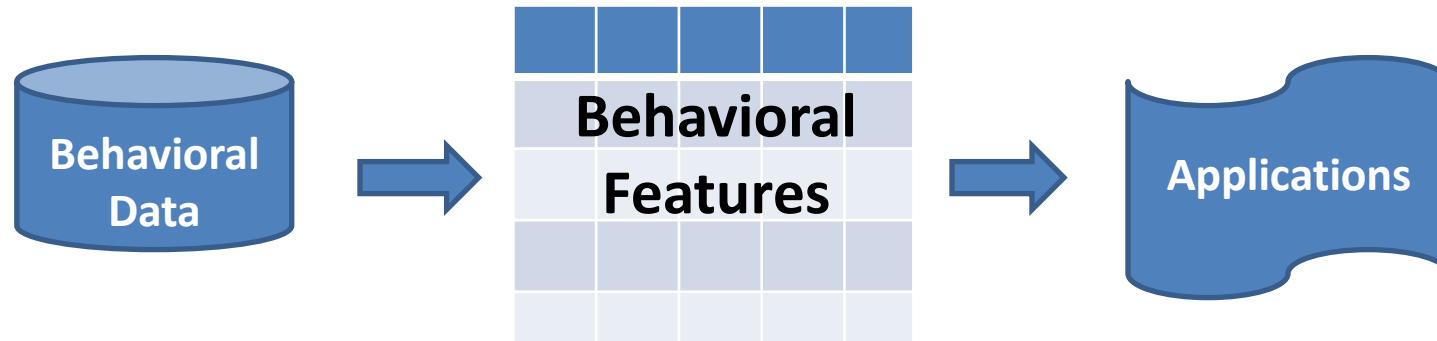
The curve for an unambiguous query like "microsoft earth" is flat, but dips with group size for a more ambiguous query like "street maps".

A Road Map

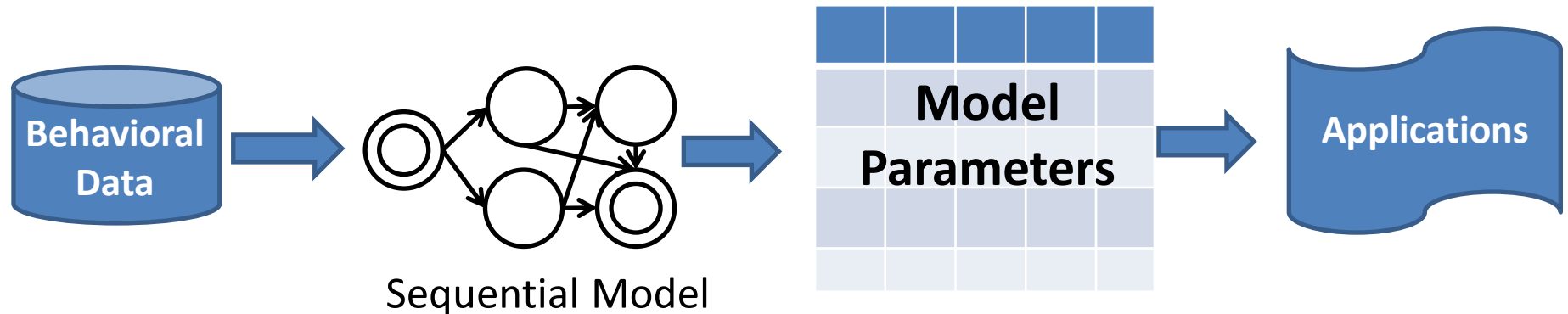


Two Approaches to User Behavior Modeling

- Deriving behavioral features



- Designing sequential models



Deriving Behavioral Features

- Basic pageview-level user behavior
 - Viewing search results
 - Clicking on search results

} Introduced in previous part of the tutorial
- More pageview-level user behavior
 - Dwell-time on pages, mouse movement, screen scrolling, printing, adding to favorite
- Session-level user behavior
 - Post-search browsing behavior
 - Behavior in query chains and search trails

Features Correlated to User Satisfaction

- [Fox05] Apply a user study to correlate behavioral features with user satisfaction
 - Besides click-through, **dwell time** and **exit type** of a search result page strong predictors of user satisfaction
 - Exit type: kill browser window; new query; navigate using history, favorites, or URL entry; or time out.
 - **Printing** and **Adding to Favorites** highly predictive of user satisfaction
 - Combining various behavior predicts user satisfaction better than click-through alone

Features Correlated to Page Relevance

- [Agichtein06] and [Agichtein06a]
- DwellTime
 - Result page dwell time
- DwellTimeDeviation
 - Deviation from expected dwell time for query

Agichtein, E, et al. Learning user interaction models for predicting web search result preferences. SIGIR'06

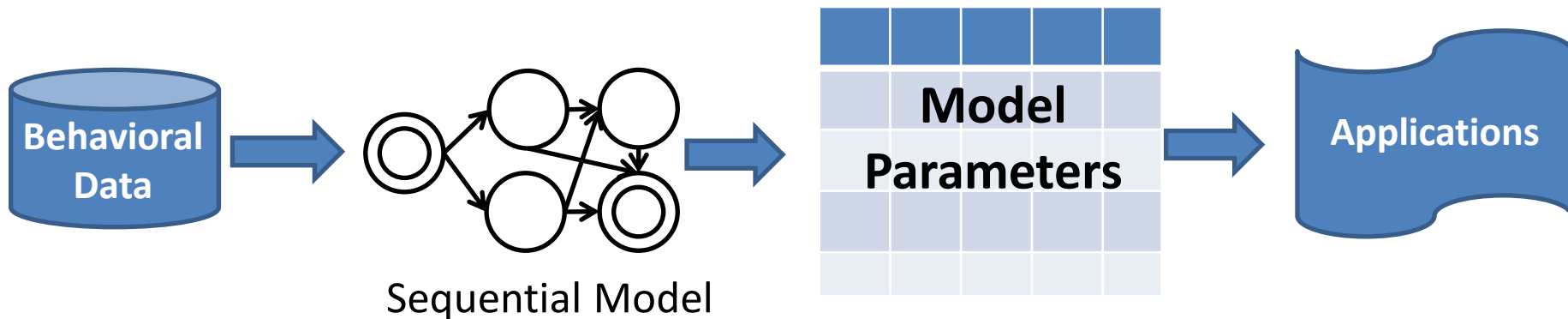
Agichtein, E. et al. Improving Web Search Ranking by Incorporating User Behavior Information. SIGIR'06.

More Features in Previous Studies

- Post-search browsing behavior for URL recommendation ([White07] and [Bilenko08])
- Behavior in sessions to categorize users as navigators and explorers ([White07a])
- Six categories of features for search results pre-fetching ([Downey07])

Two Approaches to User Behavior Modeling

- Deriving behavioral features
- **Designing sequential models**
 - Sequential patterns
 - Markov chains
 - Layered Bayesian model



Sequential Patterns for User Satisfaction

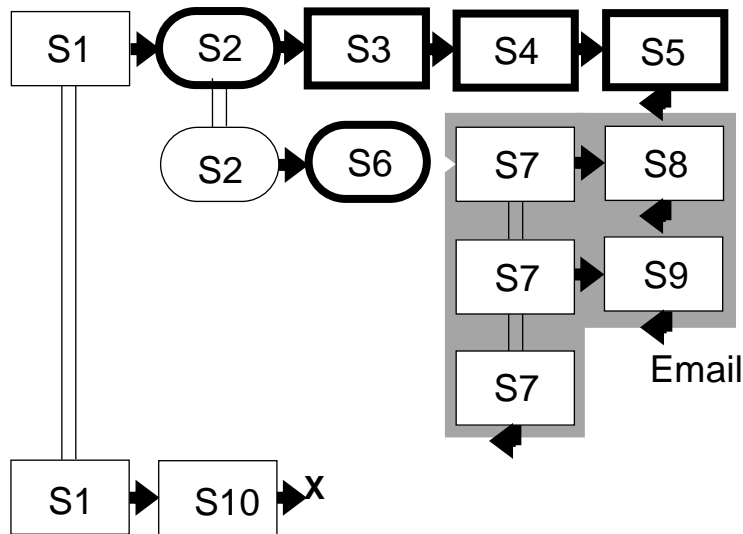
- Describe sessions with a vocabulary of five letters [Fox05]
- Explore correlations between sequential patterns and user satisfaction

Pattern	Freq.	%SAT	%PSAT	%DSAT
SqLrZ	509	81	10	7
SqLrLZ	117	75	15	9
SqLrLrZ	82	73	13	13
SqLrqLr*	70	64	25	10
SqLrLrLrZ	61	57	22	19
SqLrLr*	362	23	39	36
SqLrLrLr*	129	20	37	42
SqLrLrLrLr*	114	13	35	51

- Session starts (S)
- Submit a query (q)
- Result list returned (L)
- Click a result (r)
- Exit on result (Z)

Sequential Patterns for User Types

- Described user behavior with a vocabulary of three letters [White07a]
- Explored correlations between behavioral sequences and types of users (navigators vs. explorers)



S = search
B = browse
b = back

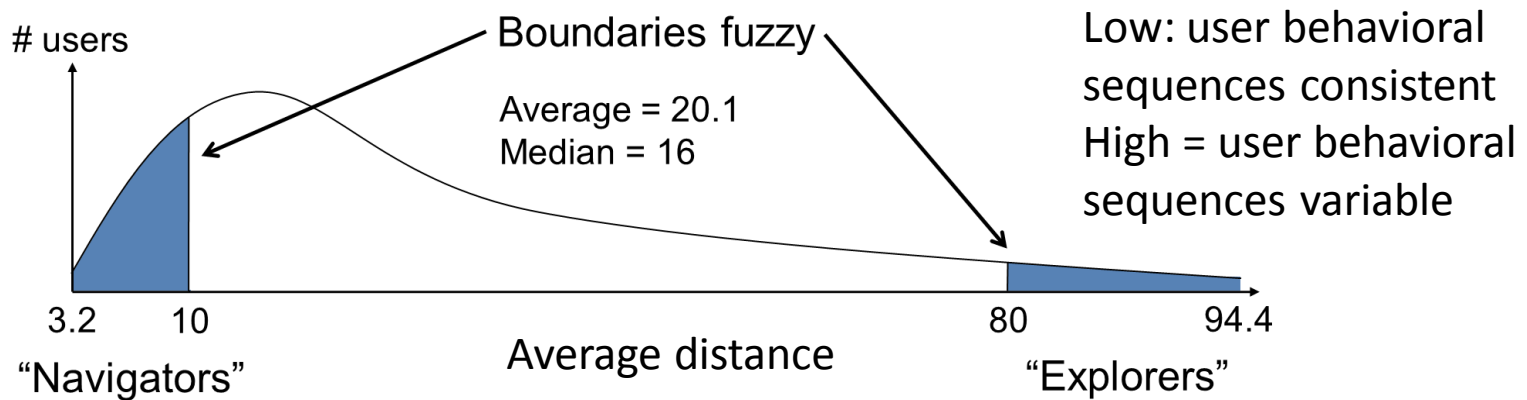
S B B B b S S

An example from a user session to a sequence

Distinguishing Navigators and Explorers

- Calculate the average distance of the behavioral sequences for each user
- Suppose a user has three sessions

	Behavioral Sequences	Pair-wise distance	Average Distance
S1	S S B b S B S	ED(1,2) = 4	$(4+4+5)/3 = 4.33$
S2	S B B b B S b S	ED(1,3) = 4	
S3	S B B B B	ED(2,3) = 5	



Limitations of Sequential Patterns

- The previous sequential pattern models have several limitations
 - Only two or three types of behavior modeled
 - May not capture user behavior well
 - No aggregate on patterns
 - May harm the generalization power on new sessions
 - Hard to incorporate other features for user activities
 - E.g., dwell time, whether first click in the session, etc

