Enterprise and Desktop Search

Lecture 3: Exploratory search

Pavel Dmitriev, **Pavel Serdyukov**, Sergey Chernov

Yahoo! Labs,
Sunnyvale, US

Delft University
Of Technology
The Netherlands

L3S Research Center,
Hannover, Germany

Outline

- Exploratory search and ways to support it
- Faceted search:
  - Interfaces
  - Interaction styles
- Faceted search solutions:
  - with structured metadata
  - with unstructured metadata
  - without ready-made metadata
- Future challenges

Relevance in the Enterprise

Search in enterprise is hard!

Initial guess is often wrong

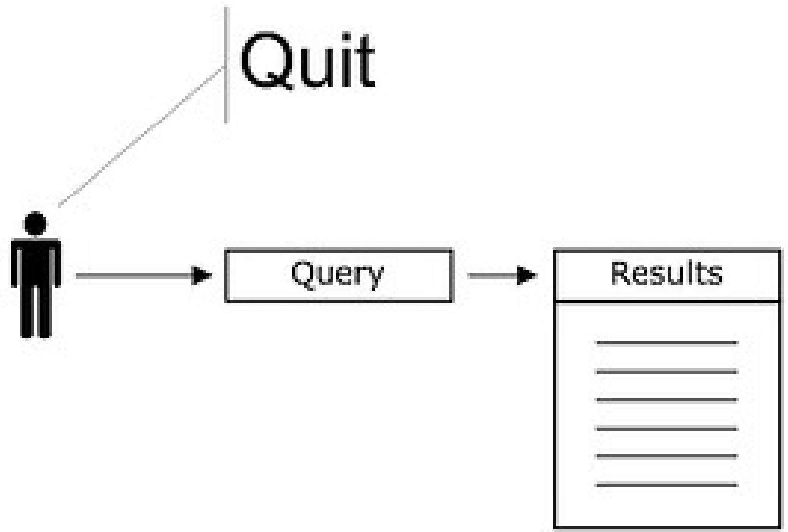
Users want to be aware of everything in the Enterprise

Users demand

more **control** over search!

They want to **explore**!

Search is a look-up?

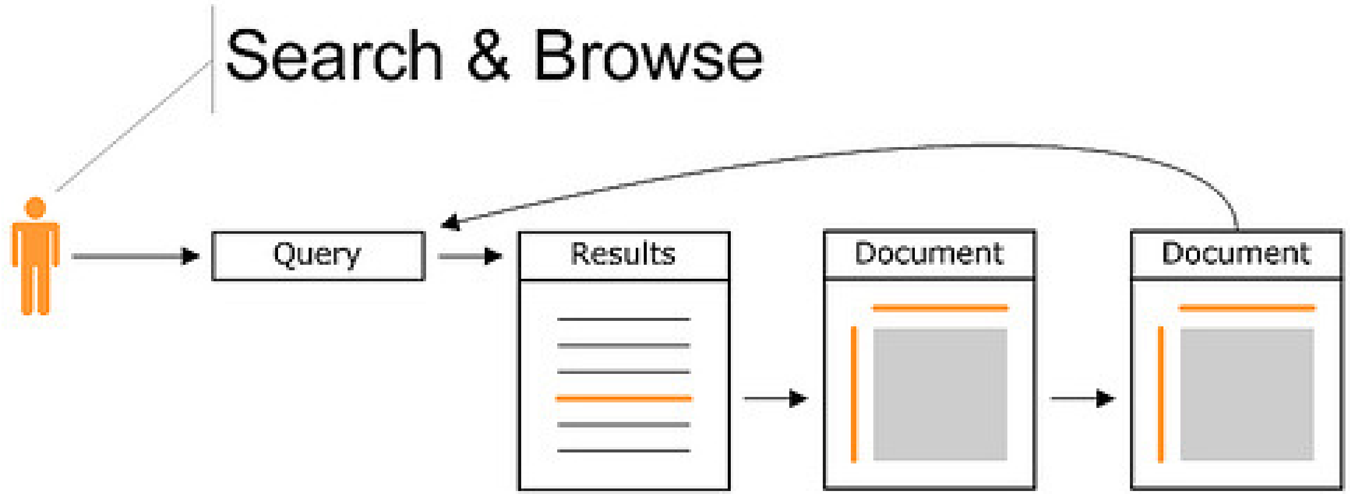


Is that all?

Certainly not in enterprises

http://www.flickr.com/photos/morville

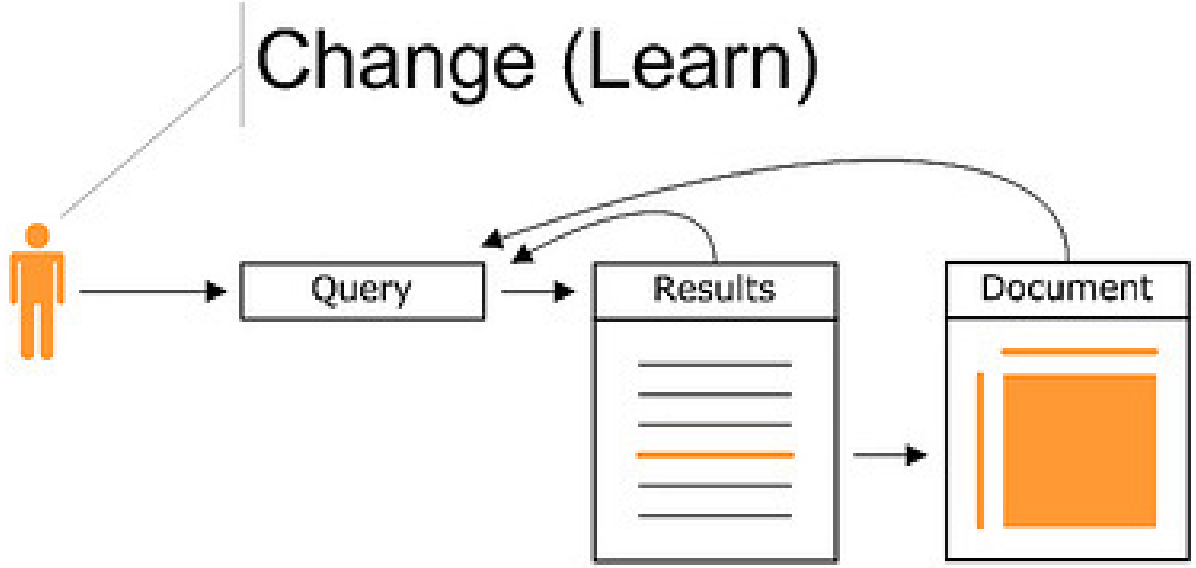
Search is a journey!



- Exploratory search involves:
  – browsing the result
  – analyzing returned documents
  – coming back to the initial ranking again and again

http://www.flickr.com/photos/morville

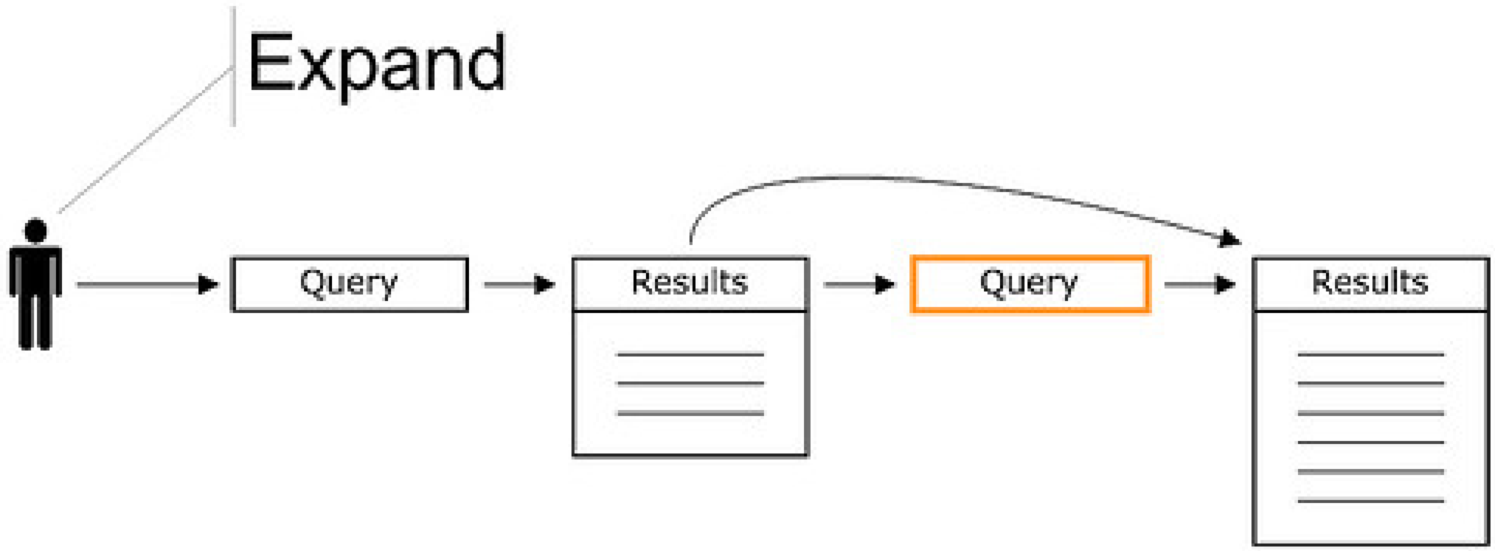
Search is a journey!



- Exploratory search involves:
  - Querying the last returned result set
  - Looking for similar documents (relevance feedback)

http://www.flickr.com/photos/morville

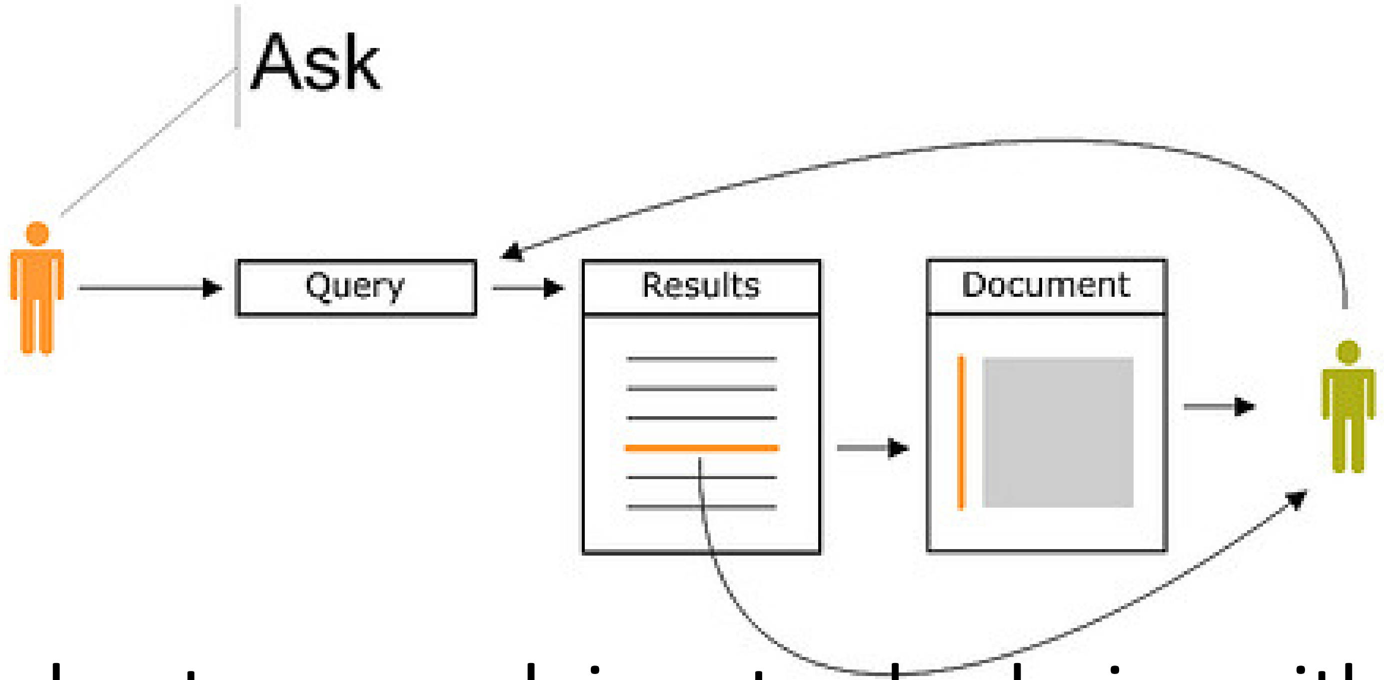
Search is a journey!



- Exploratory search is also about...
  - Query reformulation, same information need:
    - Specialization: **mp3 players** => **ipod**
    - Generalization: **ipod** => **mp3 players**

http://www.flickr.com/photos/morville

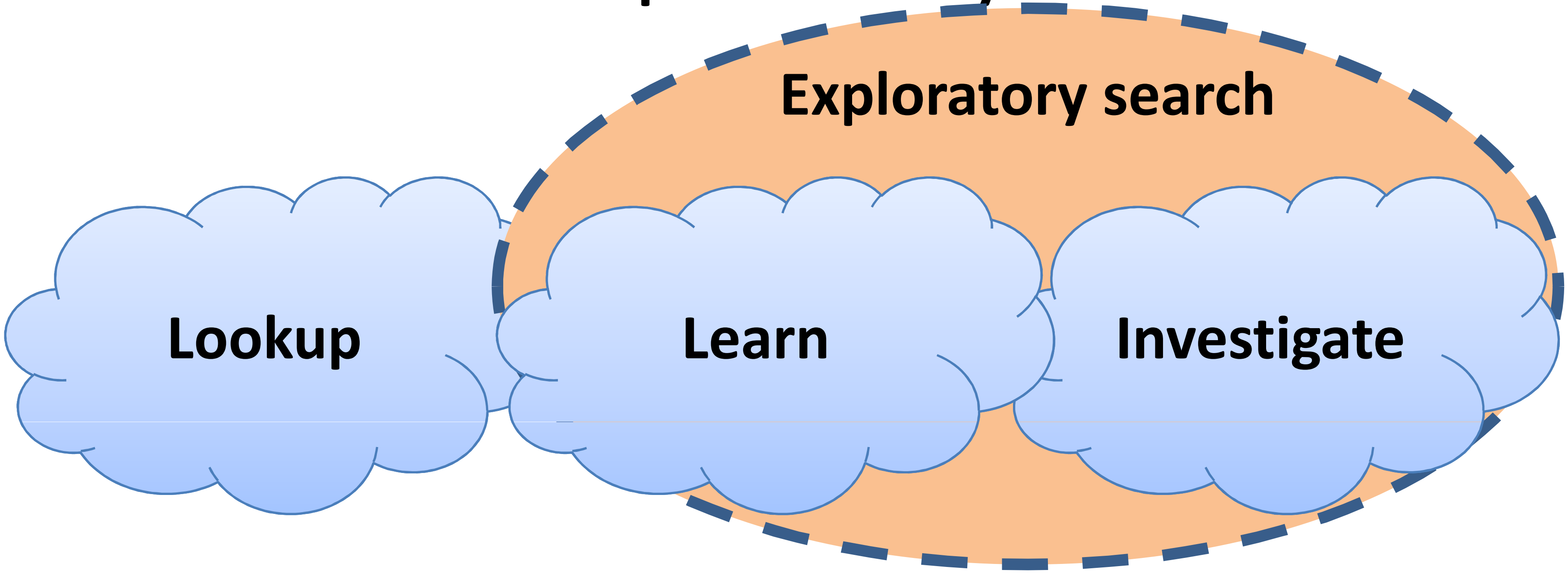
Search is a journey!



- Exploratory search is not only playing with a search box, but also… looking for people:
  - Who know the answer
  - Who know where to find answers
  - Who know much more than just an answer

http://www.flickr.com/photos/morville

What is exploratory search



Question answering
Fact retrieval
Known-item search
Navigational search
Lasts for seconds

Knowledge acquisition
Comprehension
Comparison
Discovery
Serendipity

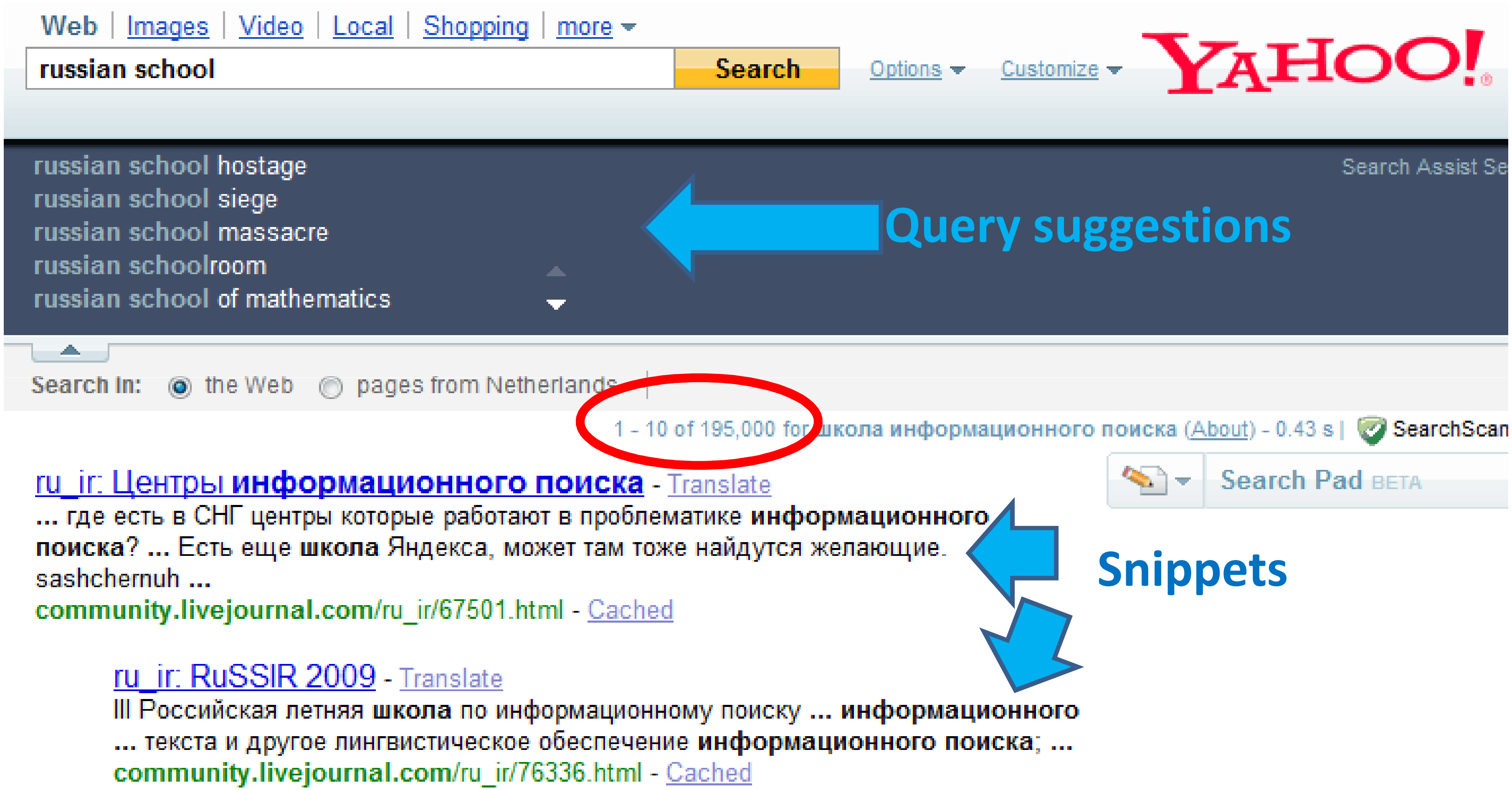
Incremental search
Driven by uncertainty
Non-linear behavior
Result analysis
Lasts for hours

Exploratory search: from finding to understanding.
Marchionini. Commun ACM. 2006

Support exploratory behavior

- Support learning
  - About the search topic
  - About the collection
- Support query reformulation
  - Broadening
  - Narrowing
  - Changing the focus
- Support socialization
  - Looking for experts
  - Collaborative search

What web search engines offer



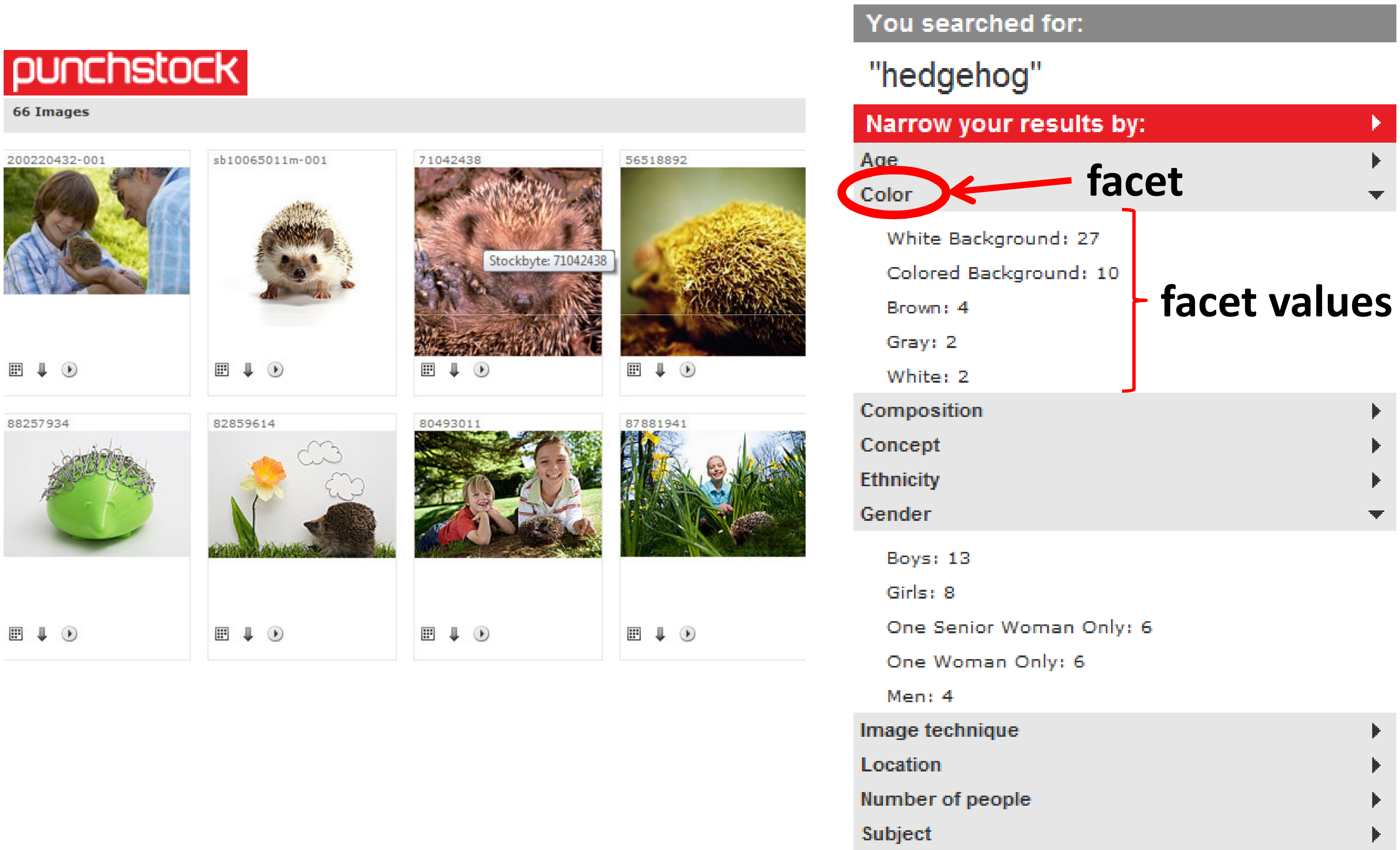
Does it really help to learn?

Can we do better?

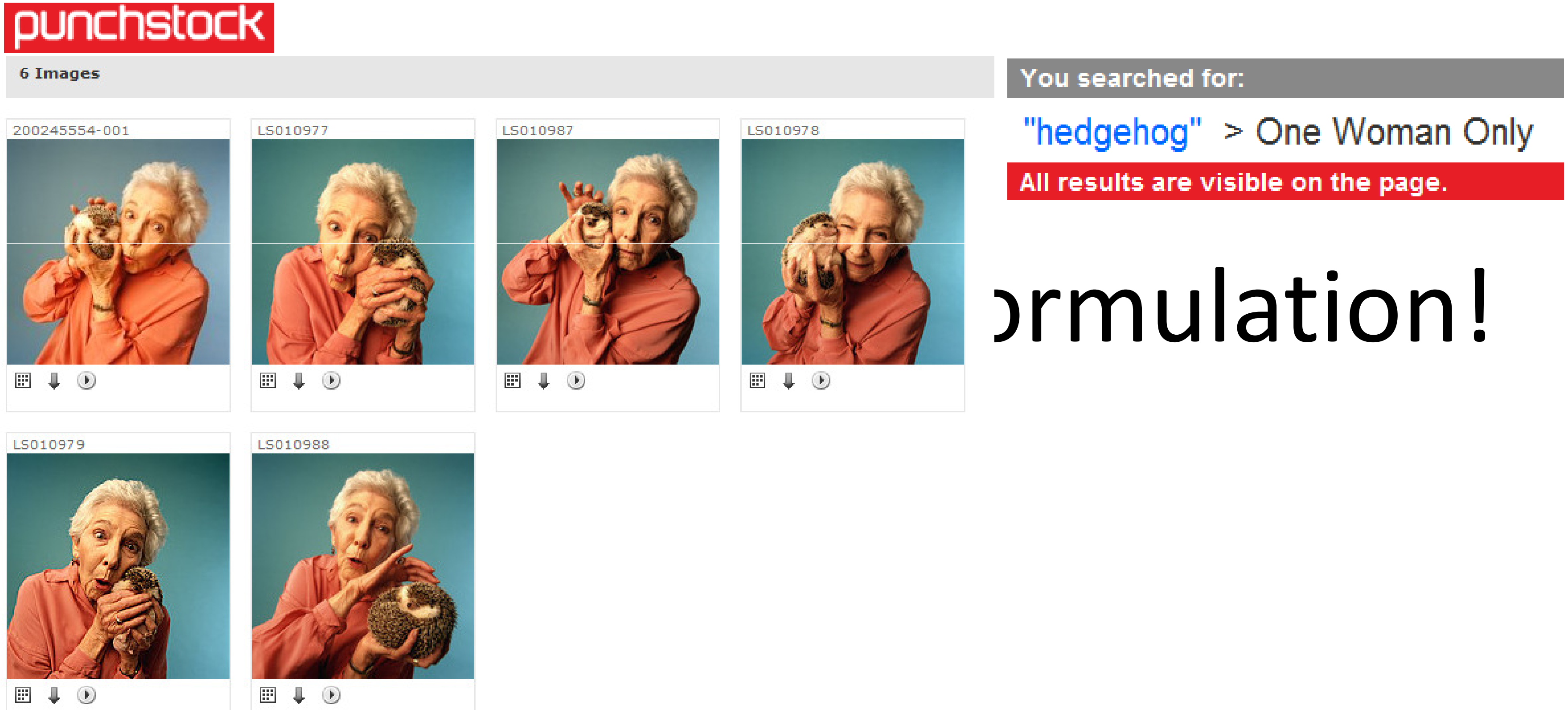
- Certainly, when we have metadata for docs!
  - So, some summarization is done for us
- Structured metadata:
  - Classic **faceted search** scenario
- Unstructured metadata
  - Tag-based analysis and navigation
- No metadata?
  - Result clustering
  - More? Let’s see…

Faceted search:
with structured metadata

What is faceted search?

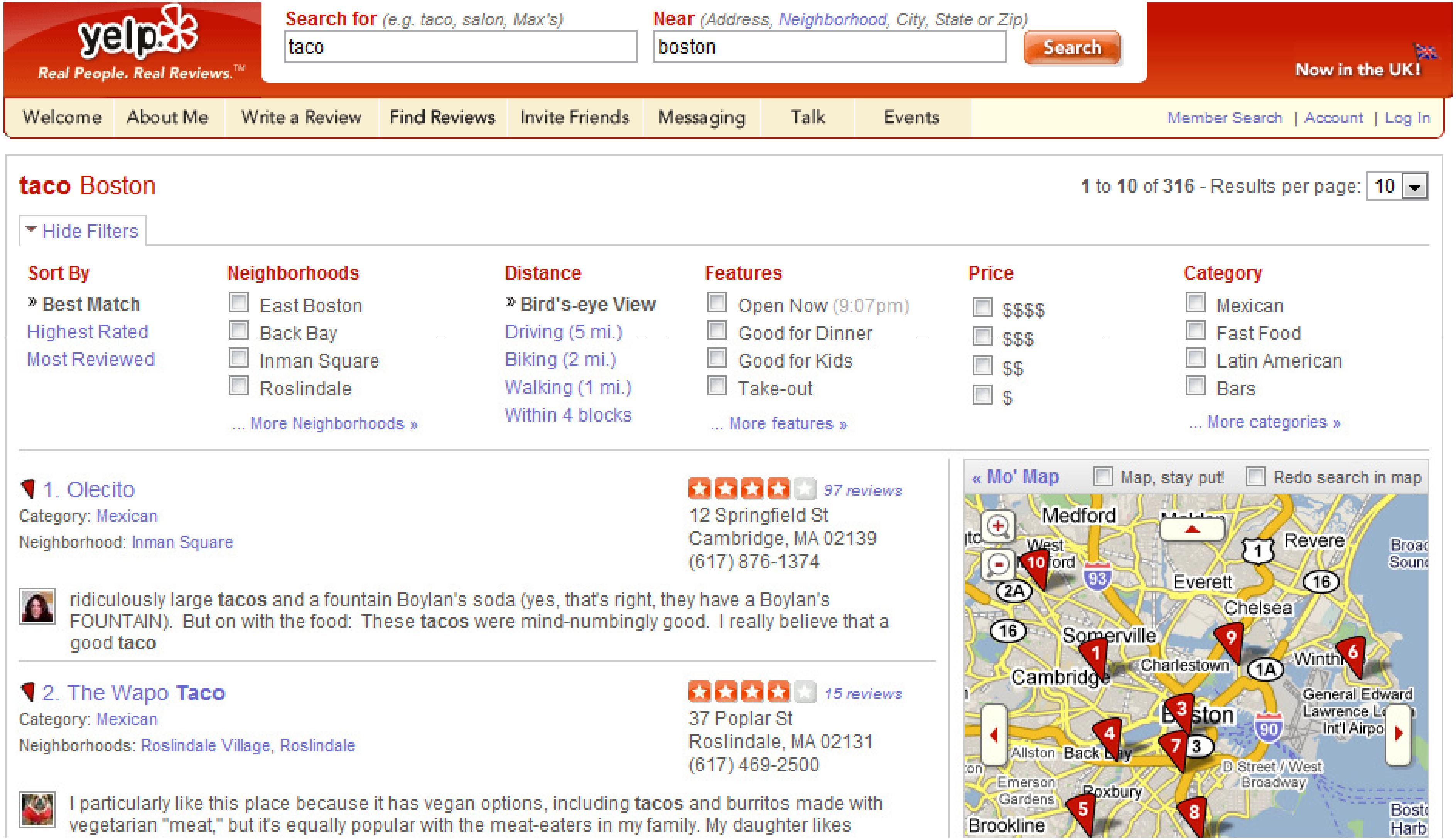


What is faceted search?

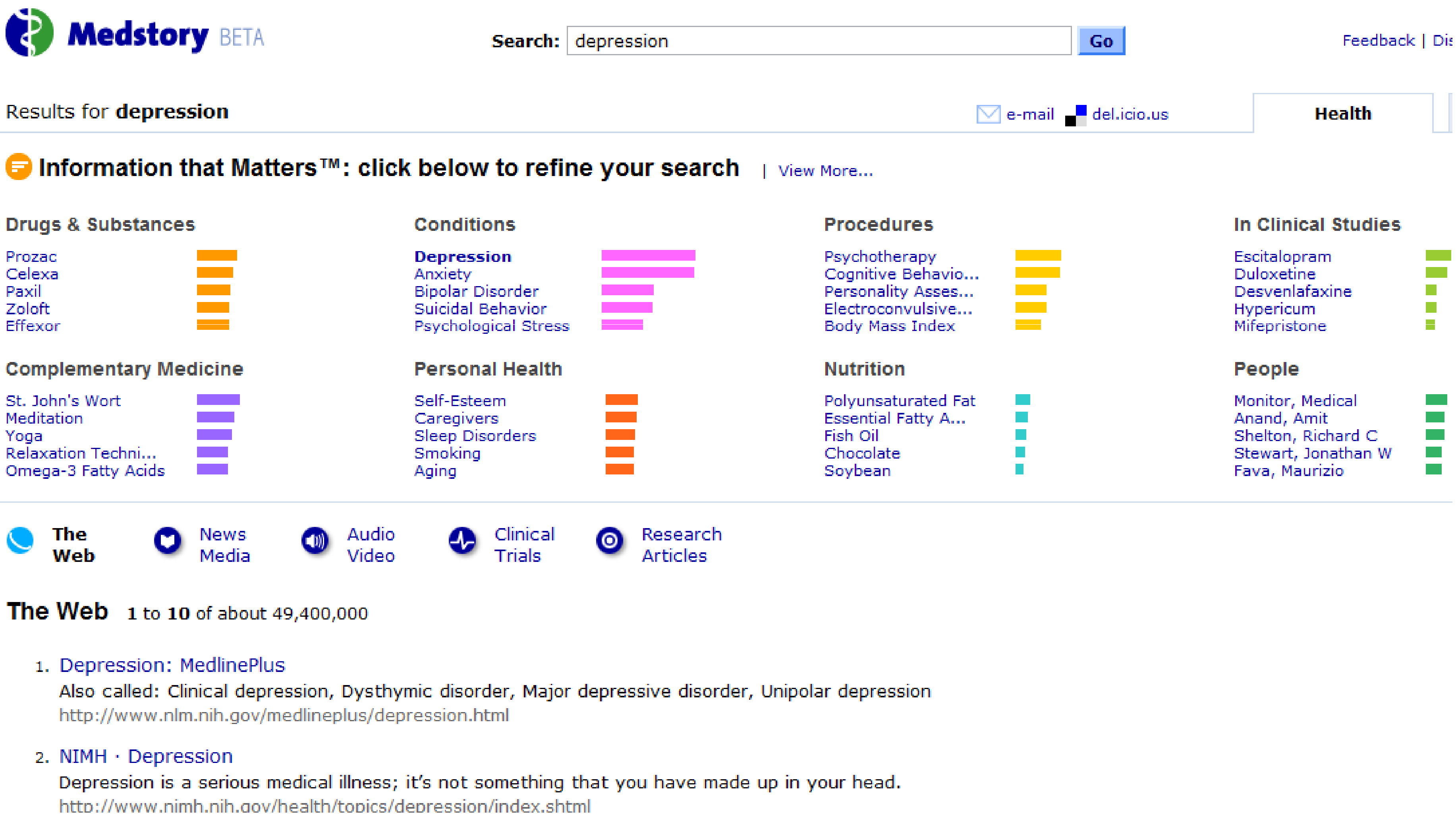


rmulation!

What is faceted search?

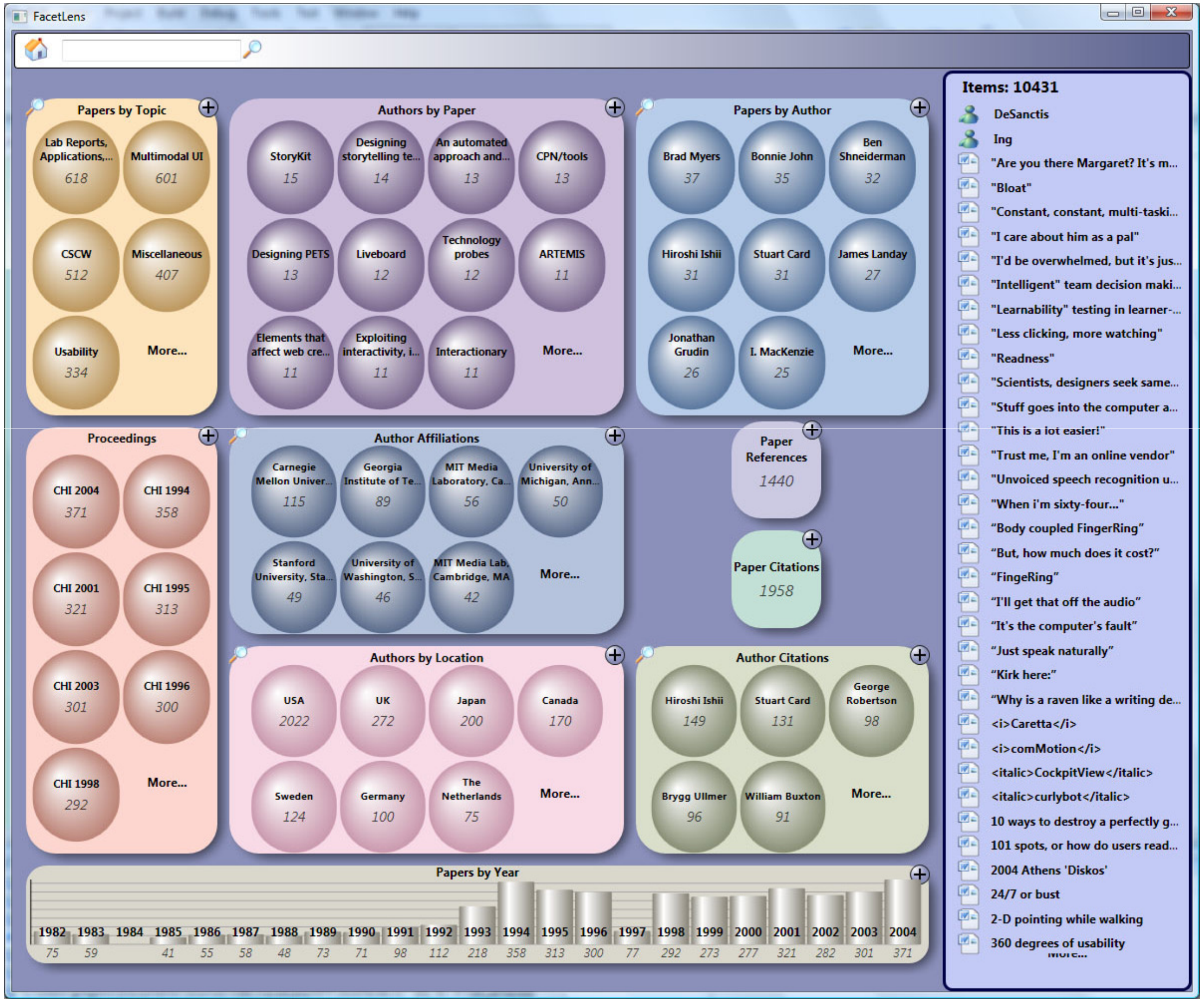


What is faceted search?