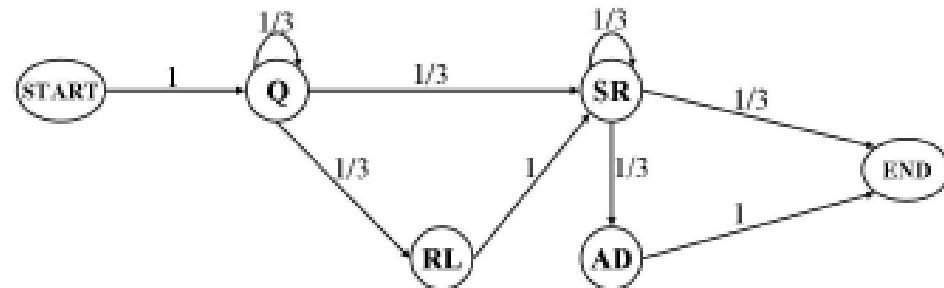
A Markov Chain Model

- Model rich types of user clicks with a Markov chain [Hassan10]
 - START: the user starts a new goal
 - A query (Q)
 - A click of any of the following types:
 - Algorithmic Search Click (SR)
 - Sponsored Search Click (AD)
 - Related Search Click (RL)
 - Spelling Suggestion Click (SP)
 - Shortcut Click (SC)
 - Any Other Click (OTH), such as a click on one of the tabs
 - END: the user ends the search goal

User Activity Markov Model

- The Markov model is defined as $G = (V, E, w)$
 - $V = \{Q, SR, AD, RL, SP, SC, OTH\}$ is the set of possible user actions during the session
 - $E \subseteq V \times V$ is the set of possible transitions between any two actions
 - $w: E \rightarrow [0..1]$ is the transition probability from state s_i to state s_j

$$w(s_i, s_j) = \frac{N_{s_i, s_j}}{N_{s_i}}$$



Goal 1: Q 4s RL 1s SR 53s SR 118s END
Goal 2: Q 3s Q 5s SR 10s AD 44s END

Predict Search Success

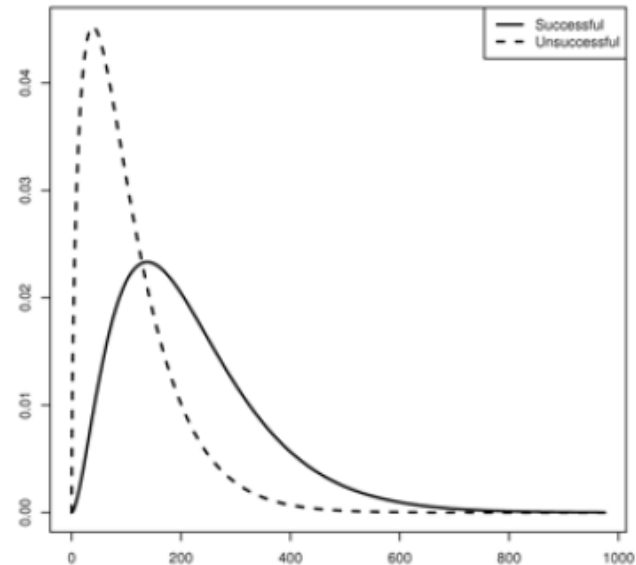
- Offline: train two Markov models
 - M_s model: trained by success sessions
 - M_f model: trained by failure sessions
- Online: given a session $S=(s_1, \dots, s_n)$
 - Calculate the likelihood with M_s and M_f , respectively

$$LL_M(S) = \sum_{i=2}^n W(S_{i-1}, S_i)$$

$$Pred(S) = \begin{cases} 1 & \text{if } \frac{LL_{M_s}(S)}{LL_{M_f}(S)} > \tau \\ 0 & \text{otherwise.} \end{cases}$$

Adding Time to Model

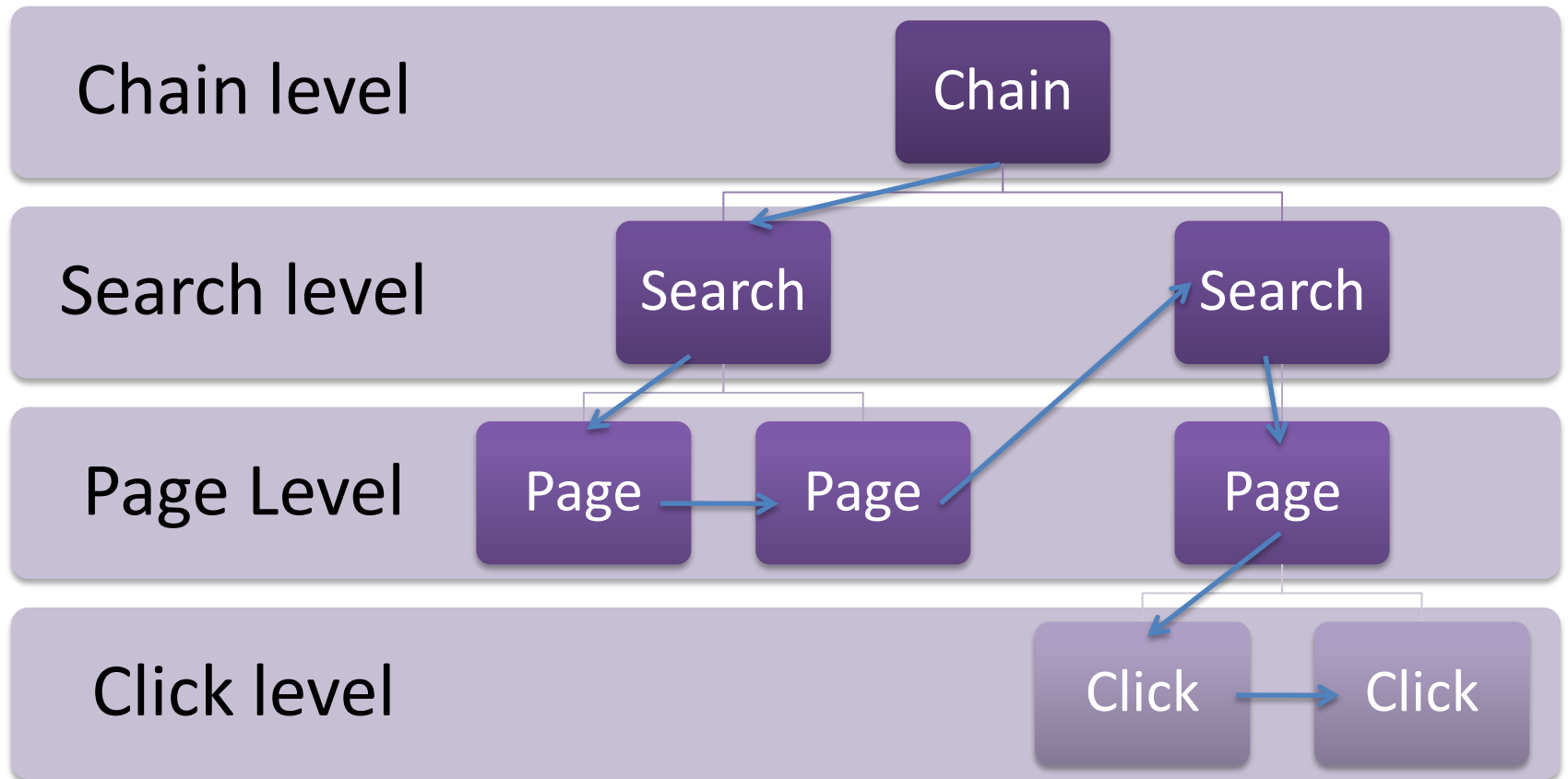
- Time distributions of state transitions are different in successful and unsuccessful sessions
- Assume the transition time follows gamma distribution
- Time distributions incorporated into the transition probabilities of the Markov models



Hassan, A. et al. Beyond DCG: user behavior as a predictor of a successful search, WSDM'10.

Time distributions of SR \rightarrow Q transitions for successful and unsuccessful sessions

From Flat Model to Hierarchical Model

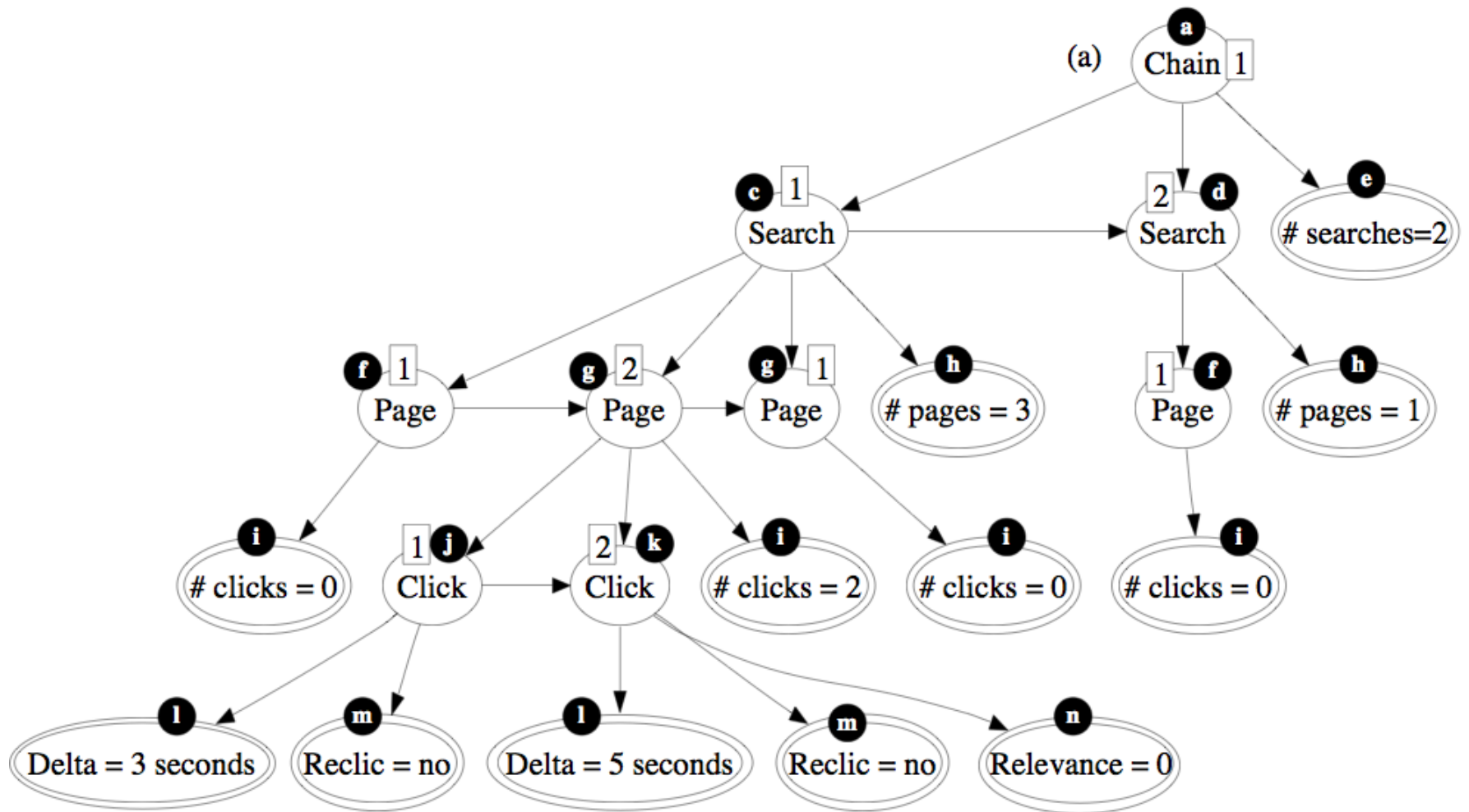


Piwowarski, B., et al. Mining user web search activity with layered Bayesian networks or how to capture a click in its context. WSDM'09.

A Bayesian Network Model

- Four hidden variables
 - Chain, search, page, click
 - Each hidden variable has a predefined number of states
- Each hidden variable is associated with some observed features
 - Chain: # searches
 - Search: # pages requested
 - Page: # clicks
 - Click: a) dwell time; b) whether “re-click”; c) relevance judgement if available

Example of BN



Piwowarski, B., et al. Mining user web search activity with layered Bayesian networks or how to capture a click in its context. WSDM'09.