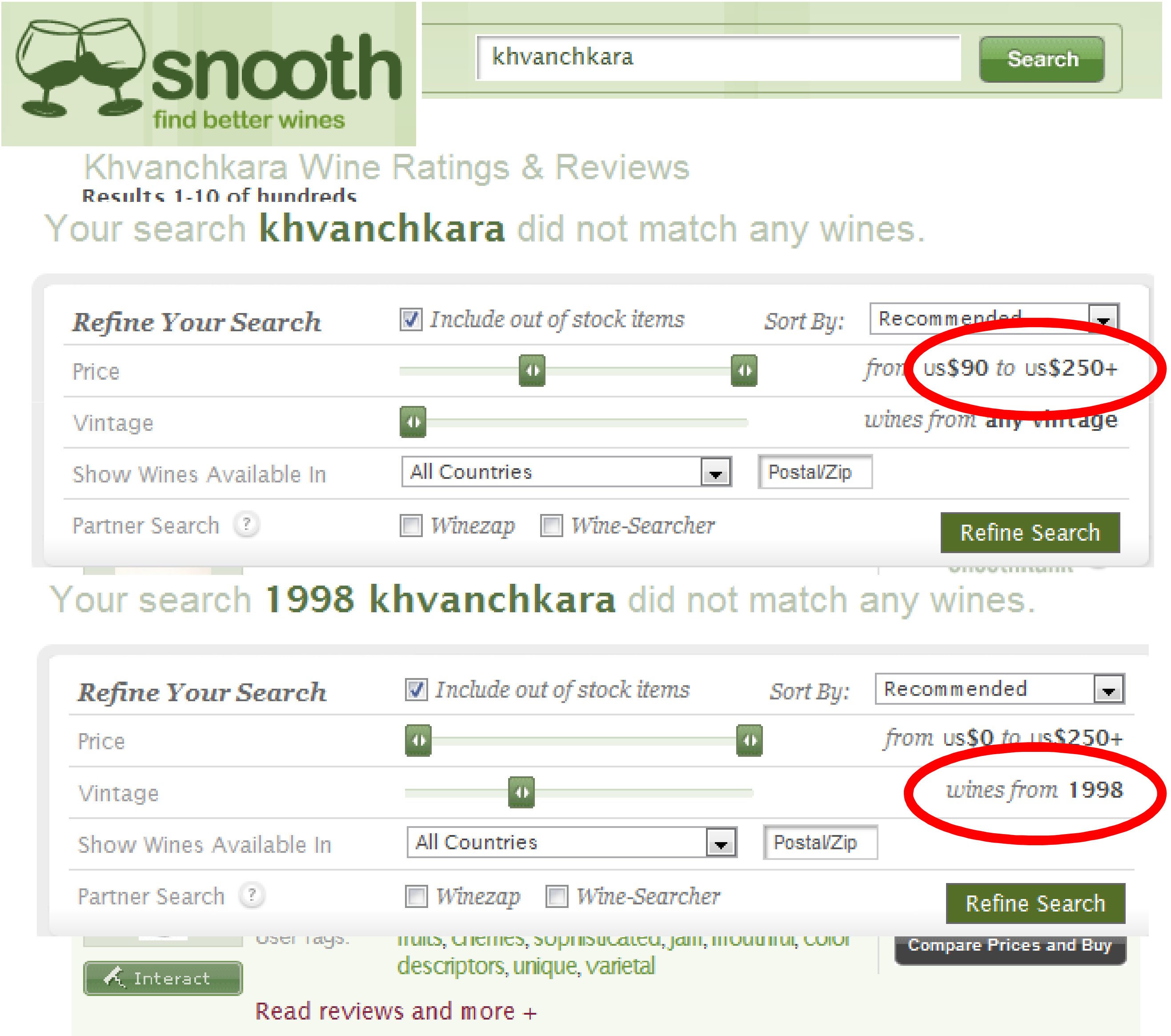
What is faceted search?

What is faceted search?



FacetLens (Microsoft Research)

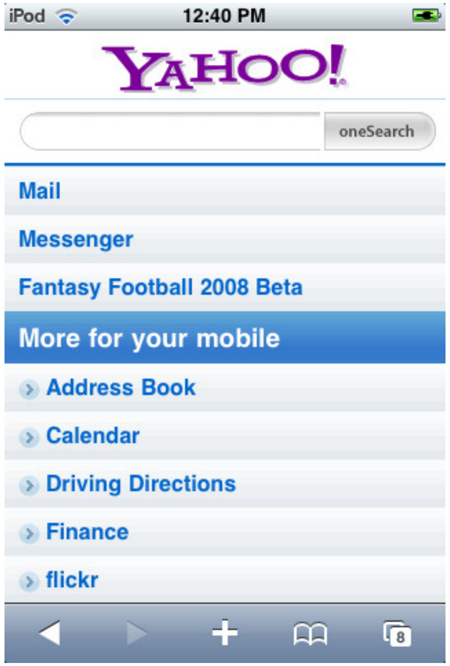
What is not faceted search?

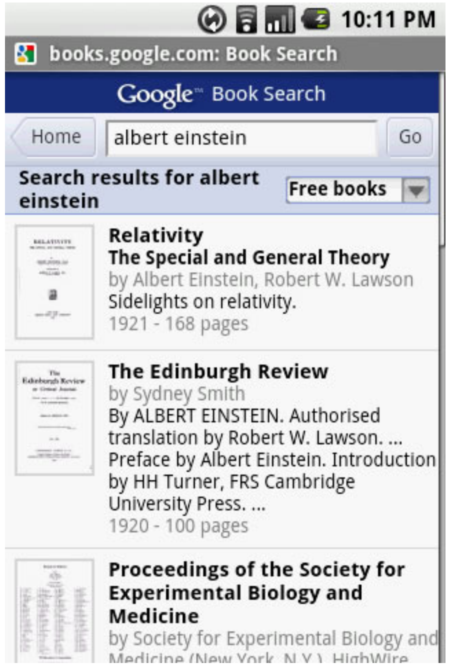


Too many facets ?
Too many facet values?

Information overload

Mobile interfaces

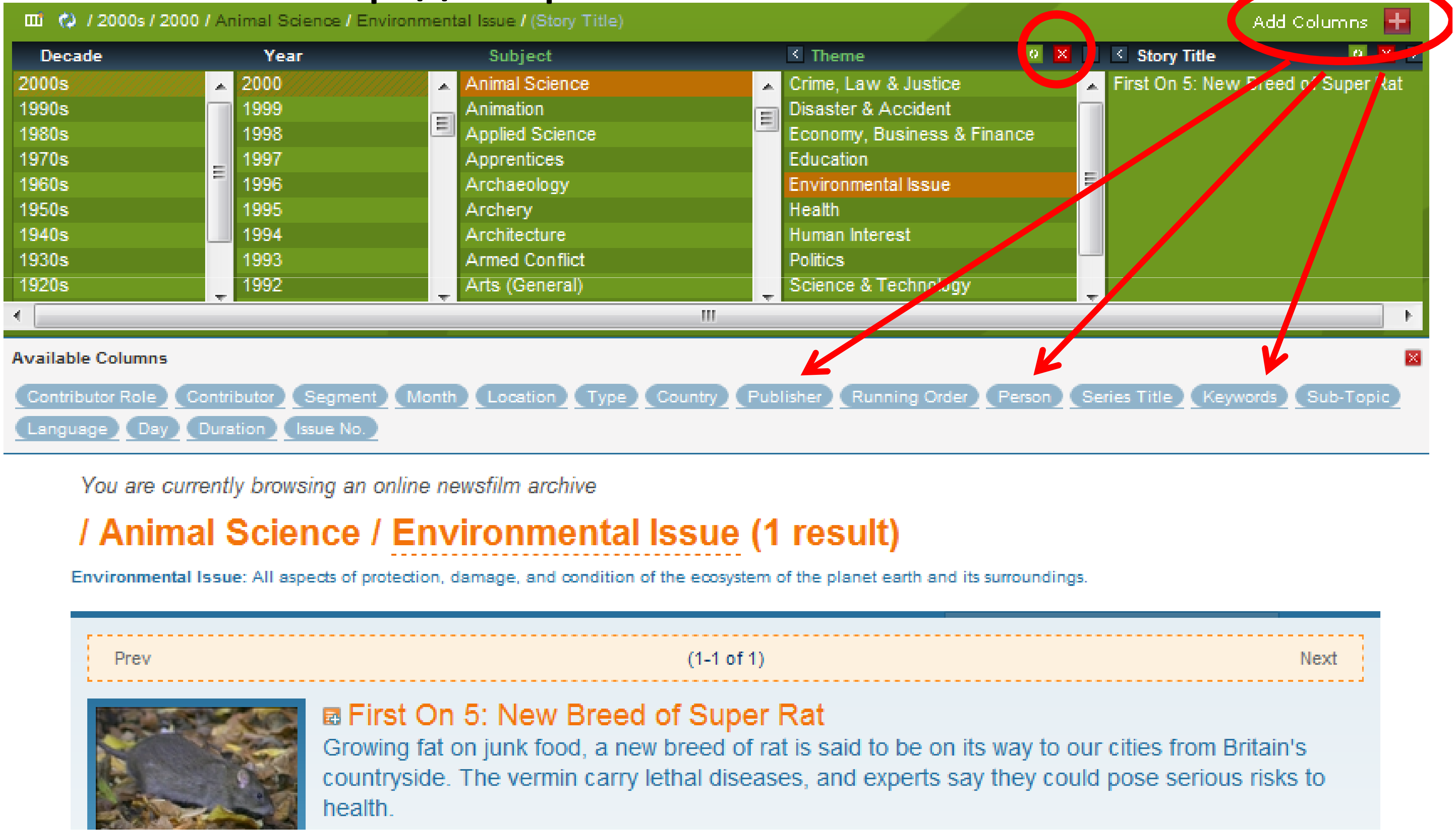




Facet selection: interface-based approach

mspace
exploration reinvented

http://mspace.fm



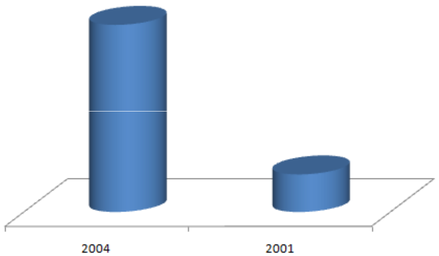
Redundancy-based facet selection

- Favor facets with high coverage in the result
  - Plenty of data formats in the enterprise
  - Metadata is not unified
  - There is no one classification scheme
  - **Select most frequent facets!**
- Avoid presenting highly correlating facets*
  - So, either language or nationality
- Consolidate similar facets:
  - author, editor, contributor => **people**

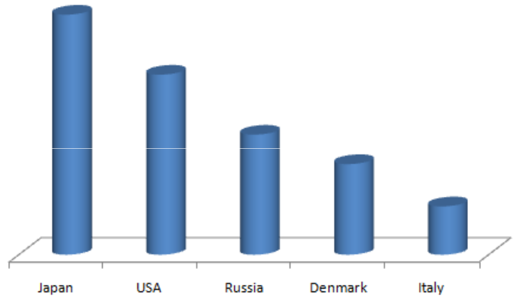
***Beyond Basic Faceted Search.** Ben-Yitzhak et. al. WSDM 2008

Interestingness-based facet selection

- Measure surprisingness of values distribution
- Favor facets with high-entropy distribution



vs



$$Entropy = \sum_{i=1}^{n} P(w_i \mid R) \log P(w_i \mid R)$$

- Favor facets with query-specific distribution

$$Divergence = \sum_{i=1}^{n} (P(w_i \mid C) - P(w_i \mid R)) \log \frac{P(w_i \mid R)}{P(w_i \mid C)}$$

Facet values ranking

- Measure ***Relevance*** of facet value!
- Rank by frequency in result set
  - Most popular approach
- Rank by $\dfrac{P(f = v_i \mid R)}{P(f = v_i \mid C)}$
- Rank by aggregated document relevance:
  - Sum scores of all documents with value v_i

$$Relevance(v_i) = \sum_{\substack{Doc \in Result, \\ Doc(f) = v_i}} Score(Doc)$$

Collaborative facet values ranking (I)

- Suppose we have long history of interactions
  - Queries + returned documents
  - Maybe even clicks
  - Maybe even documents judged as relevant
- So, let's build a user model!
- User preferences over all ever issued queries:

$$\frac{\sum_{R \in User_k} P(f = v_i \mid R)}{P(f = v_i \mid C) \cdot |User_k|}$$

for result sets of all issued queries

Number of queries

Collaborative facet values ranking (II)

- Utilize collaborative filtering techniques*:

$$\alpha \frac{\sum_{R \in User_k} P(f = v_i \mid R)}{P(f = v_i \mid C) \cdot |User_k|} + (1-\alpha) \underbrace{\frac{\sum_{User_j \in Users} \frac{\sum_{R \in User_j} P(f = v_i \mid R)}{|User_j|}}{P(f = v_i \mid C) \cdot |Users|}}_{\text{average preferences over all users}}$$

- Consider only users with similar tastes:

$$\alpha \frac{\sum_{R \in User_k} P(f = v_i \mid R)}{P(f = v_i \mid C) \cdot |User_k|} + (1-\alpha) \frac{\sum_{User_j \in Users} sim(User_k, User_j) \frac{\sum_{R \in User_j} P(f = v_i \mid R)}{|User_j|}}{P(f = v_i \mid C) \cdot |Users|}$$

For example, cosine similarity
or divergence of distributions

***Personalized Interactive Faceted Search.** Koren et. al. WWW 2008

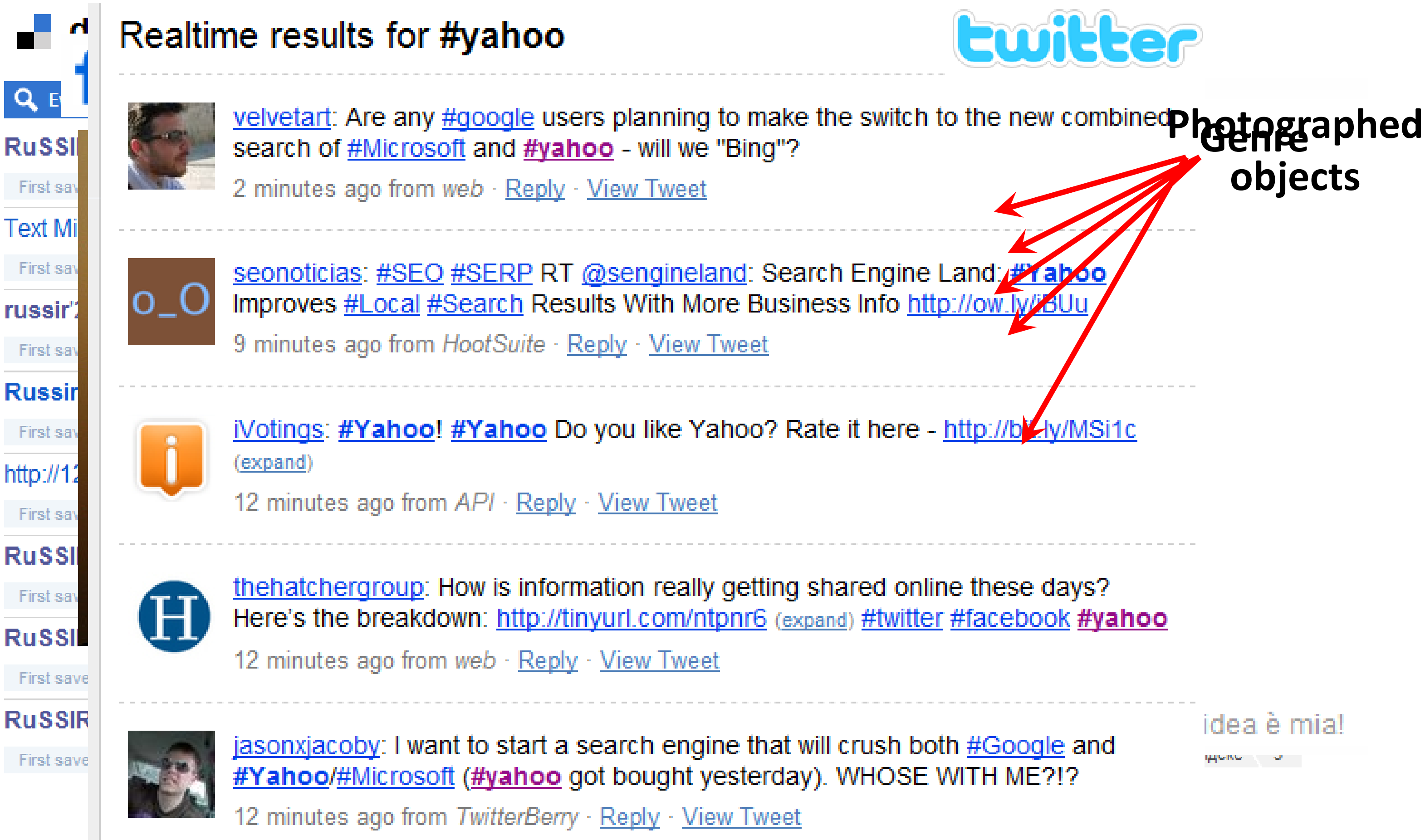
Summary

- Faceted search is must
  - When metadata is structured
- Interfaces are crucially important to satisfy the user and help to learn
  - Need to be simple, but customizable
  - Allow to **navigate** the result
- Summarization should be
  - Result-set oriented
  - Giving answers right away
- Facets/values should be selectively presented!

Faceted search with unstructured metadata: Tags!

Tagging

- Make the way to annotate as easy as possible
- Get metadata for free



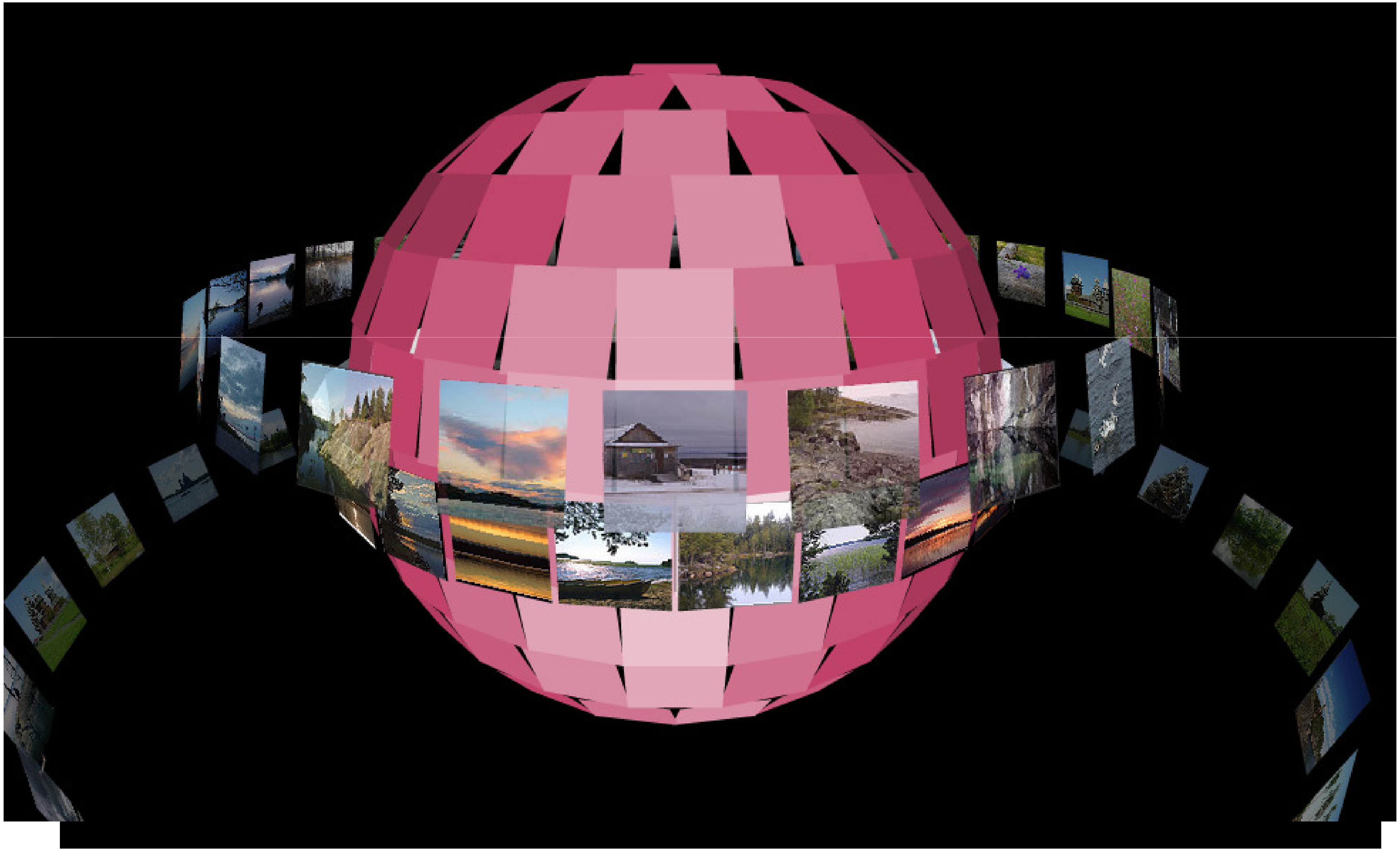
Tagging

- Disadvantages:
  - Nor ranked by relevance to the tagged resource
  - Not organized
  - Not categorized
- But still plenty of ways to summarize!
  - Find “relevant” tags
  - Demonstrate their importance to the user
  - Guess the tag purpose
  - Guess the tag meaning

Tag cloud

http://www.wordle.net/

Tag space



http://taggalaxy.de/

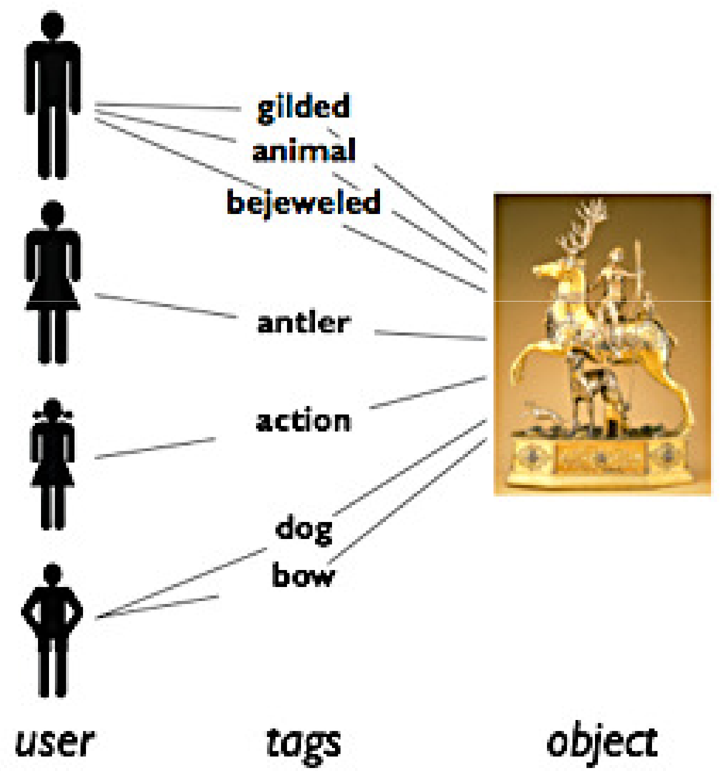
How to measure tag size?

$$fontsize_i = \frac{fontsize_{\max}(tfidf_i - tfidf_{\min})}{(tfidf_{\max} - tfidf_{\min})}$$

tf – tag frequency in the result set
idf – inverted tag frequency in the collection
tfidf – non-normalized tag importance

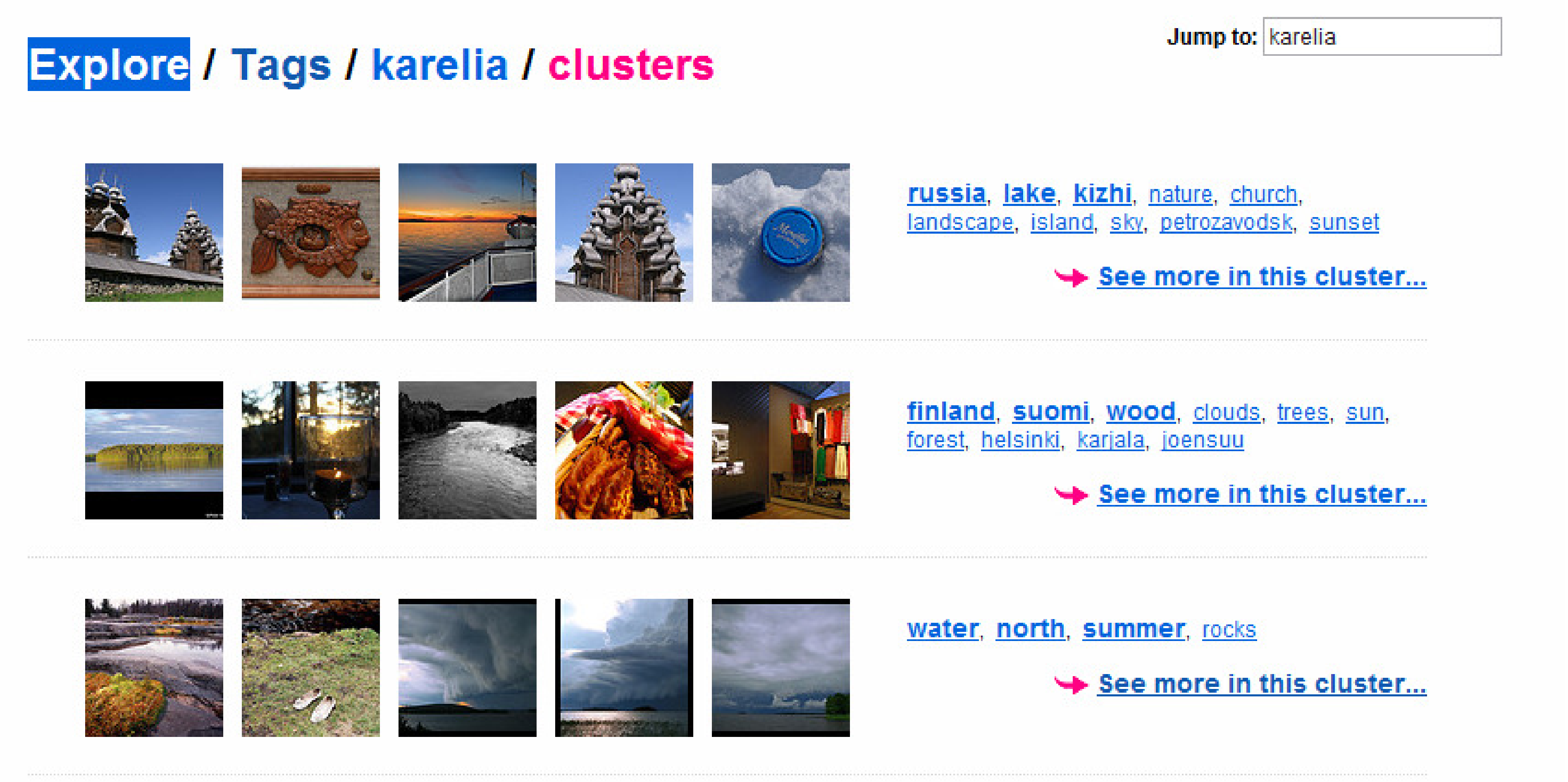
Cloud or clouds?

- Group tags by topic!
- Cluster them*!
- Similarity function?
- Tags as vectors of objects
  – But tagging can be non-collaborative
- Tags as vectors of users
  – But co-occurrence less meaningful



***Personalization in folksonomies based on tag clustering.** Gemmel et. al. AAAI 2008

Flickr example

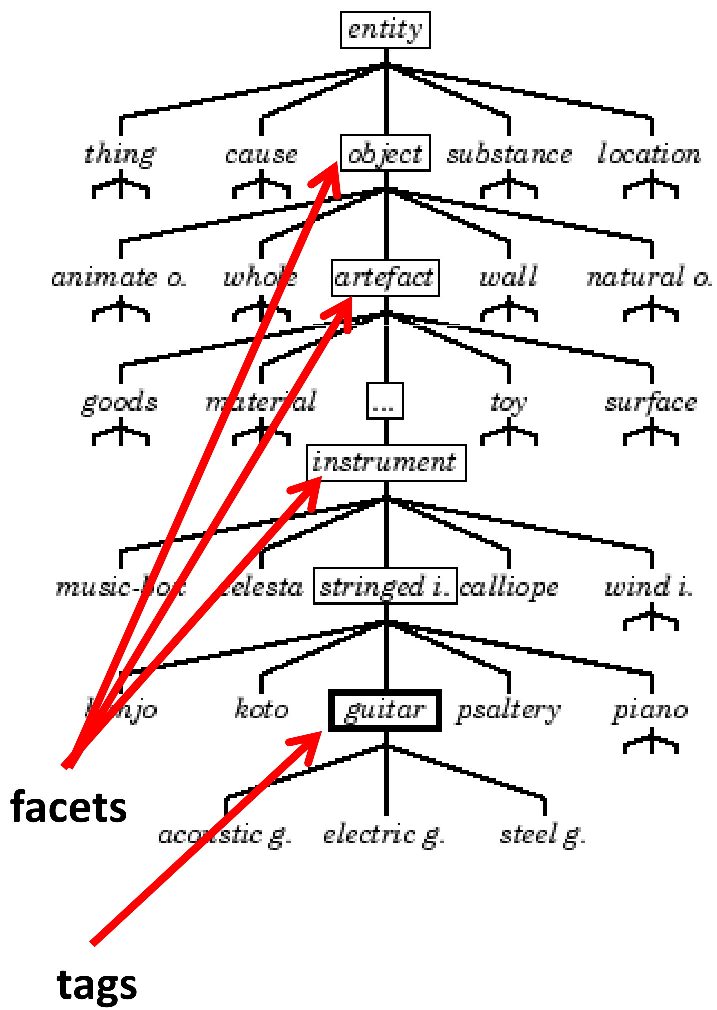


Tag classification for faceted search

- Clusters are nice, but…
  - Random
  - Not always of high quality
- We need some knowledge-based classification
  - To discover more meaningful structure
  - To represent tags as values of facets (classes)
  - To provide the feeling of control for users
- Who knows everything about a word (tag)?
  - Lexical databases: **Wordnet**
  - Encyclopedias: **Wikipedia**

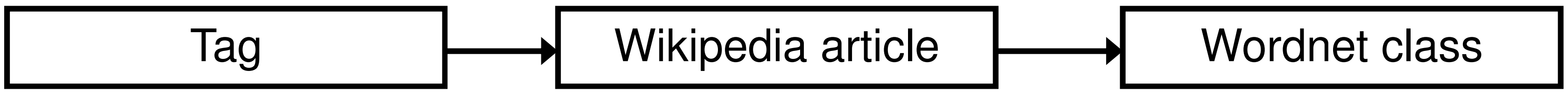
Tag classification with Wordnet

- Contains various semantic relations between word senses
  - guitar is a type of instrument
  - string is part of guitar
  - java is a type of island OR coffee OR language
- About 150 000 senses
  - of 120 00 nouns
- Match tags to nouns
- Disambiguate!
  - Find senses with minimum distance to each other in this graph



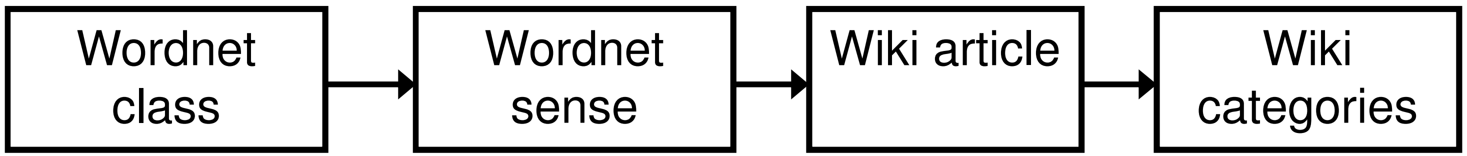
Tag classification with Wikipedia (I)

- Wordnet has nice selection of classes (facets)
- … but no so many entities (facet values)
- Let's use larger knowledge repository… **Wikipedia** - more than 3 million articles!
- But it has too many classes (categories)
  - ~ 400,000, their hierarchy is very fuzzy
- Use Wikipedia **just** as a middle layer!

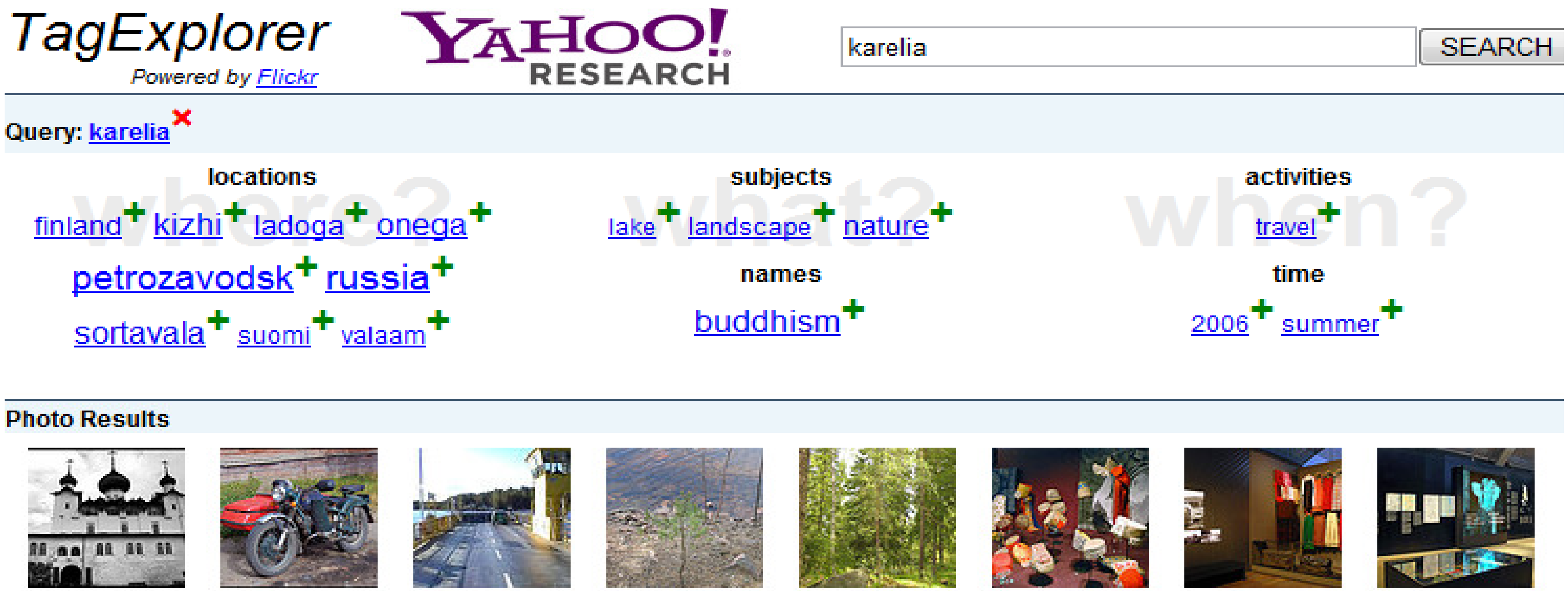


Tag classification with Wikipedia (II)

- Direct Tag => Wiki matches may be too imprecise:
  - So, use only anchor text or titles
- Some Wikis are direct match with Wordnet senses!
  - "Guitar" => en.wikipedia.org/wiki/Guitar
  - Use these matches as training data
  - Build classifier for each Wordnet noun class (~25 classes)
- What features should describe Wordnet classes?
  - Using terms as features would introduce too much **noise** and problems with **dimensionality**
  - Categories of wiki-articles are better choice!



http://tagexplorer.sandbox.yahoo.com/



- Classified 22% of Flickr tags with Wordnet
- Classified 70% of Flickr tags with Wikipedia

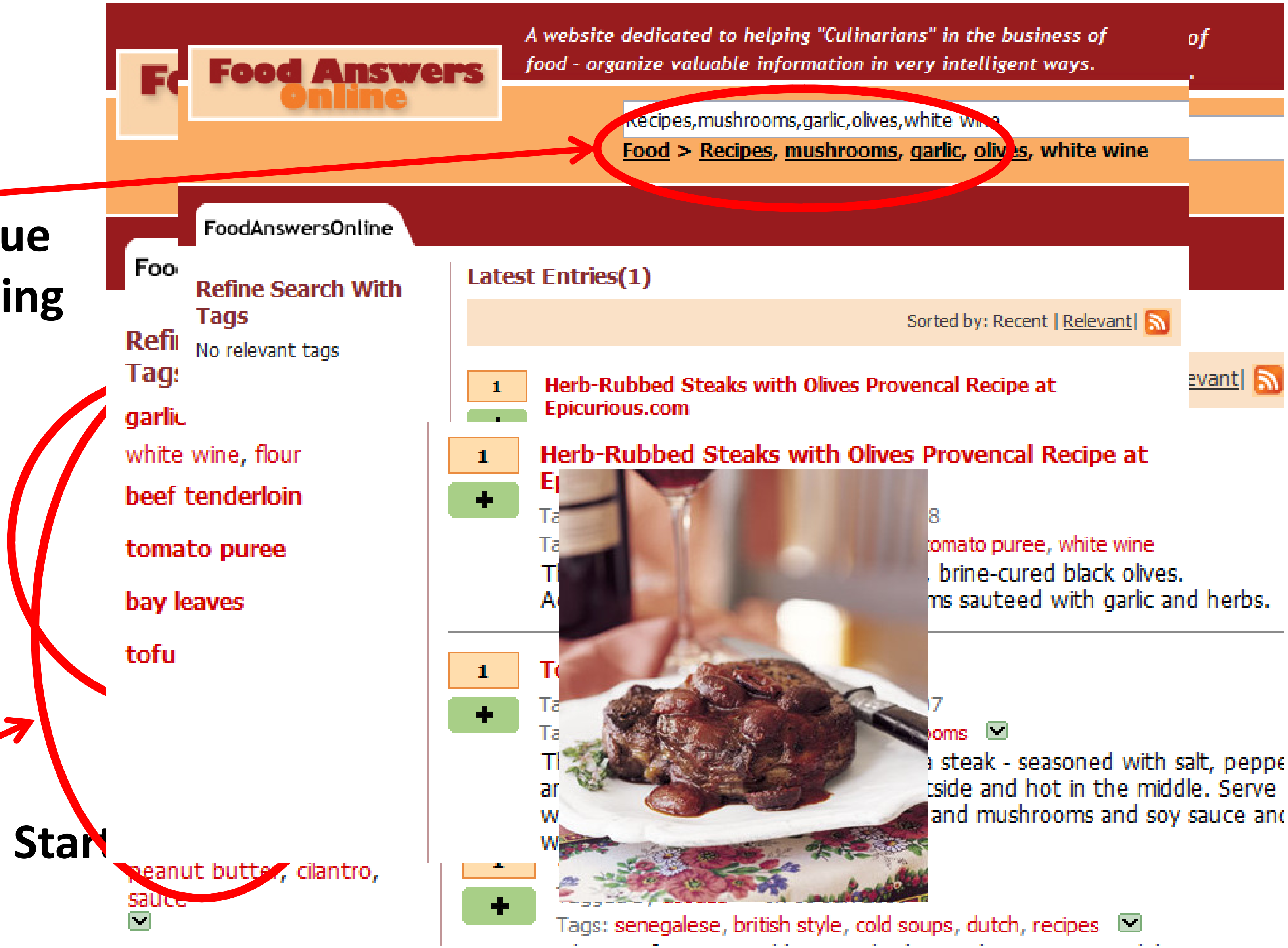
Classifying Tags using Open Content Resources. Overell et. al. WSDM 2008

Interaction with faceted search system

- Traditional way:
  - Typing, typing, typing…
  - For the sake of query reformulation
- Faceted search?