Using BN to Predict Page Relevance

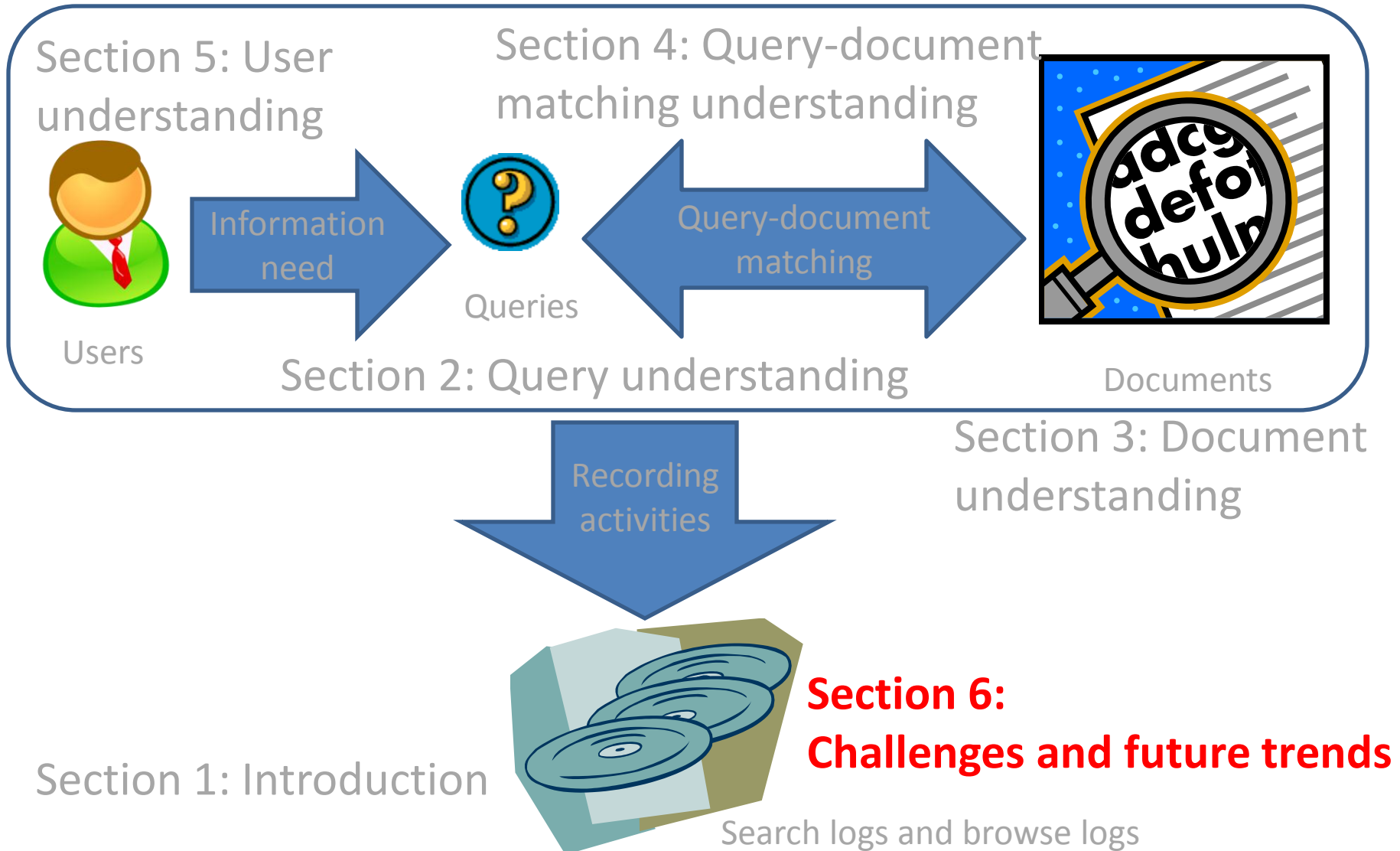
- In the offline stage
 - Learning model parameters
- In the online stage
 - Given the observed values of a user session, infer the distributions of states of the hidden variables
 - Extract BN features
 - Distribution of the states for each hidden variable
 - Maximal likelihood of the BN
- Combining BN features with other features
 - E.g., Position of search result, etc.

Piwowarski, B., et al. Mining user web search activity with layered Bayesian networks or how to capture a click in its context. WSDM'09.

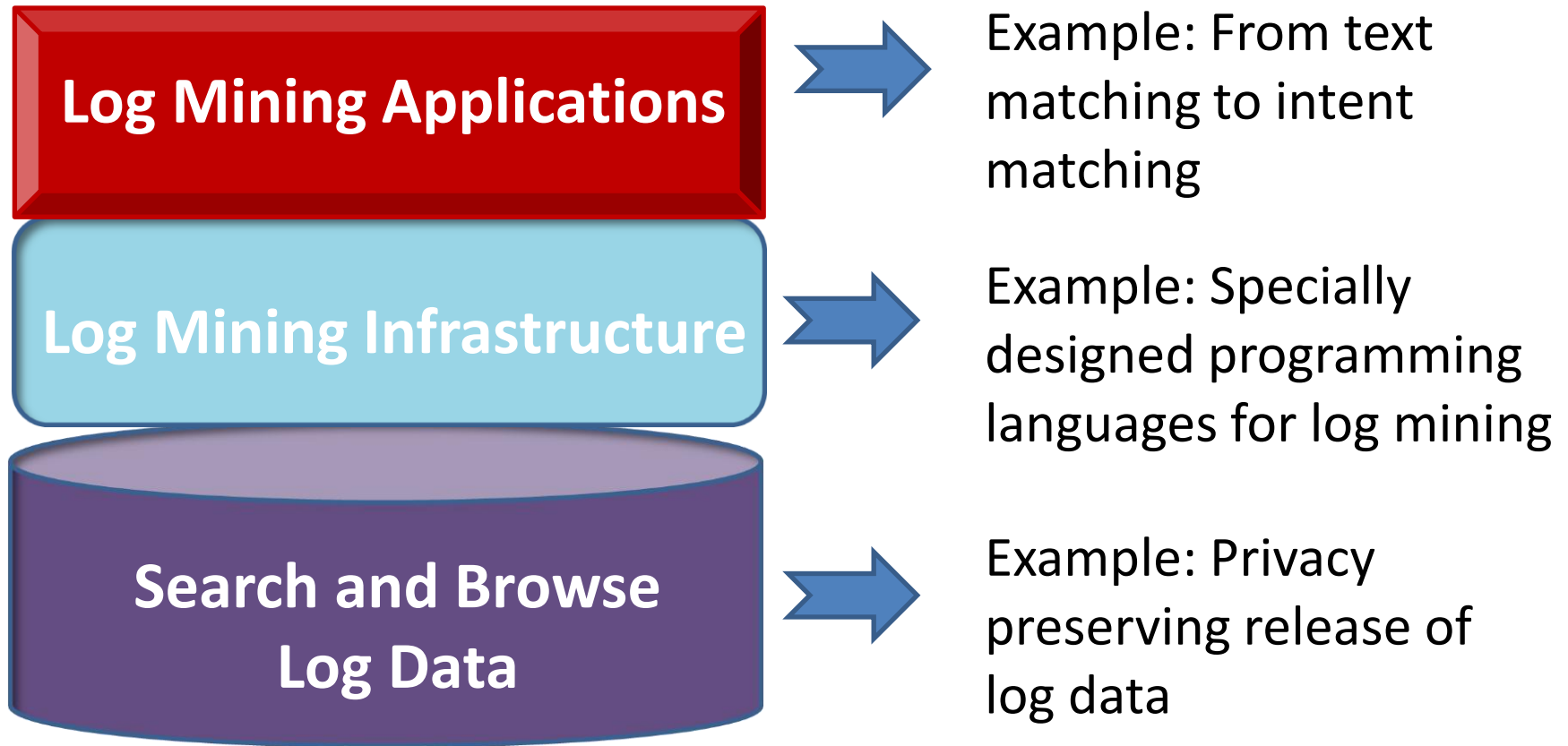
Summary of User Understanding

- What users searched or browsed in the past
 - Personalization: better understand users' intent
- How users searched or browsed in the past
 - User behavior modeling
 - Deriving behavioral features
 - Designing sequential models

A Road Map



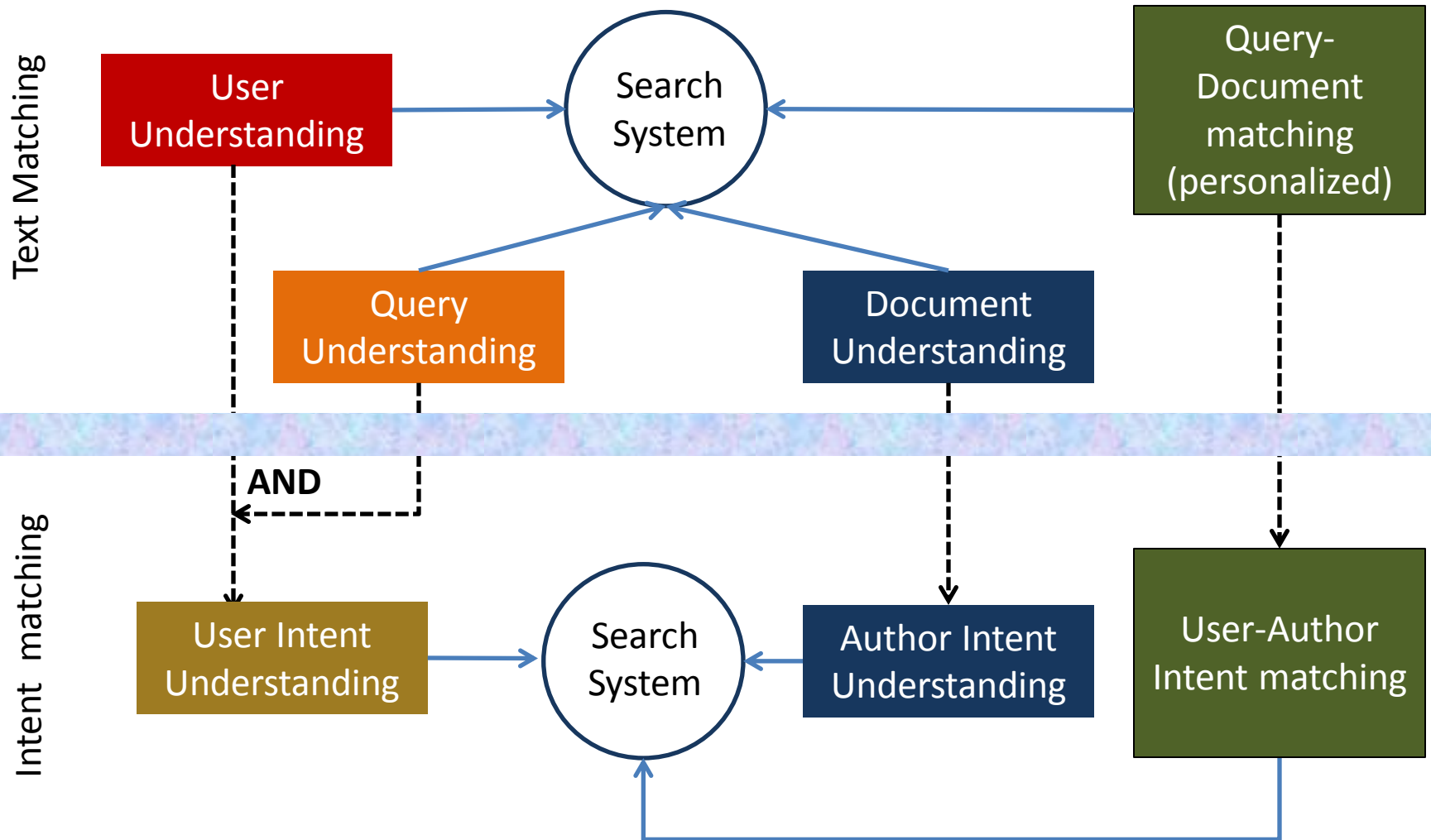
Challenges and Trends



Why Intent Matching

- Traditional IR is based on text matching between queries and documents
- In Web search, queries are often short and ambiguous
- Accurately interpreting user intent from non-perfect user queries is key to search engines
- Trend: from text matching to intent matching