Mousing & Browsing

Filtering – all search tags are made equal



Continue narrowing

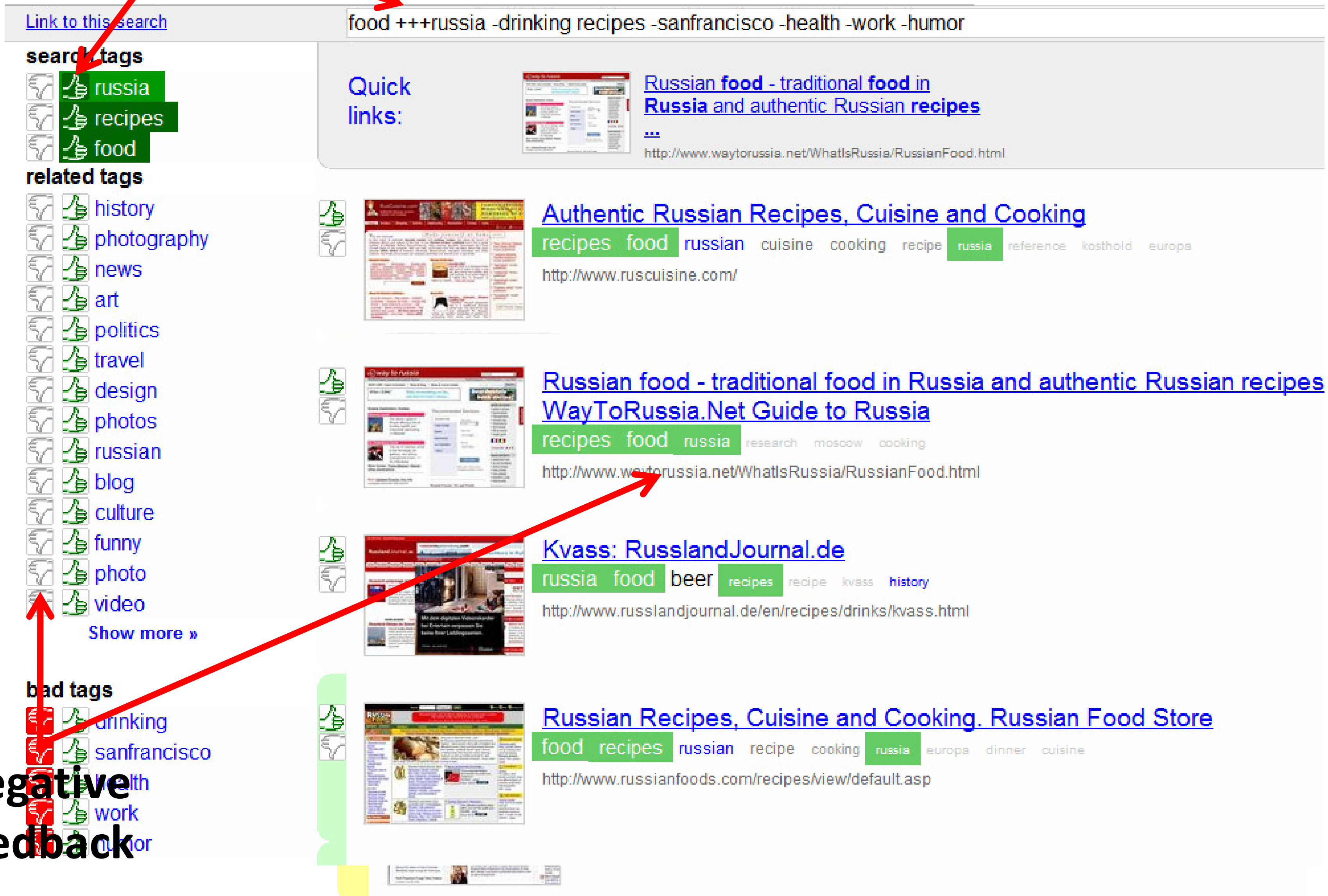
Continue

Start

Tag feedback

MrTaggy PROTOTYPE

Tag weights



Negative feedback

How to incorporate feedback (I)

$$Score(Q, D) = -D(\theta_Q || \theta_D) + \beta \cdot D(\theta_N || \theta_D)$$

Relevance lang. model

food +++russia recipes

Irrelevance lang. model

-drinking –health –work -humor

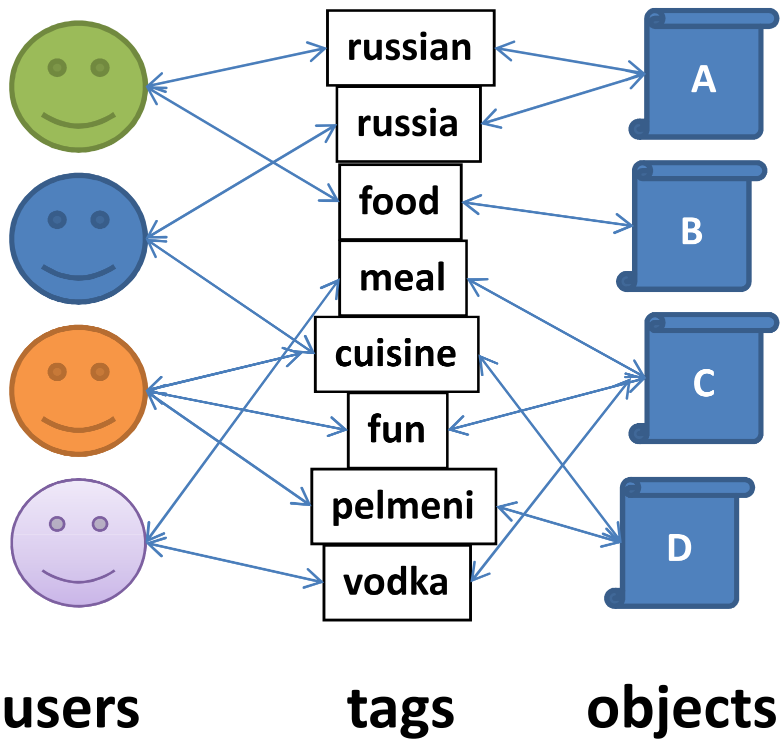
$$P('food'|Q) = 1/5$$

$$P('recipes'|Q) = 1/5$$

$$P('russia'|Q) = 3/5$$

A study of methods for negative relevance feedback Wang et. al. SIGIR 2008

How to incorporate feedback (II)



- We have a tripartite graph
  - Many tags are related, but not used in our query
- It’s good **to be close to positive** tags
- It’s good **to be far from negative** tags

How to incorporate feedback (III)

- Express language models in graph terms:

$$P(tag \mid Document) = \frac{Distance(tag, Document)^{-1}}{\sum_{tag \in alltags} Distance(tag, Document)^{-1}}$$

- How to define **distance** between nodes:
  - Length of shortest path
  - Number of shortest paths (of certain length)
  - Distance-based similarity: $\sum_{\substack{path(tag, document) \\ \in shortestpaths}} c^{-length(path)}$

$$c - parameter$$

- What else to consider?
  - Downweight paths with nodes of high indegree/outdegree

Summary

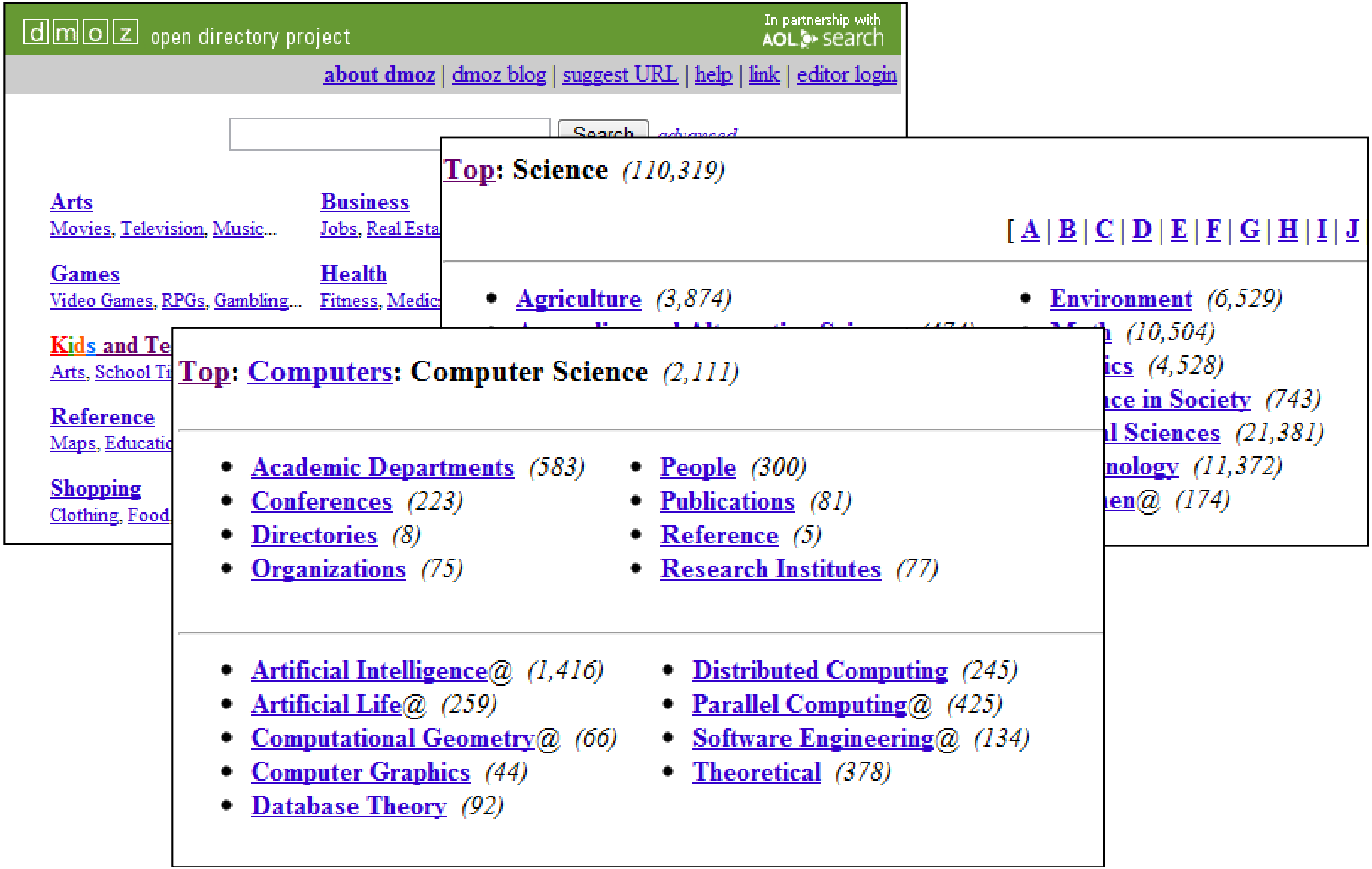
- Faceted search is possible with unstructured metadata…
  - But we need to make some effort **to structure** it!
- Visualization is always important
  - But not enough to understand the summary
- So, it’s better to explain the result
  - By clustering tags/objects
  - By classifying tags/objects into semantic categories
- And, finally, it’s about navigation and click-based query reformulation
  - Provide ways to react for the user
  - Provide ways to give different kinds of feedback

Faceted search:
No metadata!

No metadata? No panic!

- Facet-value pairs are manual classification
- Tags are basically important terms
- Why not classify automatically?
  - Categorize into known topics
  - Cluster and label clusters
- Why not automatically discover tags?
  - Extract important keywords from documents
- Well, some metadata always exists
  - Time, source….

Categorize by topic (I)



Categorize by topic (II)

- Document categorization
  - Shallow (Flat) vs. Deep (Hierarchical)
- Shallow classification: only top level
  - Makes no sense for very focused queries:
    java vs. **biology**
- Deep classification*:
  - Lack of training examples (labeled documents) with each next level of hierarchy
  - Documents can be assigned to **too many classes**

Deep Classifier: Automatically Categorizing Search Results into Large-Scale Hierarchies. Xing et. al. WSDM 2008

Categorize by topic (III)

- Solution for sparsity:
  – Suppose, we use Bayesian classification

$$P(Class \mid D) = P(Class)\prod_{w=1}^{|D|} P(w \mid Class)$$

$$P^{smoothed}(w \mid "Databases") =$$

$$= \lambda_1 P(w \mid "Databases") + \lambda_2 P(w \mid "ComputerScience") + \lambda_3 P(w \mid "Science"), \sum \lambda_i = 1$$

- Solution for “too many classes” problem
  – Many documents focus on several topics
  – Let’s care only about those that user cares about:

$$P(Class \mid D) \Rightarrow P(Class \mid D, Q) = P(Class \mid D)P(Class \mid Q)$$

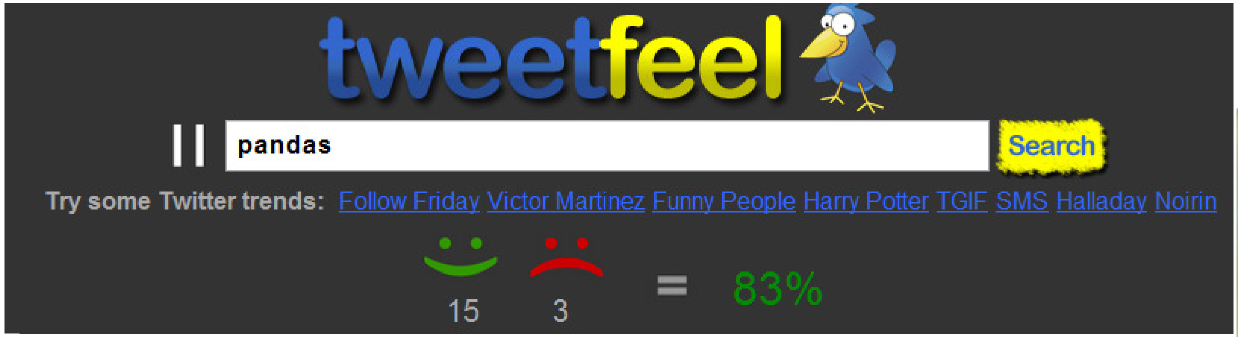
Robust Classification of Rare Queries Using Web Knowledge. Broder et. Al. SIGIR 2007

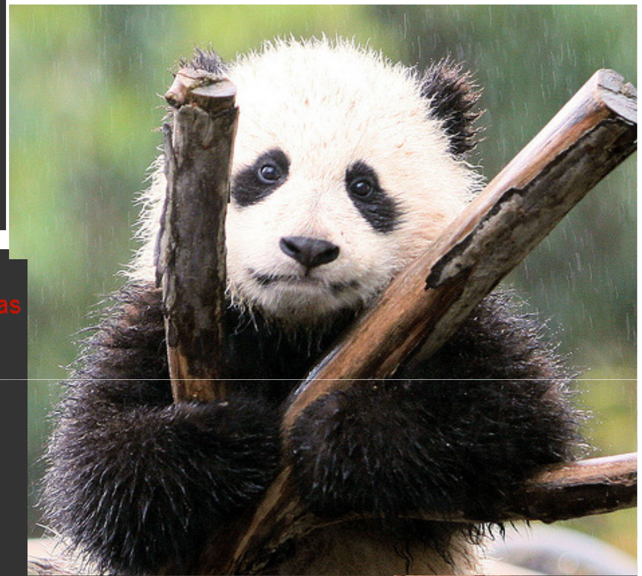
Non-topical categorization

- Classification by genre
  - patent, news article, meeting report, discussion, resume , tutorial, presentation, source code, blog post?
  - Not only words are features:
    - Average sentence length, layout structure (number of tables, lists), file format, classes of words (dates, times, phone numbers), sentence types (declarative, imperative, question), number of images, links…
- Classification by reading difficulty*
  - Compare definitions of **sugar:**
  - **Sugar** is something that is part of food or can be added to food. It gives a sweet taste © **simple.wikipedia.org/wiki/Sugar**
  - **Sugar** is a class of edible crystalline substances, mainly sucrose, lactose, and fructose. Human taste buds interpret its flavor as sweet © **wikipedia.org/wiki/Sugar**

***A Language Modeling Approach to Predicting Reading Difficulty.** Collins-Thompson et. al. 2004

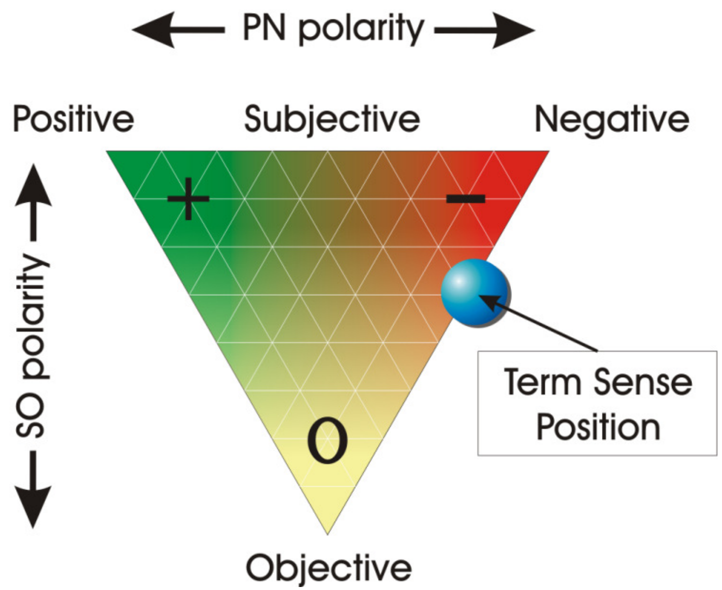
Categorization by sentiment (I)





Categorization by sentiment (II)

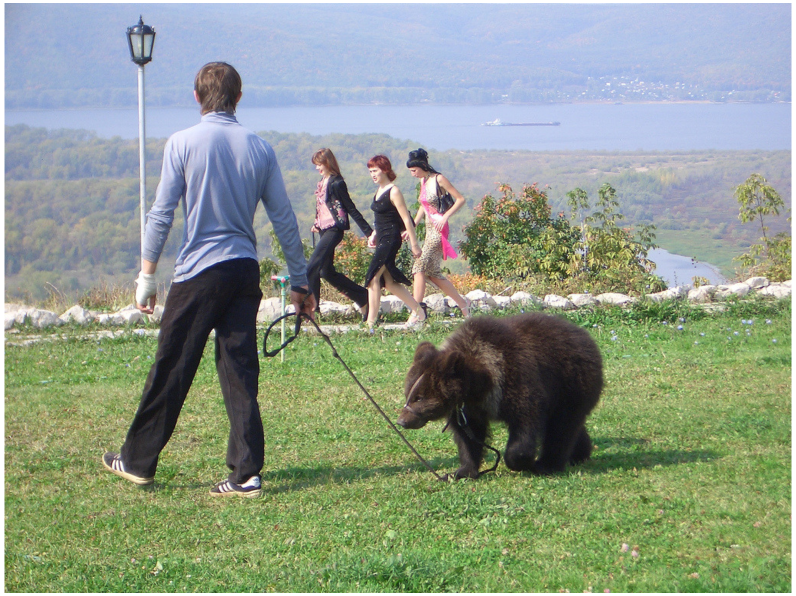
- Lexicon-based approaches:
  - Calculate ratio of negative/positive words/smileys
  - Weight contribution of every subjective term by its **inverse distance to query terms**
- Machine learning based approaches:
  - Build classification models for **texts** and **terms**:
    - Objective vs. Subjective
    - Positive vs. Negative
  - Better for each domain
  - Better use 2,3-grams
    - "long battery life"
    - "long execution time"



Categorization by location (I)

- Some documents, photos, videos, tweets…
  - are location agnostic and **some are not!**

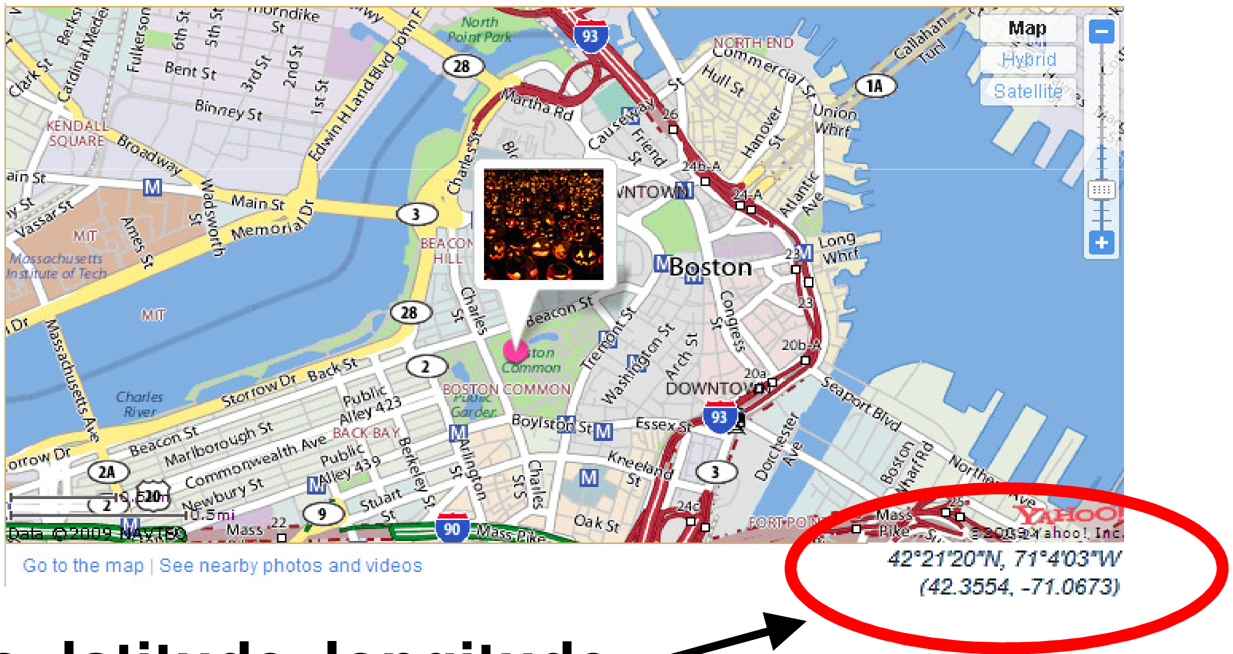
kitchen cats dogs



russia river brownbear

Categorization by location (II)

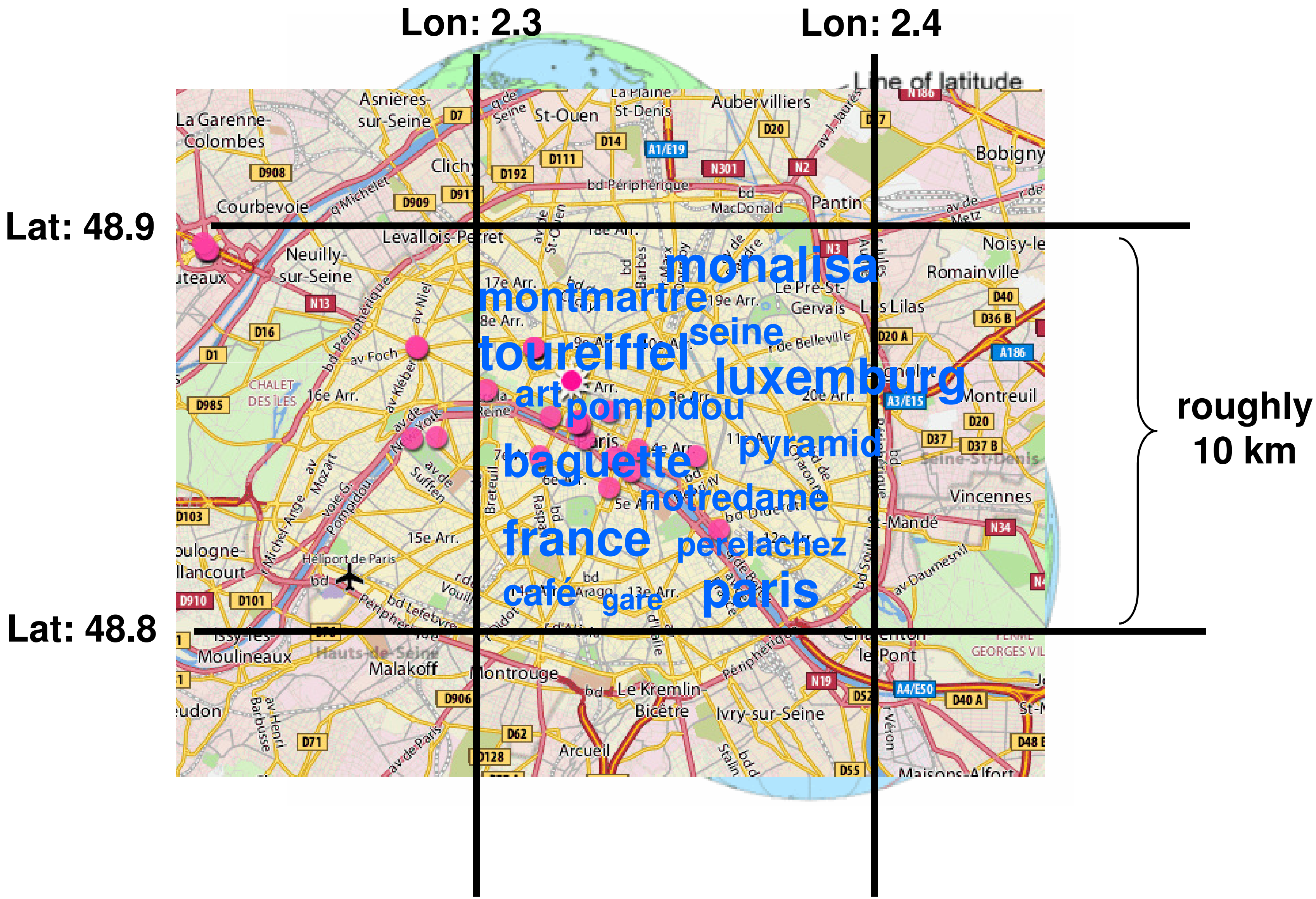
- **Some documents are geo-tagged**
  - There are more than 100 millions of them at Flickr!
  - Are we done?



geo-tags: latitude, longitude

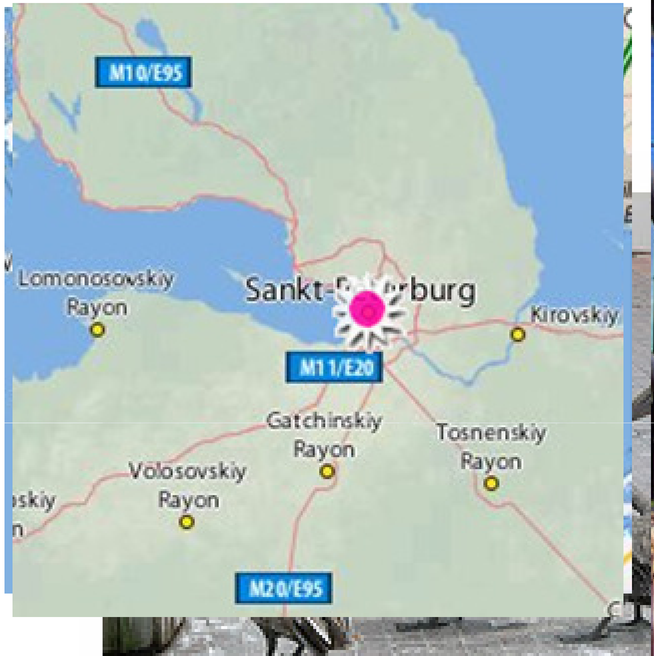
Around 96% of Flickr photos are not geo-tagged!

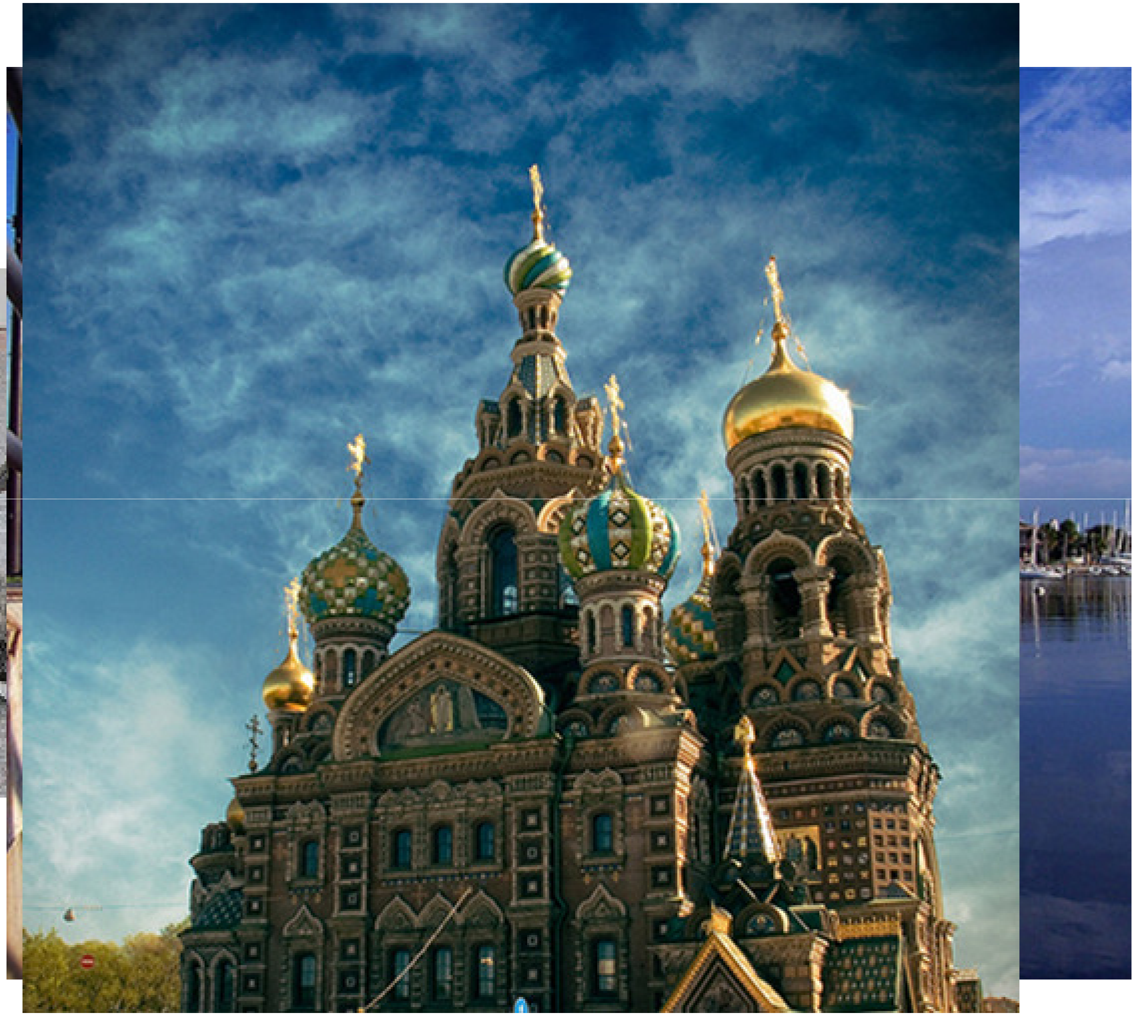
Categorization by location (III)



Categorization by location (IV)

St. Petersburg





Popular tags

russia, **church**, bridge, cathedral, light, **neva**, petersburg, water, **hermitage**, russian, **winter**, baltic

florida, **pier**, **sunrise**, tampa, st, tampabay, vinoypark, **pelican**, water, **warpedtour**, bird, petersburg, bay

Categorization by location (V)

- Locations – documents (L), tagsets – queries (T)
- Tags of photos are query terms (t_i)
- How likely that location L produced the image with a tagset T :

$$P(T \mid L) = \prod_{i=1}^{|T|} P(t_i \mid L)$$

$$P(t \mid L) = \frac{|L|}{|L|+\lambda} P(t \mid L)_{ML} + \frac{\lambda}{|L|+\lambda} P(t \mid G)_{ML}$$

- But there is much more we can do*:
  - Consider spatial ambiguity of tags?
  - Consider neighboring locations?
  - Consider that some of them are toponyms?
- Apply for place non-tagged photos? Not only photos?

***Placing Flickr Photos on a Map.**
Serdyukov P., Murdock V., van Zwol R. SIGIR 2009

Metadata extraction (I)

- Tags provide intuitive description
- Allow not only summarize, but aggregate
- Natural query terms suggestions
- Let's generate tags (***topic labels***)
  - For each document
  - For clusters of documents
  - For documents grouped by some (boring) facet
    - e.g. Year or Department
- Technically , we can build classification model for **each tag assigned to sufficient number of docs***
  - But let's do that in an unsupervised way

***Social Tag Prediction.** Heyman et. al. SIGIR 08

Metadata extraction (II)

- Plenty of ways to extract keyphrases…
  - What to consider? Several dimensions*…
- Relevance of phrase $l = w_1 w_2 w_3$ to document:

$$Score(l,D) = \alpha \frac{P(l \mid D)}{P(l \mid C)} + (1-\alpha) \sum_w \frac{P(w \mid D)}{P(w \mid C)}$$

- Relevance of document to phrase. **Minimize:**

$$Dist(l,D) = -\sum_w P(w \mid l) \frac{P(w \mid l)}{P(w \mid D)}$$

Over all docs where *l* occurs

- Uniqueness on document level. **Maximize:**

$$\max_{l' \in selected} Dist(l,l')$$

- Uniqueness on collection level. **Maximize:**

$$\frac{1}{|C|-1} \sum_{D' \neq D} Dist(l,D')$$

***Automatic Labeling of Multinomial Topic Models.** Mei et. al. KDD 2007

Metadata extraction (III)

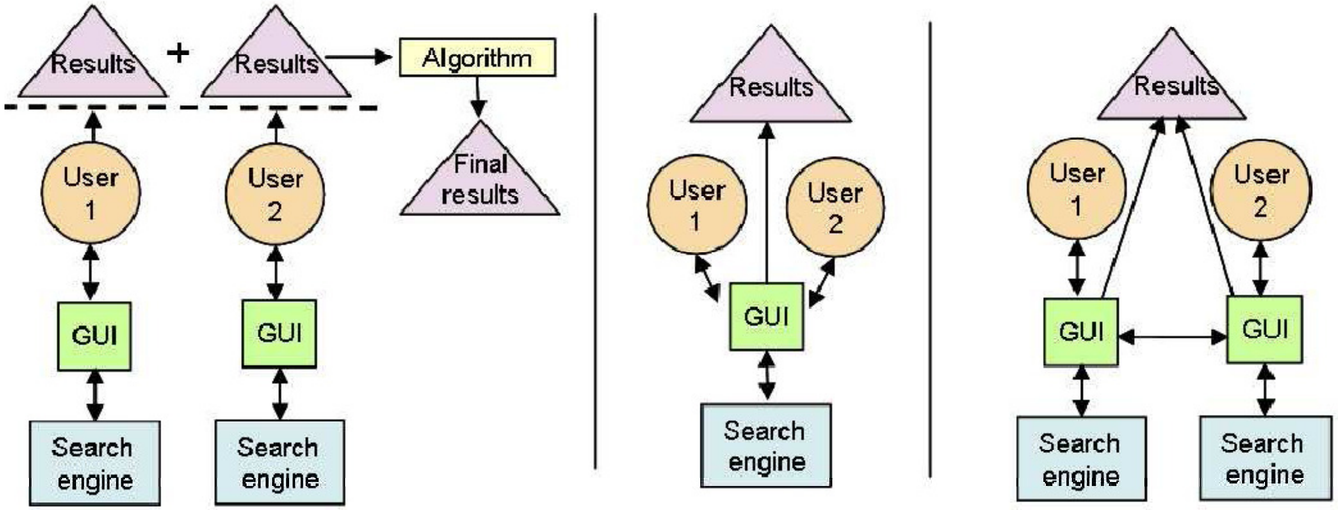
- So far not query-driven, right?
- Let’s move away from bag-of-words
- Possible algorithm:
  - Cluster sentences in a document
  - Select keywords for each cluster (as shown)
  - Find cluster(s) most relevant to a query
  - Represent document by keywords from relevant cluster(s)
- Just consider text windows around query terms

Summary

- No metadata?
- Categorize, categorize, categorize…
  - Semantic classes
  - Genres
  - Reading difficulty levels
  - Sentiments
  - Locations
  - **What else?**
- Or extract metadata from text to summarize!
  - Find tags, entities, etc...