From Text Matching to Intent Matching

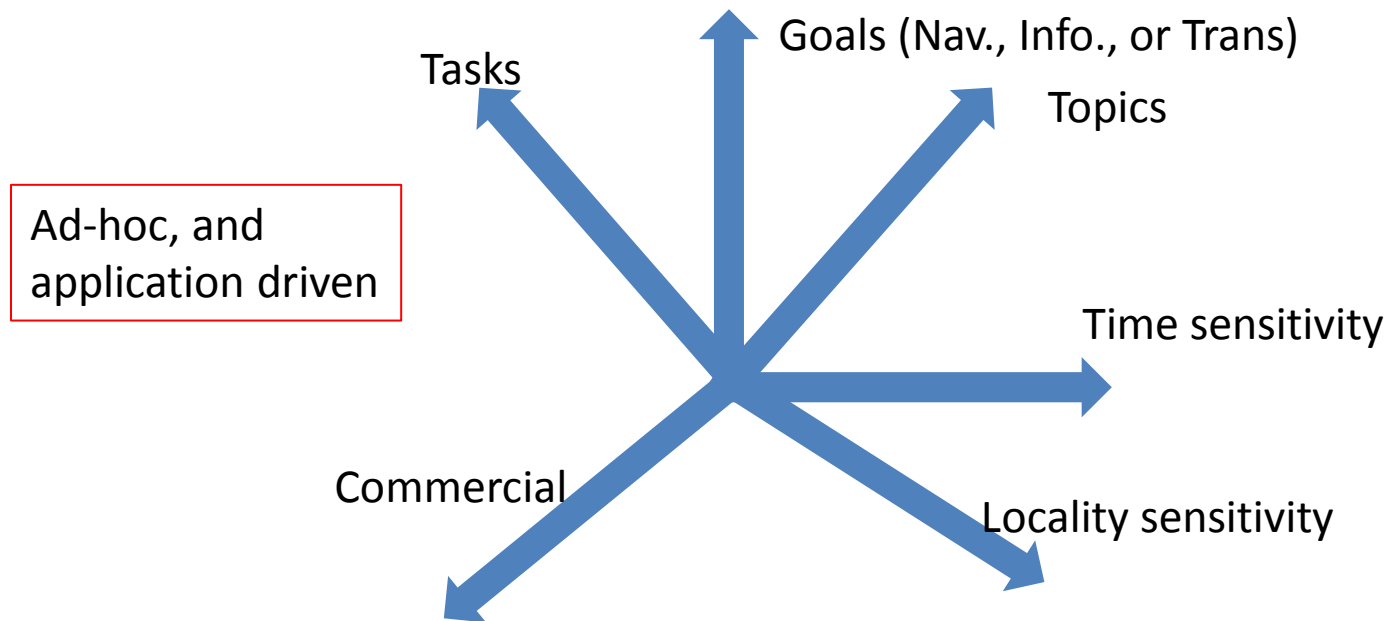


Challenges for Intent Matching

- How to represent intent?
- How to capture users' intent?
- How to infer and index authors' intent?
- How to do intent matching?

Previous Work on Intent Representation

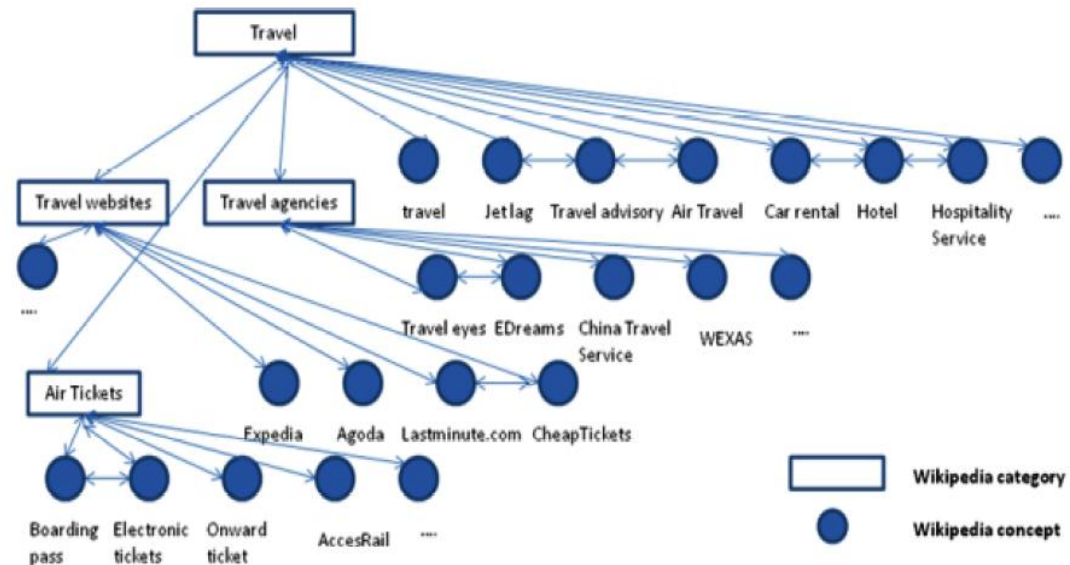
- By multiple dimensions



- By existing knowledge base
- By queries and clicked documents from log data

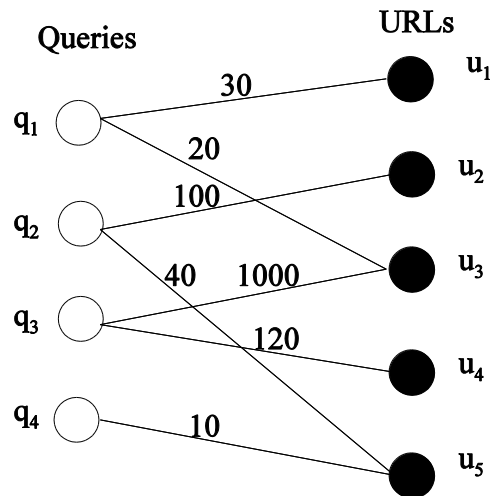
Representing Intent by Knowledge Base

- An intent is represented by a collection of Wikipedia concepts and categories [Hu09]
- Accurate and systematic, but may not adapt well to new intents emerging from the users and the Web documents



Mining Intents from Log Data

- Mining click-through bipartite
 - Each intent is represented by a collection of similar queries as well as the clicked URLs [Cao09]
- Automatically adapt to emerging user interests, but results not as good as human edited knowledge base



Cluster similar queries from the click-through bipartite

Example: queries “MSRA”, “Microsoft Research Asia”, “MSR Beijing” all lead to clicks on <http://research.microsoft.com/en-us/labs/asia/>

Future Work on Intent Representation

- Towards a systematic framework to represent intents
 - Including the multiple dimensions mentioned before
 - Hierarchical granularity
 - Example: domain → topic → concept
 - Automatically adapting to emerging intents
- Joint efforts of log mining, Web page mining, semantic Web, and natural language processing

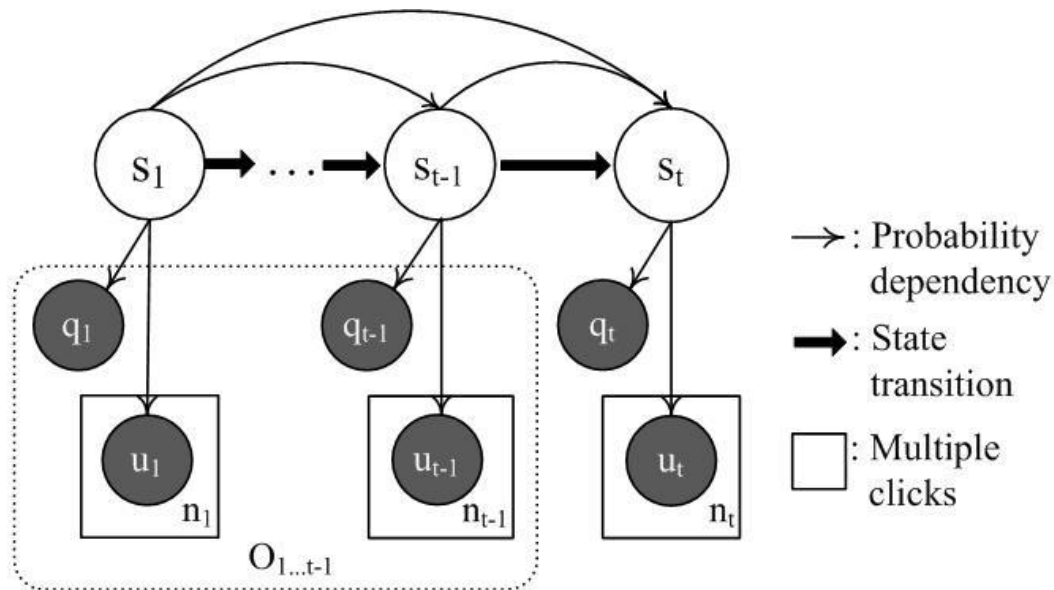
Challenges for Intent Matching

- How to represent intent?
- How to capture users' intent?
- How to infer and index authors' intent?
- How to do intent matching?

How to Capture User Intent

- Many previous studies built various query classifiers for different dimensions of intent
 - Most of them targeted at individual queries
- Context-aware intent understanding
 - Example: “GMC” refers to General Motor cars or General Medical Council
 - Context: the user searched “nissan” and “toyota” immediately before “GMC”
 - Intent: the user is likely to search for GM cars; the user may want to compare different cars

A Hidden-Markov Model Approach



- Behind each search, a user bears a search intent in mind (hidden state)
- The user formulates queries and clicks search results (observations)
- Given a query q_t and its context info $O_{1..t-1}$, the HMM model
 - **Infers** user's search intent at **time t** (for re-ranking)
 - **Predicts** user's next search intent at **time $t+1$** (for query and URL recommendation)

Cao, H., et al. Towards context-aware search by learning a very large variable length hidden markov model from search logs. WWW'09.

Future Work on User Intent Understanding

- Context plays an important role to infer the users' intent
 - Immediately previous queries and clicks
 - Web pages browsed shortly before
 - Time when the query is issued
 - Location of the user
 - Search device used by the user

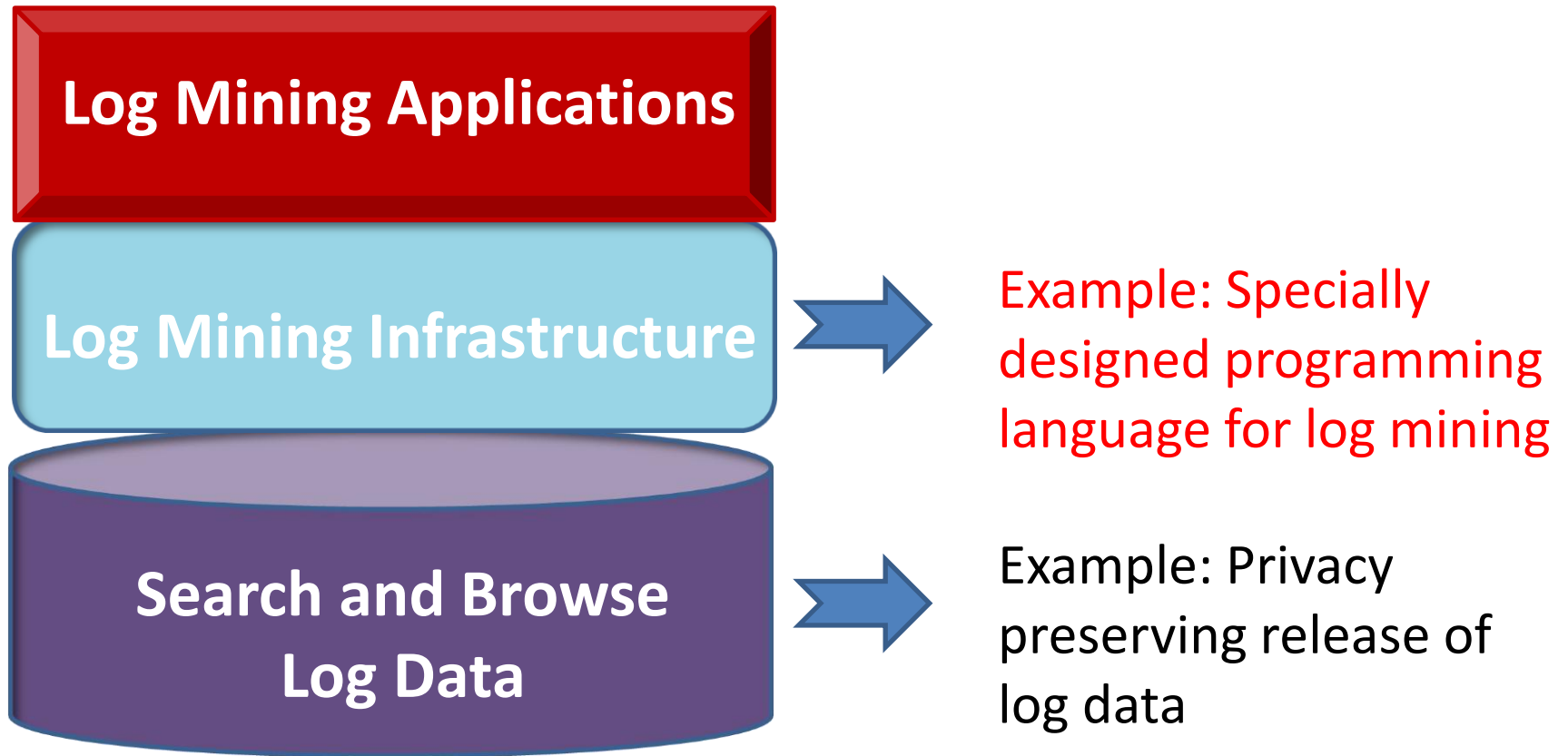
Challenges for Intent Matching

- How to represent intent?
- How to capture users' intent?
- How to infer and index authors' intent?
- How to do intent matching?

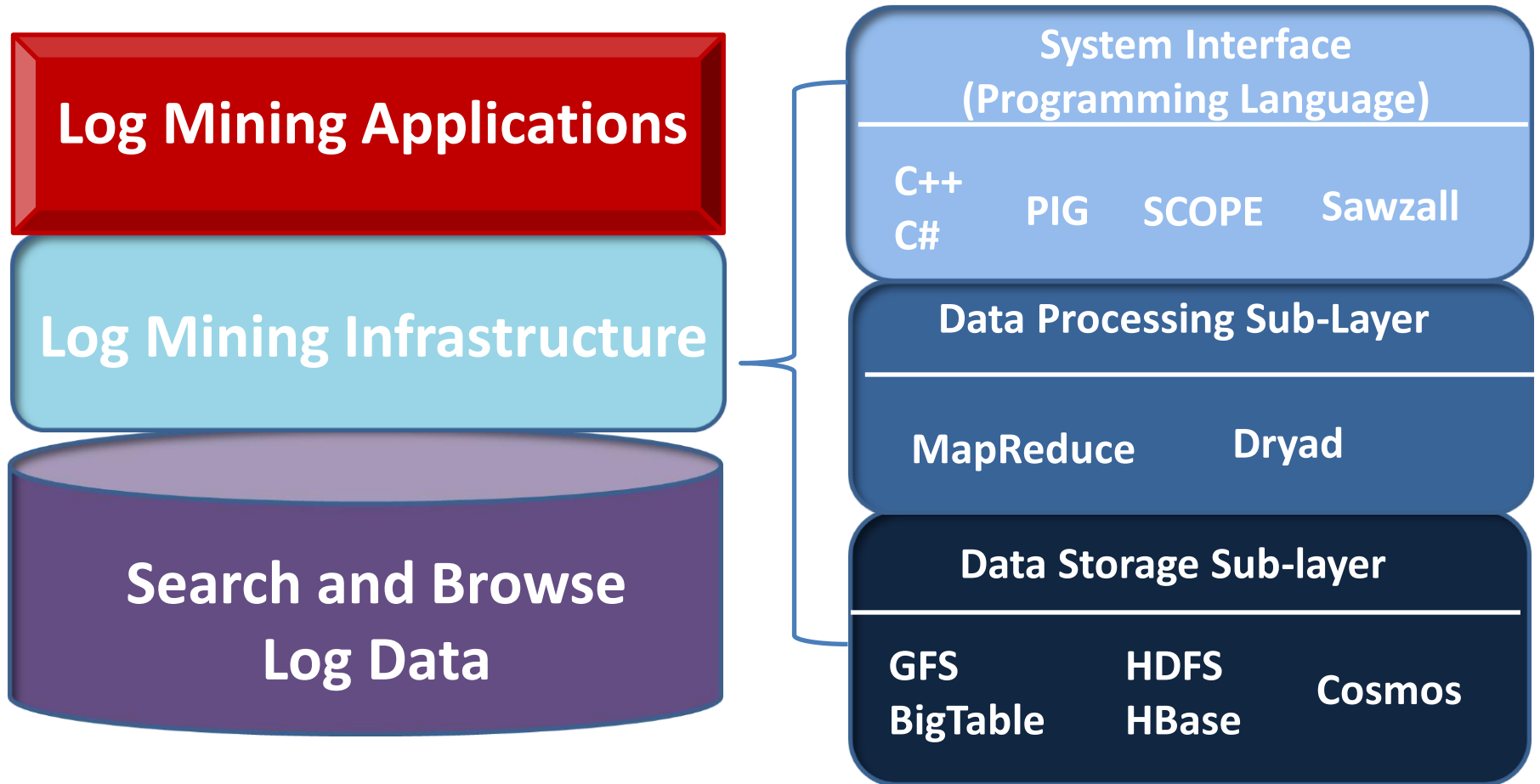
Few studies on these two topics, left for future work



Challenges and Trends

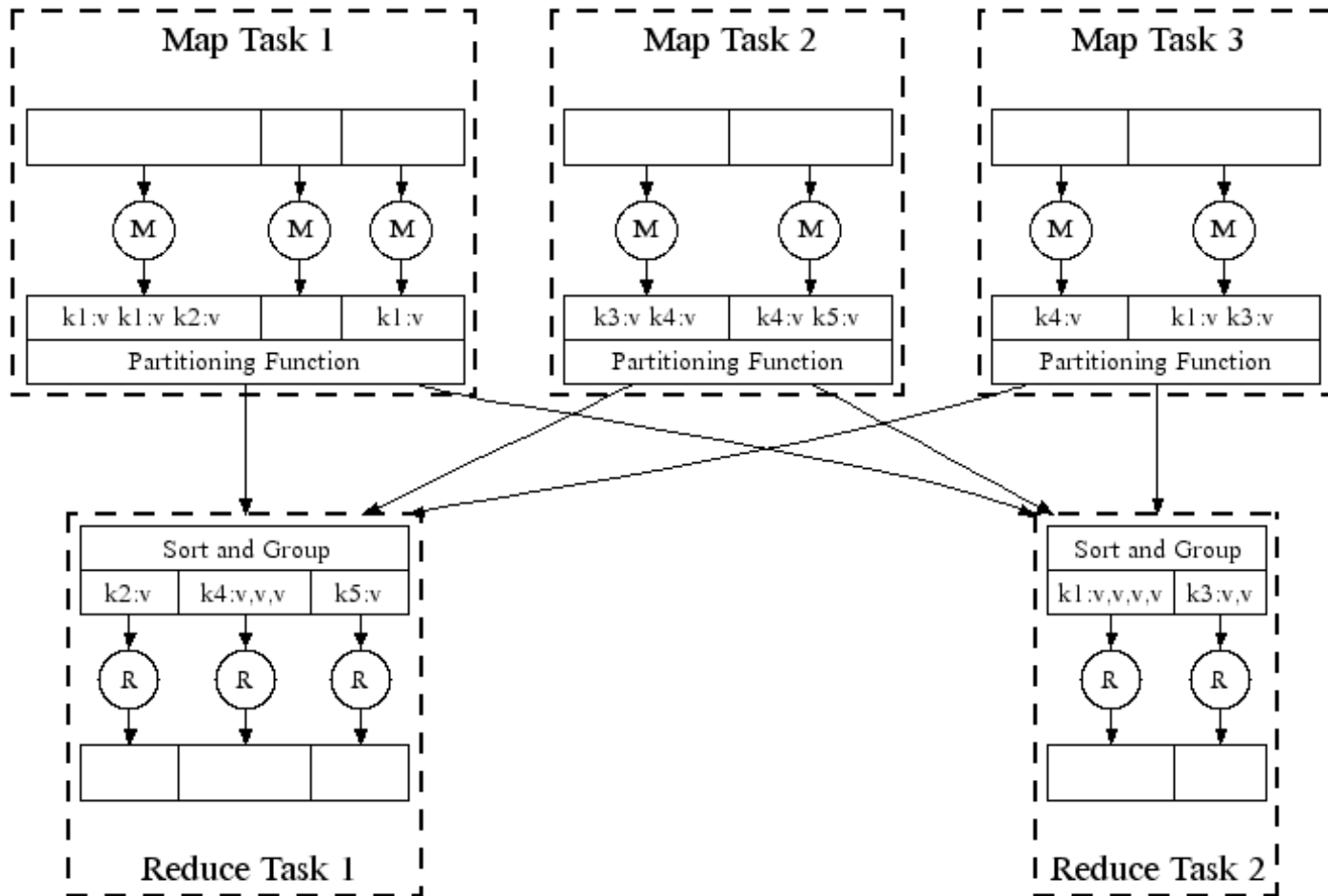


Log Mining Infrastructure



Log data may reach the magnitude of tera-types per day

Map Reduce



Example: Count Word Occurrences

```
Map(String input_record) {  
    // input_record: a line of logs  
    for each word w in input_record {  
        EmitIntermediate(w, "1");  
    }  
}
```

```
Reduce(String output_key, Iterator intermediate_values) {  
    // output_key: a word  
    // output_values: a list of counts  
    int result = 0;  
    for each v in intermediate_values {  
        result += ParseInt(v);  
    }  
    Output(result);  
}
```

Challenges for Map-Reduce Model

- Some applications are hard to be expressed in the map-reduce model
 - E.g., multiplication of large-scale matrices
- Difficult for system to automatically optimize execution plans
 - Some complex applications may involve multiple steps of map and reduce
 - Implementation in C++ or C#

The SCOPE Language

- An example: find the queries which have been requested for at least 1,000 times

```
SELECT query, COUNT(*) AS count
FROM "search.log" USING LogExtractor
GROUP BY query
HAVING count > 1000
ORDER BY count DESC;

OUTPUT TO "qcount.result";
```

- Similar to SQL
- No need to decompose a job into map and reduce
- Optimization rules borrowed from database community

Challenge: Global Optimization

- The log mining system may receive hundreds of jobs for each day
 - Many jobs may consume the same data and share similar computation steps
 - The optimization rules borrowed from database system target at optimizing a single job