What about the Future?

Collaborative exploratory search



- Collaborative search*:
  - Many queries, many people, one information goal
  - How to suggest and route queries?
  - How to route documents for evaluation?
  - How to aggregate opinions on documents?

*** Algorithmic mediation for collaborative exploratory search.** J. Pickens et. al. SIGIR 08

Aggregated exploratory search

- Find not only relevant facets/values, but…
- Find relevant domains (verticals) !

Query "hairspray"

- Present result sets from diffe
  in the order of their total rel

vertical	retrievable items
autos	car reviews, product descriptions
directory	web page directory nodes
finance	financial data and corporate inform
games	hosted online games
health	health-related articles
images	online images
jobs	job listings
local	business listings
maps	maps and directions
movies	movie show times
music	musician profiles
news	news articles
reference	encyclopedic entries
shopping	product reviews and listings
sports	sports articles, scores, and statistics
travel	travel and accommodation reviews
tv	television listings
video	online videos

References: Exploratory search

- http://en.wikipedia.org/wiki/Exploratory_search
- http://en.wikipedia.org/wiki/Faceted_search
- **Exploratory search: Beyond the Query-Response Paradigm.**
  R. White and R. Roth. 2009
- **Faceted search**. D. Tunkelang. 2009
- **Search User Interfaces.** M. Hearst. 2009.
  free at: http://searchuserinterfaces.com/
- **Opinion Mining and Sentiment Analysis.** B. Pang and L. Lee. 2008
  free at: http://www.cs.cornell.edu/home/llee/
- **A Survey on Automatic Text Summarization.** D. Das, A. Martins. 2007
  free at: http://www.cs.cmu.edu/~afm/
- **Conferences:** SIGIR, ECIR, WWW, WSDM, KDD, HCIR

References: advanced exploratory search

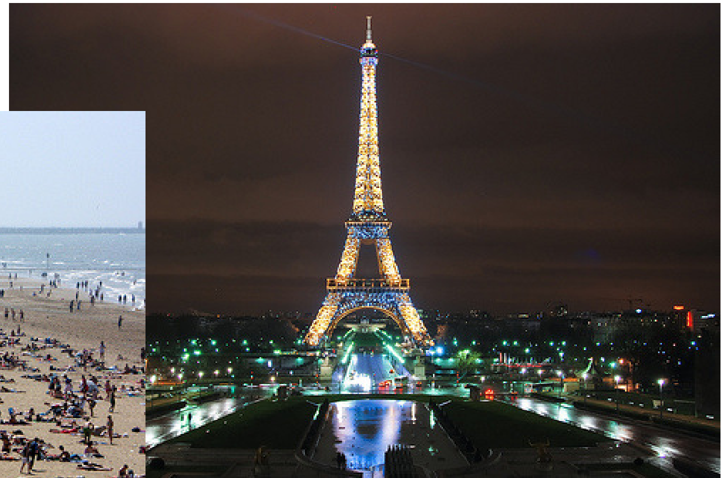
- Collaborative search:
  - http://en.wikipedia.org/wiki/Collaborative_search_engine
  - **Algorithmic mediation for collaborative exploratory search**. J. Pickens et. al. SIGIR 2008
  - **Discovering and Using Groups to Improve Personalized Search.** J. Teevan. WSDM 2009
  - Download and play: http://research.microsoft.com/en-us/um/redmond/projects/searchtogether/
- Aggregated search:
  - **Integration of News Content into Web Results.** F. Diaz. WSDM 2009. (Best paper award)
  - **Sources of evidence for vertical selection.** J. Arguello et. al. SIGIR 2009. (Best paper award)

- 4-years ... ft
- EU-pro...
  - Sumn... ntity ranking, social search

60 mins

40 mins

3.5 hours



Enterprise and Desktop Search

Lecture 4: Expert finding

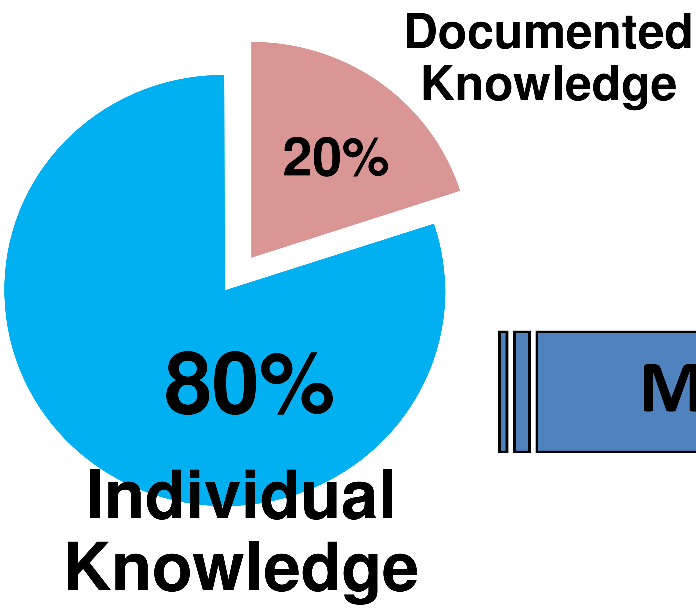
Pavel Dmitriev, Pavel Serdyukov, Sergey Chernov

Outline

- The need for expert finding
- State-of-the-art approaches
- Advanced techniques:
  - Mining for personal language models
  - Proximity-aware expert finding
  - Looking for additional evidence in the enterprise
  - Looking for additional evidence on the Web
- Future challenges

Search for experts

- Some knowledge is not easy to find
  - Not stored in documents
  - Not stored in databases
  - **It is stored in peoples' minds!**

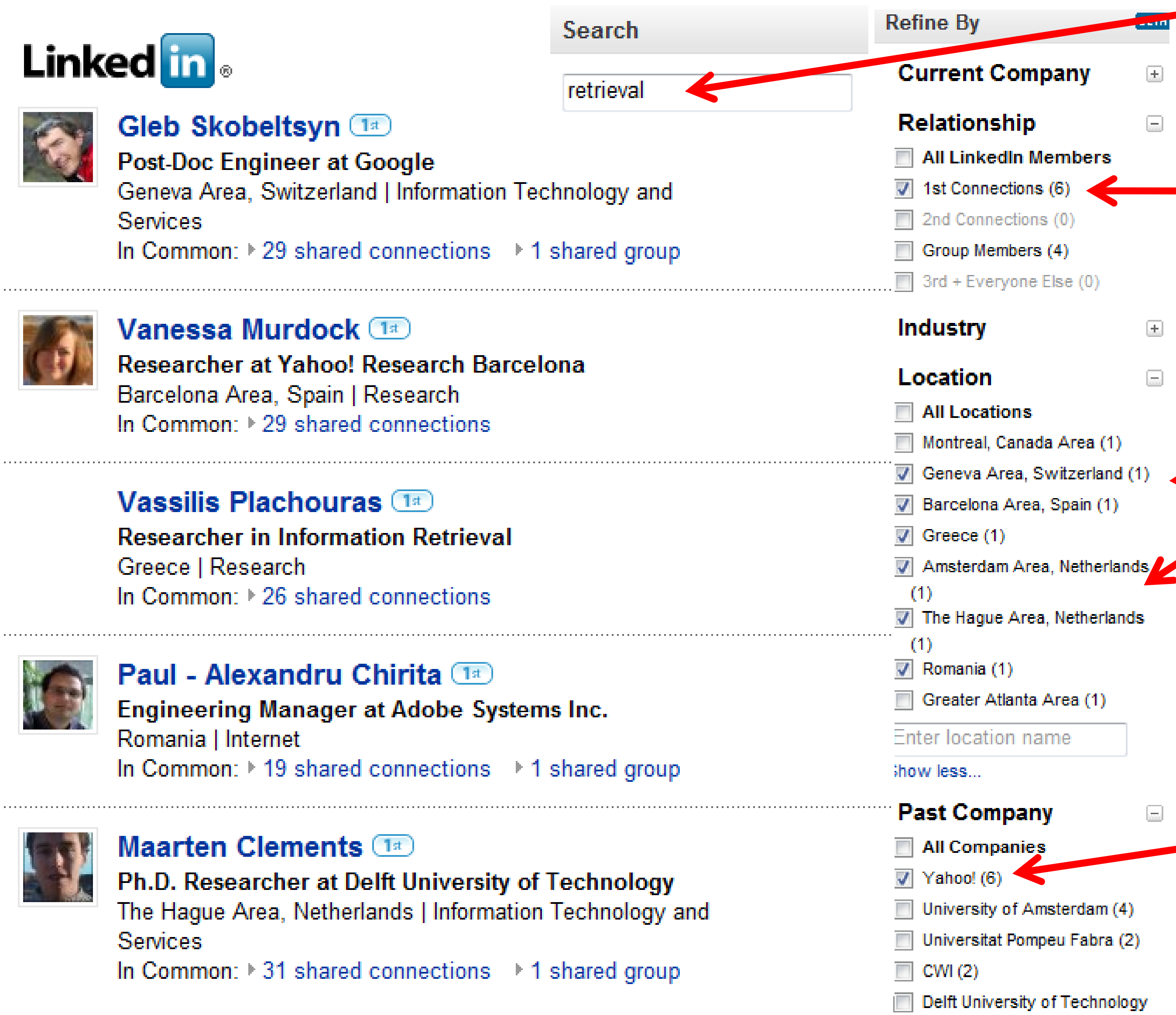


Meet People!

Search for experts

- Let's search for ~~documents~~ people
- Who is ~~relevant~~ expert on topic X?
- Basically, a special case of **faceted search**
  - Facets "people", "employees"
- Try some expert search right now:

Search in personal profiles



Search for experts in **retrieval**

Search only among known people

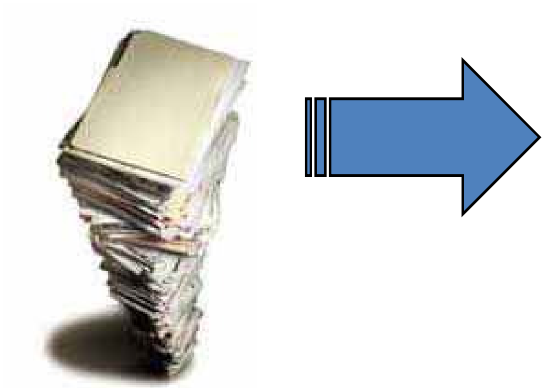
Working in Europe

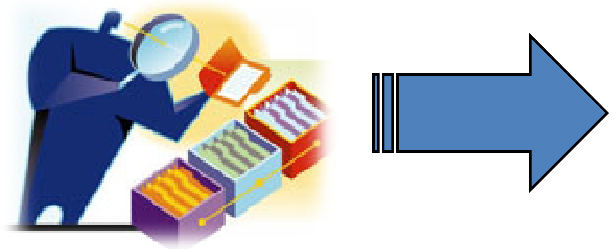
Ever worked at Yahoo!

Faceted search for experts!

Expert finding via document analysis

- Analyze self-made profiles?
  - Need some enthusiasm to maintain
  - Subjective due to over/under-estimation
- Sleuth for expertise evidence in existing documents…

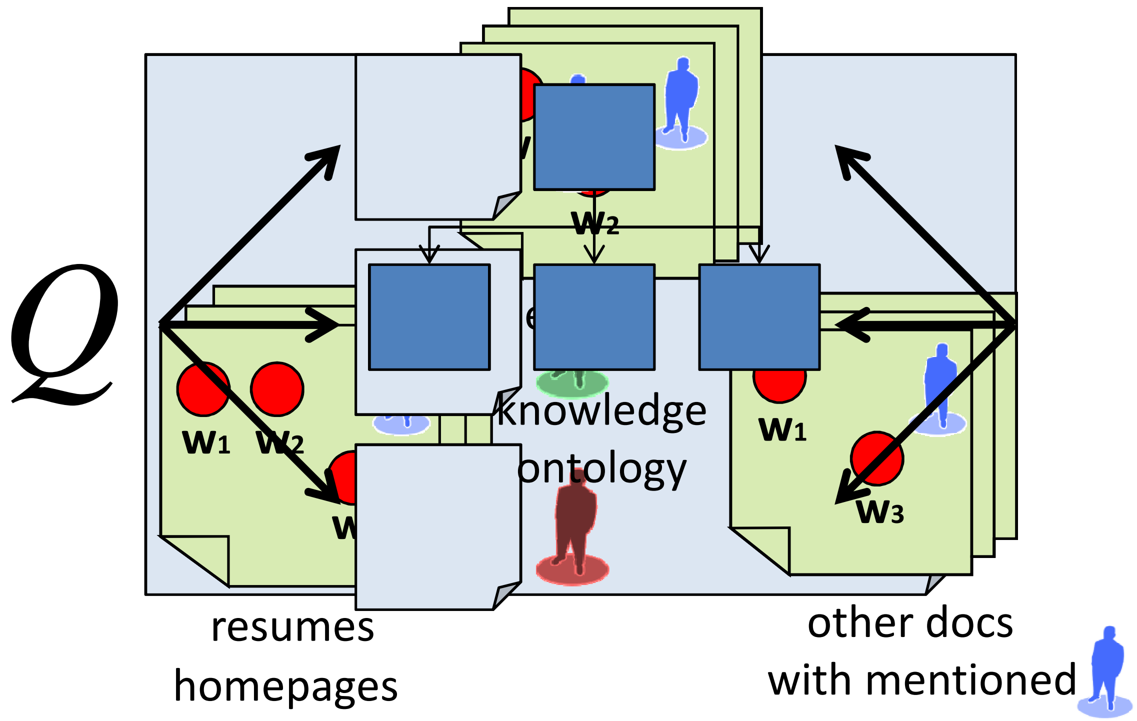




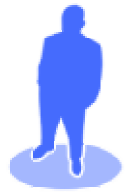
Documents **Expert Finder** **Experts**

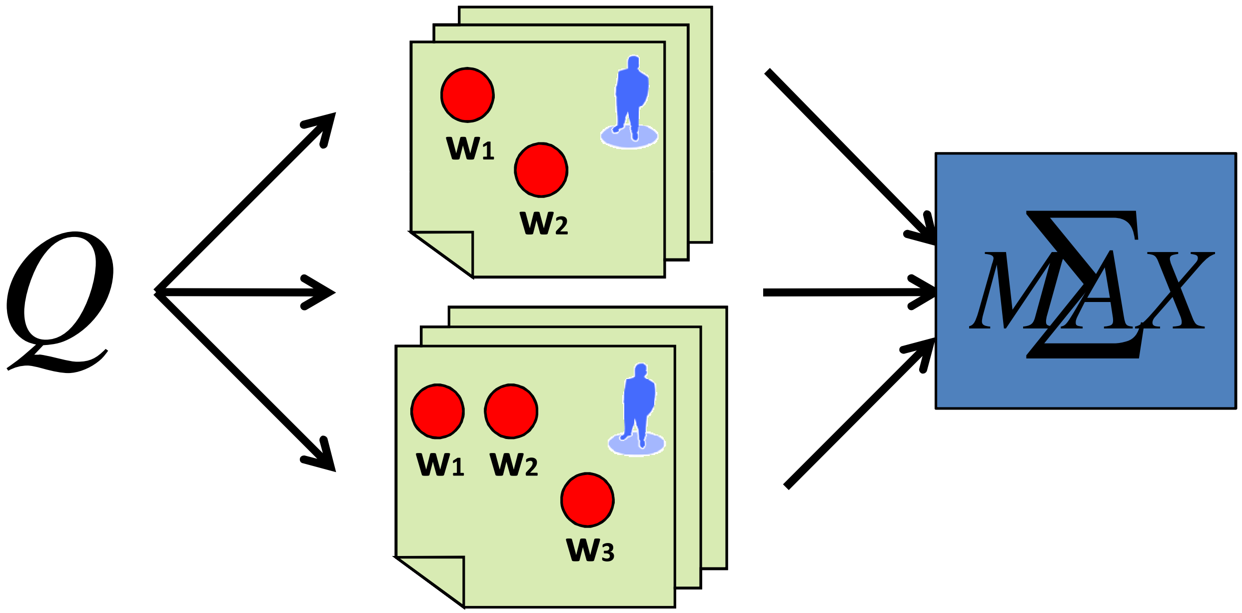
Profile-based expert finding

- 1st step: Build a personal profile for
- 2nd step: Match it to a query as a document



Document-centric expert finding

- 1st step: Rank all documents with 
- 2nd step: Aggregate document scores



- Remember facet values ranking?

Popular datasets

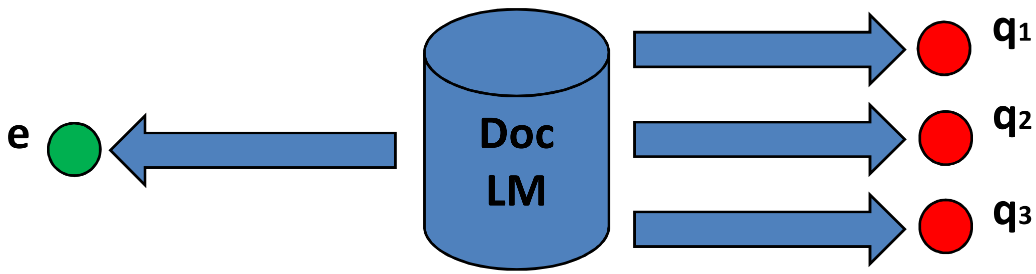
- **TREC 2005-2006: W3C data**
  - **The largest part consists of mailing lists**
  - **About 1000 candidates provided**
  - **Judgments made by participants (50 queries)**
  - **Really many "experts" per query**
- **TREC 2007-2008: CSIRO data**
  - **www.csiro.au crawl**
  - **About 3500 candidates (just all persons mentioned)**
  - **Judgments made by the organization itself (49 queries)**
  - **Very few "experts" (key persons) per query**
- **Three measures are analyzed**
  - **MAP (Mean Average Precision) and P@5**
  - **MRR (Mean Reciprocal Rank)**

Going beyond bag-of-words (I)

- **Popular Intuition:**

Expertise is proportional to the degree
of query terms and the person's
co-occurrence

- **Classic document-centric approach*:**

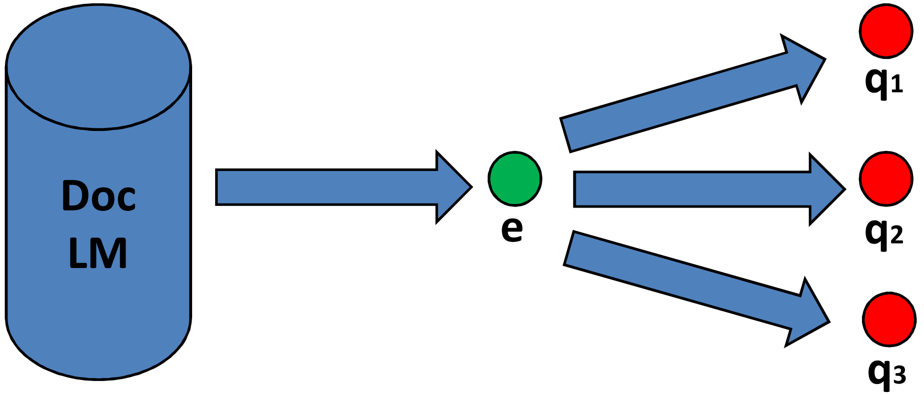


$$P(e,Q)=\sum_D P(e,Q\mid D)P(D)=\sum_D P(e\mid D)\underbrace{P(Q\mid D)P(D)}_{\approx P(Relevance\mid D)}$$

***A language modeling framework for expert finding.** Balog et. al. SIGIR 06

Going beyond bag-of-words (II)

- Full Independence is not realistic
- Persons are responsible for terms!