Common data access
Common computation

Global optimization needed

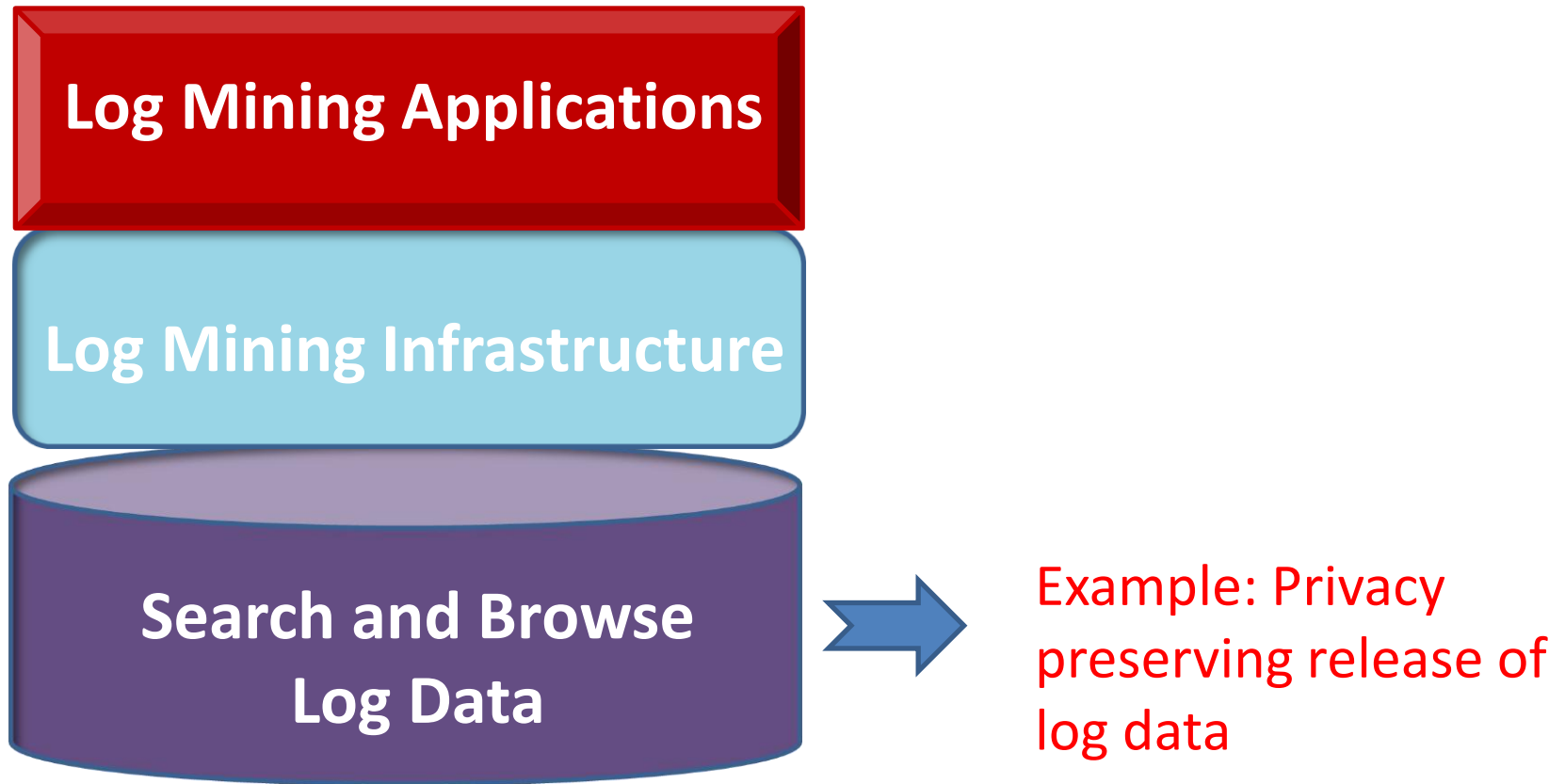
Challenge: Comparison of Languages

- There have been various programming languages, e.g., SCOPE, PIG¹, Sawzall²
- The computation power of those languages is not clear
- There has been no comparison of those languages

1. Apache. Pig. <http://incubator.apache.org/pig/>, 2008.

2. Pike, R. et al. Interpreting the data: Parallel analysis with Sawzall. Scientific Programming , 2005.

Challenges and Trends



The AOL Data Release

- AOL data release, 2006
 - 650K users, 20 million Web search queries
 - Users anonymized by hash functions, IP omitted
 - User No. 4417749 was identified through her query history by a newspaper journalist
 - CTO resigned, 2 employees fired
 - Class action law suit pending
 - CNN Money: “101 dumbest moments in business”

Michael Arrington (2006-08-06). "AOL proudly releases massive amounts of user search data". TechCrunch.

Barbaro, M. and Zeller, T. 2006. A face is exposed for AOL searcher no. 4417749. In The New York Times.

Horowitz, A. et al. 101 dumbest moments in business, the year's biggest boors, buffoons, and blunderers. In CNN Money, 2007.

Private Information in Log Data

- Direct identity information
 - Social security number, credit card number, driver's license number, address, phone number, email address, etc
- Indirect identity information
 - DOB, zip code, gender, age, etc
 - May identify a person when joined with other data sources
- Potentially sensitive subjects
 - Health condition, financial condition, political affiliation, religious affiliation, etc
 - Dependency on identity information

Privacy-Preserving Log Release

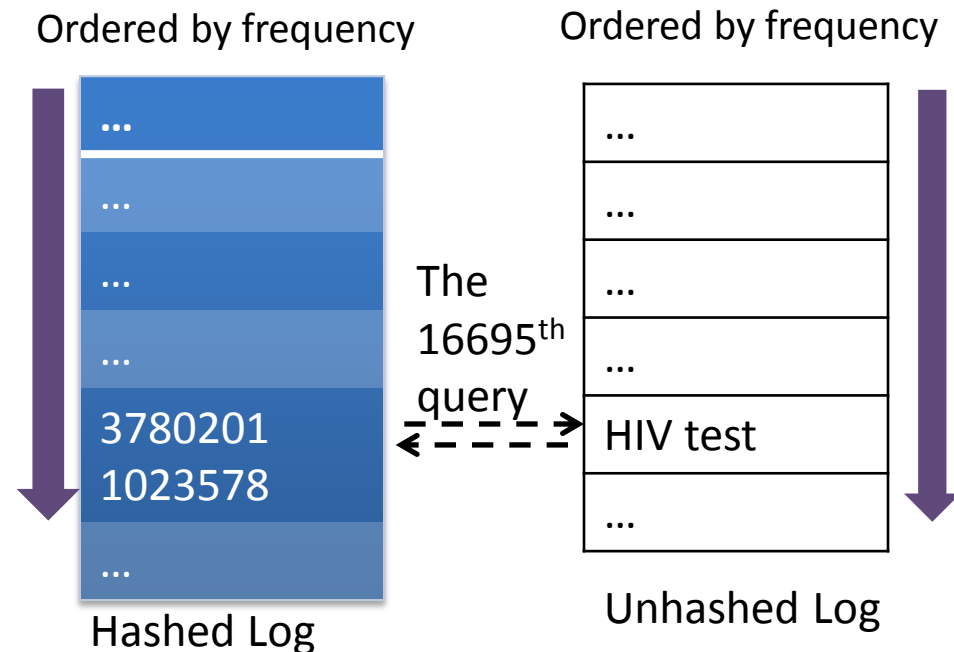
- Typical steps in breaching privacy
 - Step 1: link all query terms related to an individual
 - Step 2: find private information from the linked queries
- Approaches to privacy-preserving log releasing
 - Breaking the linkage of queries
 - Scrubbing private information from queries

Breaking Linkage of Queries

- Deleting identifier
 - Remove all user identifiers (e.g., IP, cookie ID) of each log record
 - Note that hashing identifier (as in AOL release) does not break the linkage of queries
- Aggregating queries [Korolova09]
 - Only aggregate information is released
 - E.g., count of queries, total number of clicks, etc.
 - Noise may be added to the aggregate information

Scrubbing Private Information (1)

- Token-based hashing
 - Hash each token in queries
 - Serious privacy leaks are possible when joined with unhashed log data
 - Major idea: mapping between the hash codes and query words
 - Reverse engineer some queries using frequency statistics
 - Reveal more terms using their co-occurrence relationship in queries



Scrubbing Private Information (2)

- Removing infrequent queries
 - Frequent queries are unlikely to contain personal information
- Removing long digit strings
 - Possibly SSN, phone number, credit card number, etc
- Removing email addresses

Adar, E. User 4xxxxx9: anonymizing query logs. Workshop at WWW'07.

Xiong L. and Agichtein, E. Towards privacy-preserving query log publishing.

Workshop at WWW'07.

Future Directions (1)

- Metrics for privacy and utility
 - Any privacy-preserving technique is a tradeoff between privacy and utility
 - Example: deleting user IDs => losing sessions
 - Each previous work targeted at particular applications and had its own definitions of privacy and utility
 - Need explicit and general metrics for privacy and utility

Xiong L. and Agichtein, E. Towards privacy-preserving query log publishing. Workshop at WWW'07.

Future Directions (2)

- Approaches from other communities
 - Database community
 - Network community

Adar, E. User 4xxxxx9: anonymizing query logs. Workshop at WWW'07.

Xiong L. and Agichtein, E. Towards privacy-preserving query log publishing. Workshop at WWW'07.

Future Directions (3)

- Technical solutions plus policy-based protections
 - Privacy laws
 - Privacy policies
 - Confidentiality and licensing agreements
 - Institutional review boards

Summary

Section 5: User understanding



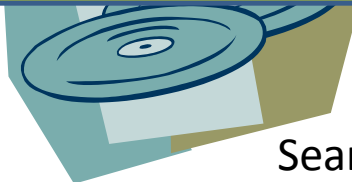
Users

Section 4: Query-document matching understanding



- Search & browse logs
 - Search logs: collected by search engine servers; store queries, clicks, and search results
 - Browse logs: collected by client-side browser plug-ins or ISP proxy servers; store queries, clicks, and browse information
- Log mining applications
 - Query understanding, document understanding, user understanding, query-document matching,
- Four data structures
 - Query histogram, click-through bipartite, click patterns, session patterns

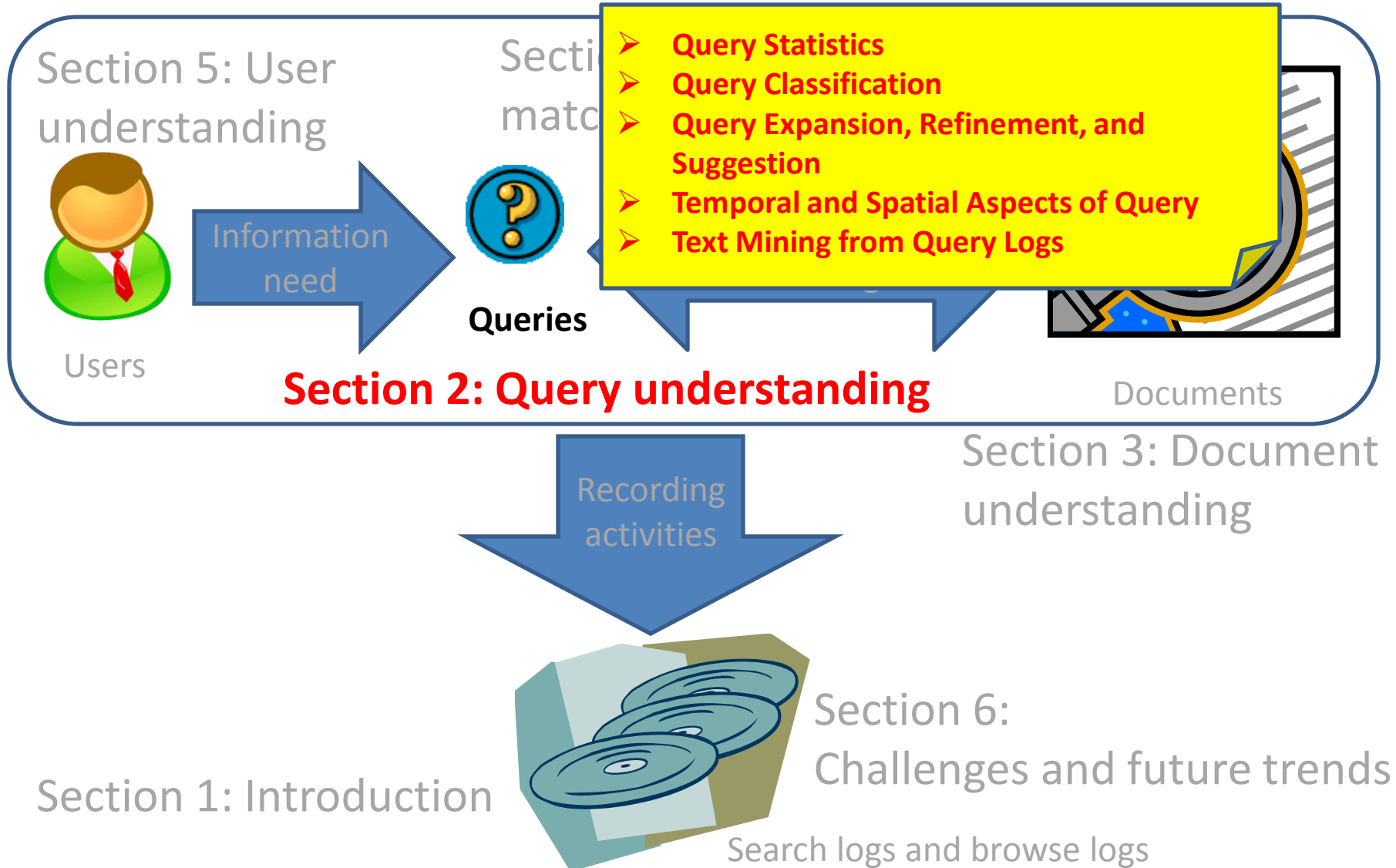
Section 1: Introduction



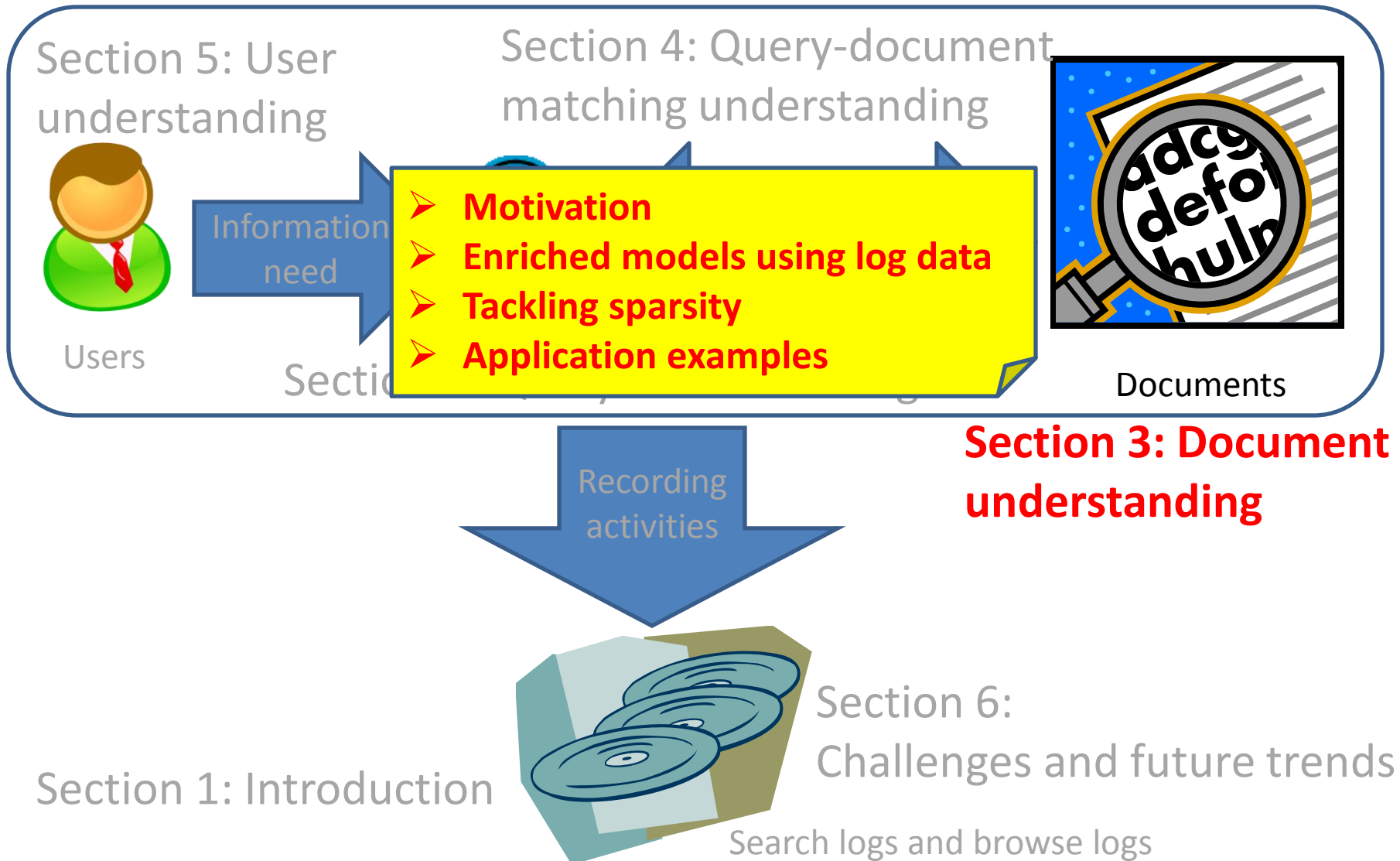
Challenges and future trends

Search logs and browse logs

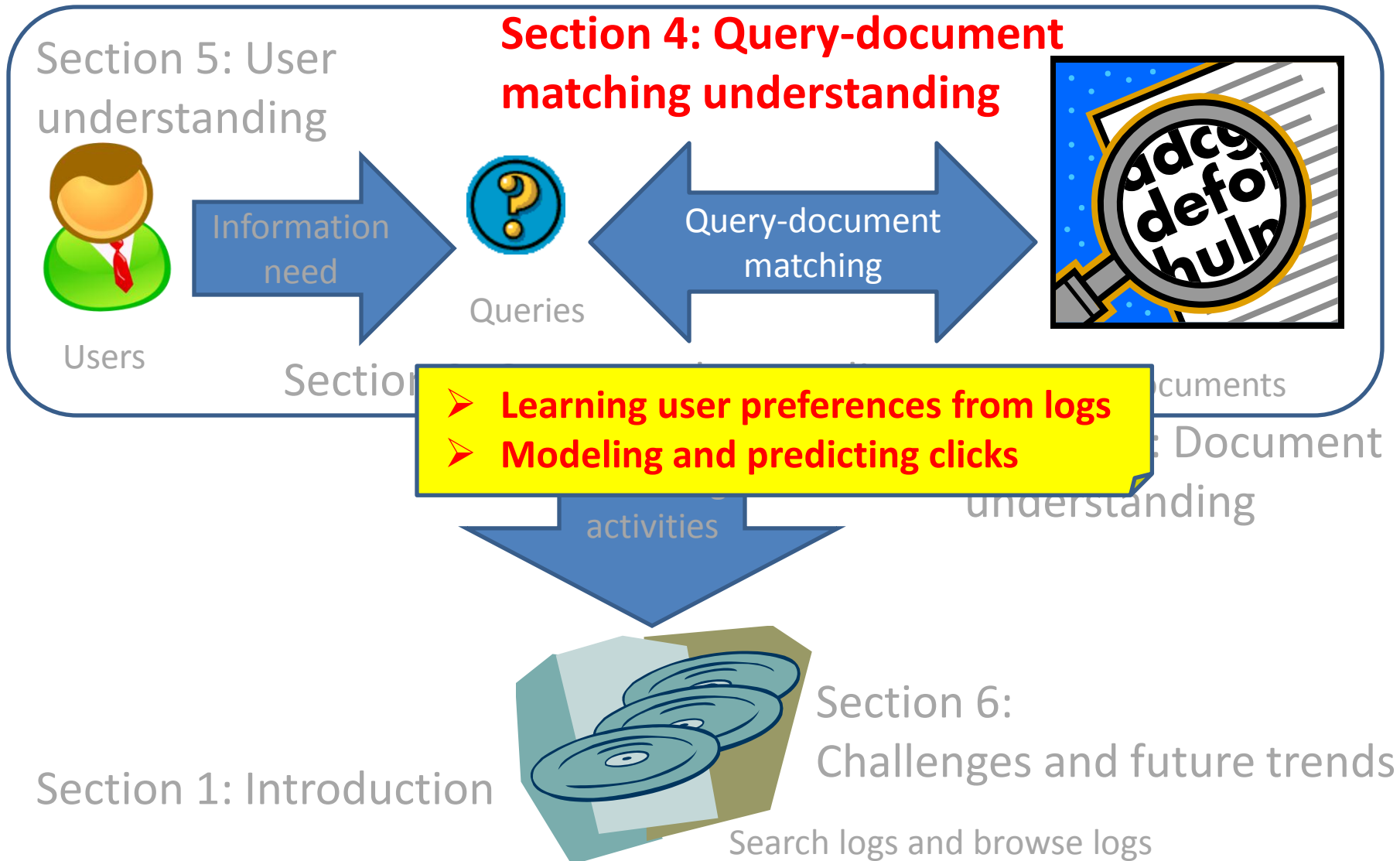
Summary



Summary



Summary



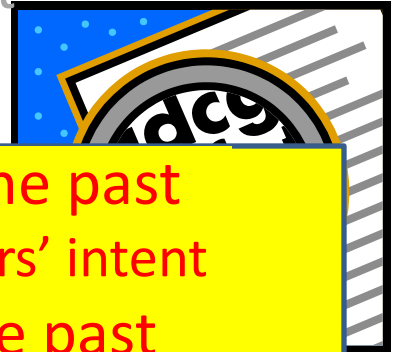
Summary

Section 5: User understanding



Users

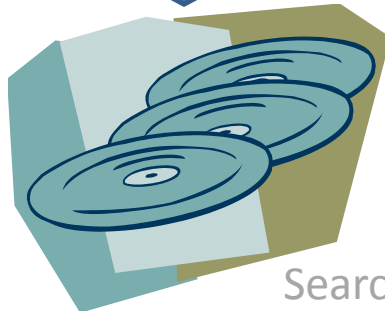
Section 4: Query-document matching understanding



- What users searched or browsed in the past
 - Personalization: better understand users' intent
- How users searched or browsed in the past
 - User behavior modeling
 - Deriving behavioral features
 - Designing sequential models

ent

Section 1: Introduction



Section 6:
Challenges and future trends

Search logs and browse logs

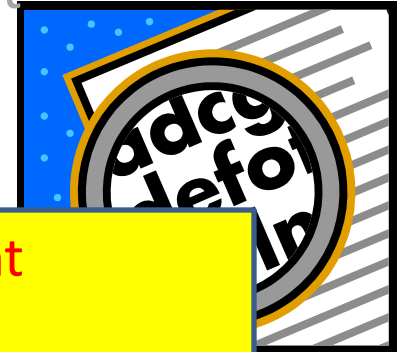
Summary

Section 5: User understanding



User

Section 4: Query-document matching understanding



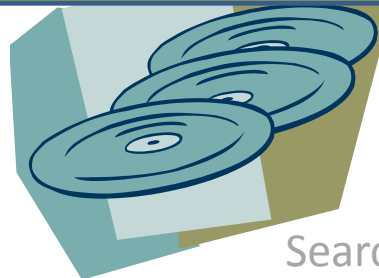
ents

document

g

- Application layer: from text matching to intent matching
- Infrastructure layer: specially designed programming language for log mining
 - Global optimization at the system level
- Data layer: privacy preserving release of log data
 - Combination of technical approaches and policy approaches

Section 1: Introduction



**Section 6:
Challenges and future trends**

Search logs and browse logs

References

- [Adar07] Adar, E. User 4xxxx9: anonymizing query logs. Workshop at WWW'07.
- [Agarwal09] Agarwal, D., et al. Spatio-temporal models for estimating click-through rate. WWW'09
- [Agichtein06] Agichtein, E, et al. Learning user interaction models for predicting web search result preferences. SIGIR'06
- [Agichtein06a] Agichtein, E. et al. Improving Web Search Ranking by Incorporating User Behavior Information. SIGIR'06.
- [Agrawal09] Agrawal, R., et al. Generating labels from clicks. WSDM'09
- [Arrington06] Arrington, M. AOL proudly releases massive amounts of user search data. TechCrunch. 2006
- [Backstrom08] Backstrom, L., et al. Spatial variation in search engine queries. WWW'08.
- [Barbaro06] Barbaro, M. and Zeller, T. A face is exposed for AOL searcher no. 4417749. In The New York Times, 2006.
- [Beeferman00] Beeferman, D. and Berger, A.L. Agglomerative clustering of a search engine query log. KDD'00
- [Beitzel07] Beitzel, S.M. et al. Temporal analysis of a very large topically categorized web query log, J. Am. Soc. Inf. Sci. Technol., vol. 58, no. 2, pp. 166–178, 2007
- [Beitzel07a] Beitzel, S.M., et al. Automatic classification of web queries using very large unlabeled query logs. ACM Trans. Inf. Syst., 25(2):9, 2007.
- [Bilenko08] Bilenko, M. and White, R. W. Mining the search trails of surfing crowds: identifying relevant websites from user activity. WWW'08
- [Broder02] Broder, A. A Taxonomy of Web Search. SIGIR'02
- [Cacheda01a] Cacheda, F. and Vina, A. Experiences retrieving information in the World Wide Web. In Proceedings of the 6th IEEE Symposium on Computers and Communications, 2001
- [Cacheda01b] Cacheda, F. and Vina, A. Understanding how people use search engines: a statistical analysis for e-business. In Proceedings of the e-Business and e-Work Conference and Exhibition, 2001
- [Cao09] Cao, H. et al. Towards Context-Aware Search by Learning A Very Large Variable Length Hidden Markov Model from Search Logs. WWW'09.
- [Chaiken08] Chaiken, R., et al. SCOPE: easy and efficient parallel processing of massive data sets. VLDB'08
- [Chapelle09] Chapelle, O. and Zhang, Y. A Dynamic Bayesian Network Click Model for Web Search Ranking. WWW'09

References

- [Chien05] Chien, S. and Immorlica, N. Semantic similarity between search engine queries using temporal correlation. WWW'05
- [Ciaramita08] Ciaramita, M., et al. Online learning from click data for sponsored search. In WWW'08
- [Cid06] Cid, A., et al. Automatic maintenance of web directories using click-through data. ICDEW '06
- [Cooper08] Cooper, A. A survey of query log privacy – enhancing techniques from a policy perspective. ACM Transactions on the Web. 2008.
- [Craswell07] Craswell, N. and Szummer, M. Random walks on the click graph. SIGIR'07
- [Craswell08] Craswell, N., et al. An experimental comparison of click position-bias models. WSDM'08
- [Cui02] Cui, H., et al. Probabilistic query expansion using query logs. WWW'02
- [Dean04] Dean, J. and Ghemawat, S. MapReduce: simplified data processing on large clusters. OSDI'04.
- [Dou07] Dou, Z. et al. A large-scale evaluation and analysis of personalized search strategies. WWW'07.
- [Dou08] Dou, Z., et al. Are click-through data adequate for learning web search rankings? CIKM'08
- [Dupret08] Dupret, G. E. and Piwowarski, B. A user browsing model to predict search engine click data from past observations. SIGIR'08
- [Fonseca05] Fonseca, B. M., et al. Concept-based interactive query expansion. CIKM'05
- [Fox05] Fox et al. Evaluating implicit measures to improve web search. TOIS'05.
- [Fuxman08] Fuxman, A., et al. Using the wisdom of the crowds for keyword generation. WWW'08
- [Gao09] Gao, J., et al. Smoothing clickthrough data for web search ranking. SIGIR'09
- [Guo08] Guo, J., et al. A unified and discriminative model for query refinement. SIGIR'08
- [Guo09] Guo, F., et al. Click chain model in web search. WWW'09
- [Hassan10] Hassan, A. et al. Beyond DCG: user behavior as a predictor of a successful search. WSDM'10.
- [He02] He, D. et al. Combining evidence for automatic web session identification. Information Processing and Management, 2002.
- [Holscher00] Holscher, C. and Strube, G. Web search behavior of internet experts and newbies. International Journal of Computer and Telecommunications Networking, 2000
- [Horowitz07] Horowitz, A. et al. 101 dumbest moments in business, the year's biggest boors, buffoons, and blunderers. In CNN Money, 2007.
- [Hu09] Hu J., et al. Understanding user's query intent with Wikipedia. WWW'09.

References

- [Huang03] Huang, C.-K., et al. Relevant term suggestion in interactive web search based on contextual information in query session logs. *Journal of the American Society for Information Science and Technology*, 2003
- [Jansen00] Jansen, B. J., et al. Real life, real users, and real needs: a study and analysis of user queries on the web. *Information Processing & Management* 2000
- [Jansen01] Jansen, B. J., and Pooch, U. Web user studies: a review and framework for future work. *Journal of the American Society of Information Science and Technology* 2001
- [Jansen04] Jansen, B. J. and Spink, A. An Analysis of Documents Viewing Patterns of Web Search Engine Users. In *Web Mining: Applications and Techniques*. Editor: Anthony Scime, 2004
- [Jansen05] Jansen, B. J. and Spink, A. An analysis of web searching by European AlltheWeb.com users. *Information Processing and Management: an International Journal*. 2005
- [Jansen06] Jansen, B.J. and Spink, A. How are we searching the world wide web? a comparison of nine search engine transaction logs. *Information processing & management* 2006
- [Jin04] Jin et al. Web usage mining based on probabilistic latent semantic analysis. *KDD'04*.
- [Joachims02] Joachims, T. Optimizing search engines using clickthrough data. *KDD '02*
- [Joachims05] Joachims, T., et al. Accurately interpreting clickthrough data as implicit feedback. *SIGIR'05*
- [Jones06] Jones, R., et al. Generating query substitutions. *WWW'06*
- [Jones08] Jones, R. and Klinkner K.L. Beyond the session timeout: automatic hierarchical segmentation of search topics in query logs. *CIKM'08*.
- [Korolova09] Korolova, A., et al. Releasing search queries and clicks privately. *WWW'09*.
- [Kumar07] Kumar, R. et al. On anonymizing query logs via token-based hasing. *WWW'07*.
- [Li08] Li, X., et al. Learning query intent from regularized click graphs. *SIGIR'08*.
- [Liu02] Liu, F., Yu, C., and Meng, W. Personalized web search by mapping user queries to categories. *CIKM'02*.
- [Liu08] Liu, Y., et al. BrowseRank: letting web users vote for page importance. *SIGIR'08*
- [Liu09] Liu, C., et al. BBM: bayesian browsing model from petabyte-scale data. *KDD'09*
- [Mei08] Mei, Q. -Z. and Church, K. Entropy of search logs: how hard is search? with personalization? with backoff? *WSDM'08*.
- [Pasca07] Pasca, M. Organizing and searching the world wide web of facts--step two: harnessing the wisdom of the crowds. *WWW'07*

References

- [Pasca07a] Pasca, M. and Durme, B. V. What you seek is what you get: extraction of class attributes from query logs. IJCAI'07
- [Pitkow02] Pitkow, J. et al. Personalized search. Commun. ACM'02.
- [Piwowarski09] Piwowarski, B., et al. Mining user web search activity with layered Bayesian networks or how to capture a click in its context. WSDM'09.
- [Poblete08] Poblete, B. and Yates, R. B. Query-sets: using implicit feedback and query patterns to organize web documents. WWW'08
- [Pretschner99] Pretschner, A. and Gauch, S. Ontology based personalized search. IJCAI'99.
- [Qiu06] Qiu, F. and Cho, J. Automatic identification of user interest for personalized search. WWW'06.
- [Radlinski07] Radlinski, F. and Joachims, T. Active exploration for learning rankings from clickthrough data. KDD'07
- [Radlinski08] Radlinski, F. and Joachims, T. Minimally invasive randomization for collecting unbiased preferences from click-through data. AAAI'08
- [Radlinski05] Radlinski, F. and Joachims, T. Query chains: learning to rank from implicit feedback. KDD'05
- [Richardson07] Richardson, M., et al. Predicting clicks: estimating the click-through rate for new Ads. WWW'07
- [Sculley09] Sculley, D., et al. Predicting bounce rates in sponsored search advertisement. KDD'09
- [Shen05] Shen, X., et al. Context-sensitive information retrieval using implicit feedback. SIGIR'05.
- [Shen05a] Shen, X., et al. Implicit user modeling for personalized search. CIKM'05.
- [Silverstein99] Silverstein, C., et al. Analysis of a very large web search engine query log. SIGIR'1999
- [Silvestri09] Silvestri, F. and Yates, R. B. Query Log Mining. WWW'09 tutorial.
- [Speretta05] Speretta, M. and Gauch, S. Personalized Search Based on User Search Histories. WI'05.
- [Spink02] Spink, A., et al. From e-sex to e-commerce: Web search changes, Computer 2002
- [Spink02a] Spink, A., et al. US versus European Web searching trends. SIGIR'2002
- [Sun05] Sun et al. CubeSVD: a novel approach to personalized web search. WWW'05.
- [Sun05a] Sun, J.-T., et al. Web-page summarization using clickthrough data. SIGIR '05
- [Tan06] Tan, B., et al. Mining long-term search history to improve search accuracy. KDD'06.

References

- [Teevan05] Teevan, J., et al. Personalizing search via automated analysis of interests and activities. SIGIR'05
- [Teevan07] Teevan, J., et al. Information Re-Retrieval: Repeat Queries in Yahoo's Logs SIGIR'07.
- [Teevan08] Teevan, J. et al. To personalize or not to personalize: modeling queries with variation in user intent. SIGIR'08.
- [Vlachos04] Vlachos, M., et al. Identifying similarities, periodicities and bursts for online search queries. SIGMOD'04
- [Wang03] Wang et al. ReCom: reinforcement clustering of multi-type interrelated data objects. SIGIR'03.
- [Wang07] X. Wang and C. Zhai. Learn from web search logs to organize search results. SIGIR'07
- [Wedig06] Wedig S. and Madani, O. A large-scale analysis of query logs for assessing personalization opportunities. KDD' 06.
- [Welch08] Welch, M. J. and Cho, J. Automatically identifying localizable queries. SIGIR'08
- [Wen01] Wen, J.-R., et al. Clustering user queries of a search engine. WWW' 01
- [White07] White, R. W., et al. Studying the use of popular destinations to enhance web search interaction. SIGIR'07
- [White07a] White, R.W. and Drucker S.M. Investigating behavioral variability in web search. WWW'07.
- [White09] White, R.W., et al. Predicting user interests from contextual information. SIGIR'09.
- [Wolfram01] Wolfram, D., et al. Vox populi: the public searching of the web. Journal of the American Society of Information Science and Technology, 2001
- [Xiong07] Xiong L. and Agichtein, E. Towards privacy-preserving query log publishing. Workshop at WWW'07.
- [Xue04] Xue, G. R., et al. Optimizing web search using web click-through data. CIKM'04
- [Yates04] Yates, R.B., et al. Query Recommendation using Query Logs in Search Engines. EDBT workshop 2004.
- [Yates04a] Yates, R. B., et al. Query Clustering for Boosting Web Page Ranking. AWIC'04.
- [Yi09] Yi, X., et al. Discovering users' specific geo intention in Web search. WWW'09
- [Zhao06] Zhao, M., et al. Adapting document ranking to users' preferences using click-through data. AIRS'06
- [Zhu09] Zhu, G. and Mishne, G. Mining rich session context to improve web search. KDD'09