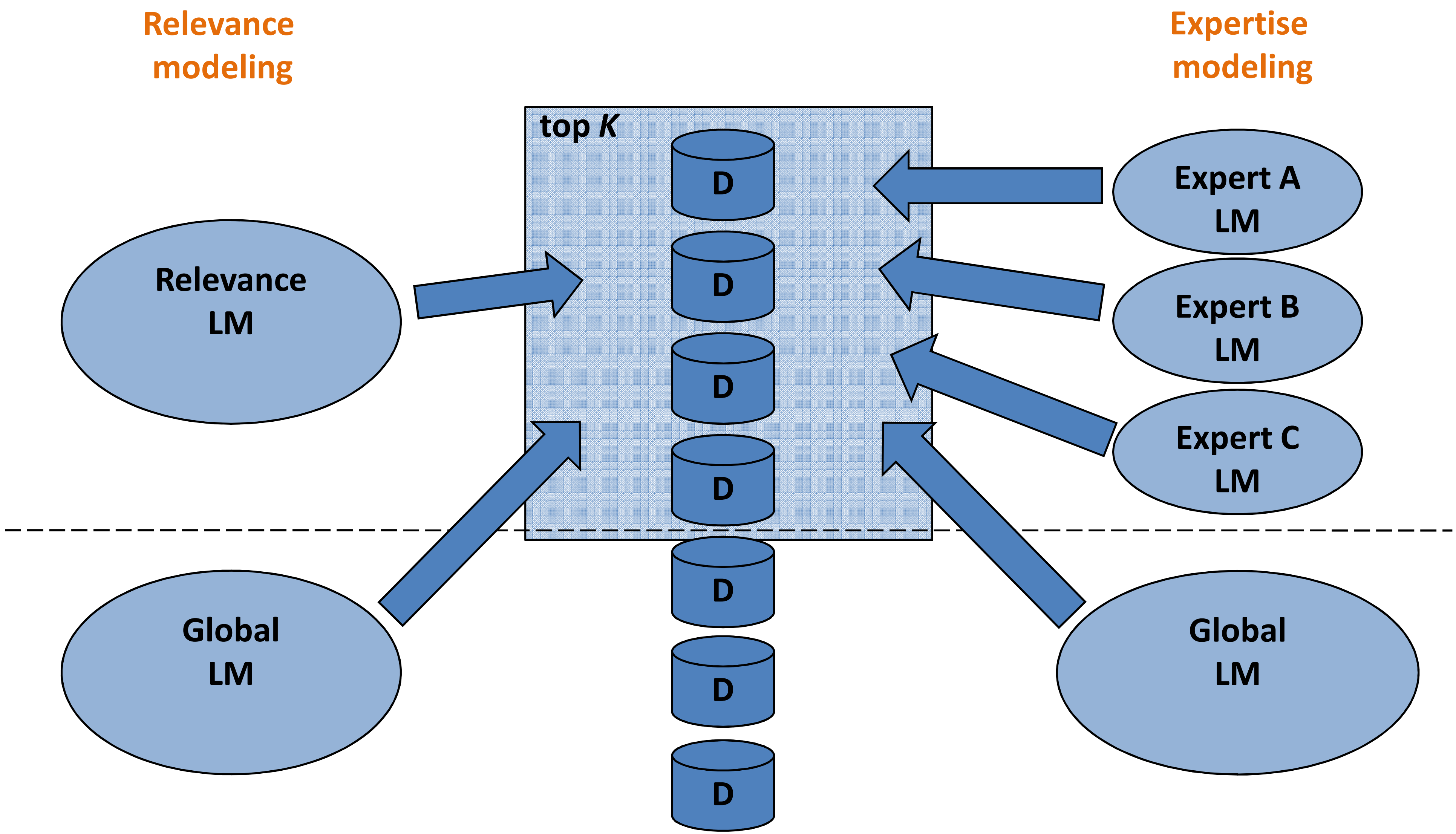


?

$$\sum_{D} P(e, Q \mid D) P(D) = \sum_{D} P(Q \mid e) P(e \mid D) P(D)$$

Modeling documents as mixtures of persons for expert finding. Serdyukov and Hiemstra. ECIR 2008

Mining personal language models (I)



Mining personal language models (II)

- Likelihood of Top *K* retrieved documents

$$\prod_{D}\prod_{w\in D}((1-\lambda_G)(\sum_{i=1}^{m} P(w \mid e_i)P(e_i \mid D)) + \lambda_G P(w \mid G))^{c(w,D)}$$

unknown params

  - $c(w,D)$ - count of terms *w* in document *D*
  - λ_G - probability of term generation from the Global LM

- $P(e \mid D)$? Previously, was inferred from:
  - Importance of a document's field
  - Number of candidates in a document

Mining personal language models (III)

- Steps for EM iterations:

E – step :

$$P(e \mid w, D) = \frac{(1-\lambda_G)P(e \mid D)P(w \mid e)}{(1-\lambda_G)(\sum_{i=1}^{m} P(e_i \mid D)P(w \mid e_i)) + \lambda_G P(w \mid G)}$$

M – step :

$$P(w \mid e) = \frac{\sum_{D \in TopK} c(w,D)P(e \mid w,D)}{\sum_{w} \sum_{D \in TopK} c(w,D)P(e \mid w,D)}$$

$$P(e \mid D) = \frac{1 + \sum_{w \in D} c(w,D)P(e \mid w,D)}{m + \sum_{i=1}^{m} \sum_{w \in D} c(w,D)P(e_i \mid w,D)}$$

unfixed

Going beyond "personal" documents

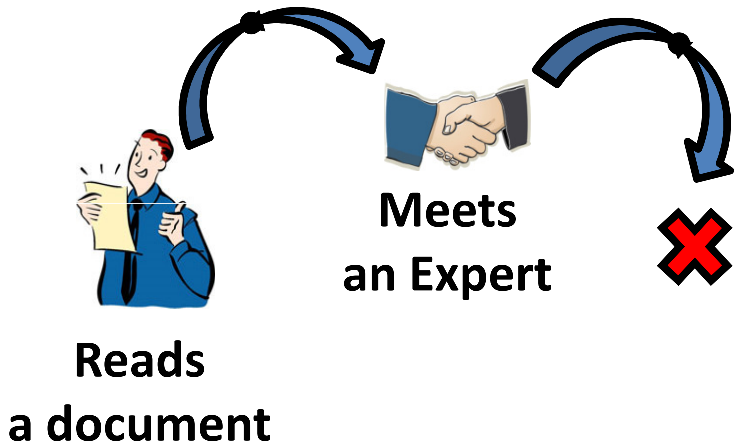
- Look at the classic approach again:

$$Expertise(e) = \sum_{D \in TopK} P(e \mid D) P(Q \mid D) P(D)$$

1. **User selects a document from the top**

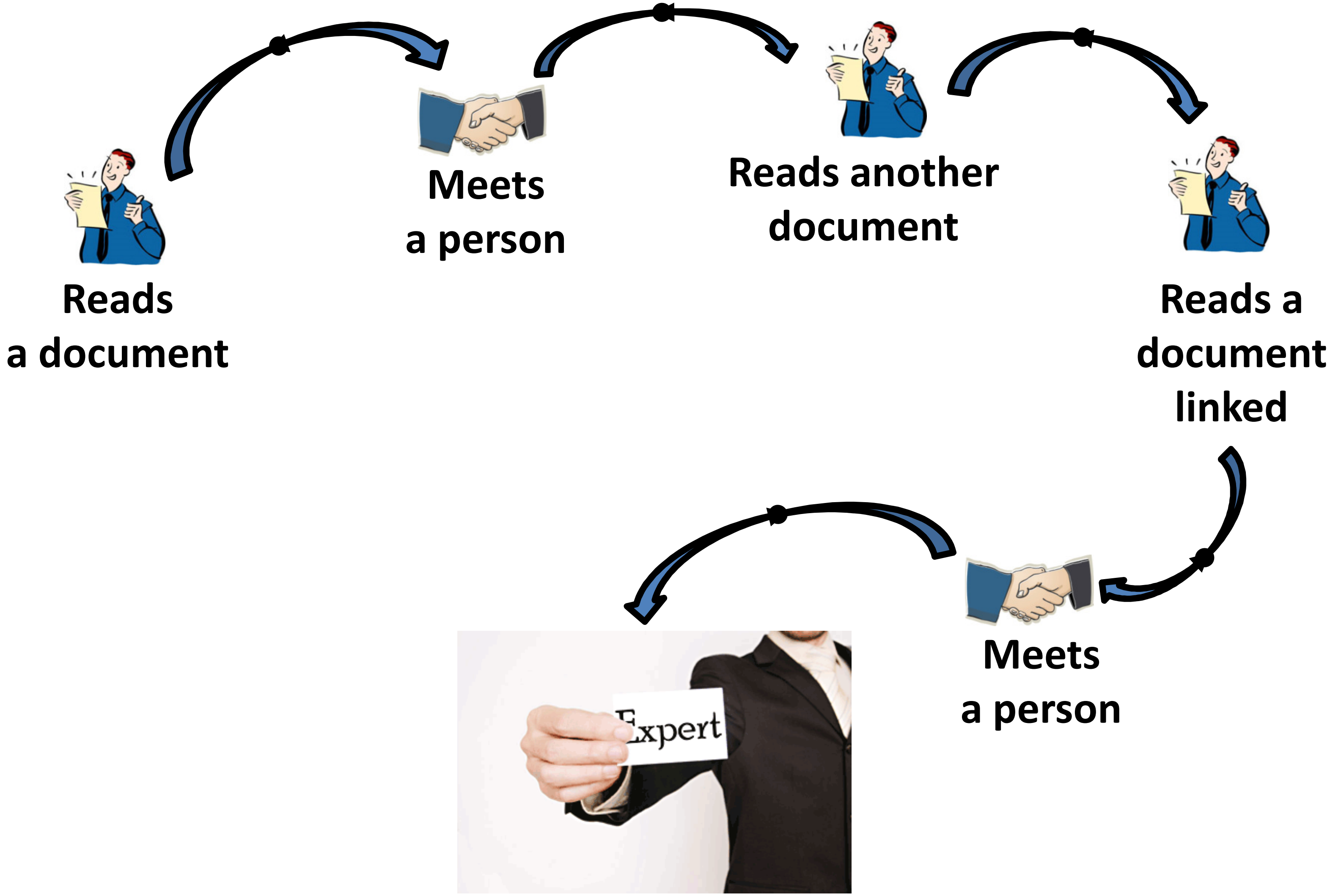
2. User selects a person from the document

3. Finished? Well, not in exploratory mood

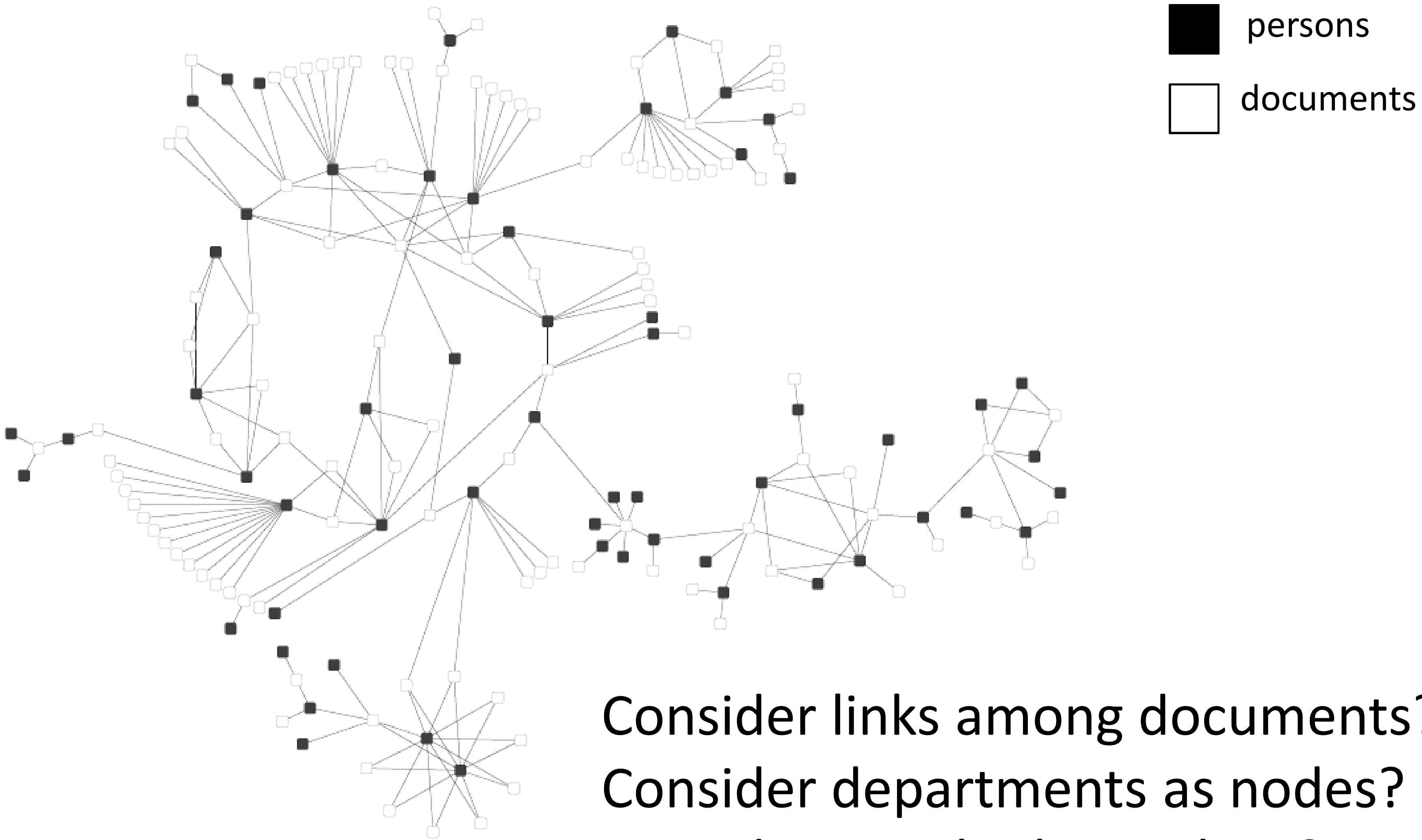


- Expertise evidence is never propagated further than to **mentioned persons**

Exploratory search for experts



Expertise graph



Consider links among documents?
Consider departments as nodes?
Consider social relationships?

Multi-step relevance propagation

- How to model this walk for expertise?
  - Although, considering that experts should be close to relevant documents
- How to propagate expertise evidence (relevance) further after the first step?
- **Answer:** Multi-step relevance propagation with random walk models
  - Finite-random walk (FRW)
  - Infinite random walk (IRW)
  - Absorbing random walk (ARW)

In **P. Serdyukov, H. Rode, and D. Hiemstra.** Modeling Multi-step Relevance Propagation for Expert Finding. In **CIKM 2008.**

Finite random walk

- Model the user as a lazy seeker:
  - So, who is the most probable expert to end up with after some ***K number of steps***?
- How to model laziness in a smart way?

$$P_0(D) = P(Q \mid D), P_0(e) = 0$$

$$P_i(D) = \underbrace{P(Q \mid D)}_{\text{Prob. to stay at } D} P_{i-1}(D) + \sum_{e \to D} P(D \mid e) P_{i-1}(e),$$

$$P_i(e) = \sum_{D \to e} \underbrace{(1 - P(Q \mid D))}_{\text{Prob. to move on from } D} P(e \mid D) P_{i-1}(D)$$

$$Expertise(e) = P_K(e)$$

Infinite random walk

- Model the user as a tireless seeker:
  - So, who is the most probable expert to end up with after ***infinite number of steps***?
- How to model tirelessness smartly?

$$P_i(e) = \sum_{D \to e} P(e \mid D) P_{i-1}(D)$$

$$P_i(D) = \underbrace{\lambda P(Q \mid D)}_{\text{Prob. of walk restart from } D} + (1-\lambda) \sum_{e \to D} P(D \mid e) P_{i-1}(e),$$

$$Expertise(e) = P_\infty(e)$$

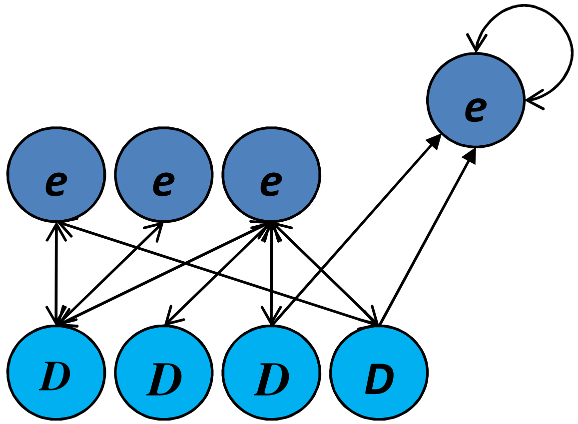
Absorbing random walk

- Absorbing walk:

$$P_0(D) = P(Q \mid D), P_0(e) = 0$$

$$P_i(D) = \sum_{e \to D} P(D \mid e) P_{i-1}(e),$$

$$P_i(e) = \sum_{D \to e} P(e \mid D) P_{i-1}(D) + P_{i-1}(e) P^{self}(e \mid e)$$



- What is the **generalization** of the classic one-step propagation:

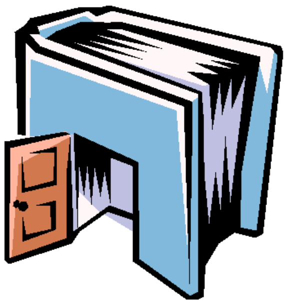
$$Expertise(e) = \sum_{D \in TopK} P^{mult}(e \mid D) P(Q \mid D) P(D)$$

Prob. to reach e from D in **minimum number of steps**

In **P. Serdyukov, H. Rode, and D. Hiemstra.** Modeling Expert Finding as An Absorbing Random Walk. In **SIGIR 2008.**

Looking for better expertise evidence

- So far considered:
  - **Documents are black boxes (black bags of words)**
  - **There is no world outside the enterprise**
- Can we do better? Look at two extremes…
- Go deeper into the document on a word-level



- Escape the enterprise…. in search for better evidence

Proximity-aware expert finding (I)

- Remember document-centric model?

$$P(e,Q) = \sum_D P(e \mid D) P(Q \mid D) P(D)$$

- Why consider independence?

$$P(Q \mid D) \Rightarrow P(Q \mid e, D) = \prod_{q \in Q} P(q \mid e, D)$$

For every occurrence of a query term and an expert mention

Proximity function

$$P(q \mid e, D) = \frac{\sum_{q \in D}^{count(q,D)} \sum_{e \in D}^{count(e,D)} k(q,e)}{Z}$$

Normalization constant

Proximity-aware expert finding (II)

- Linear function:

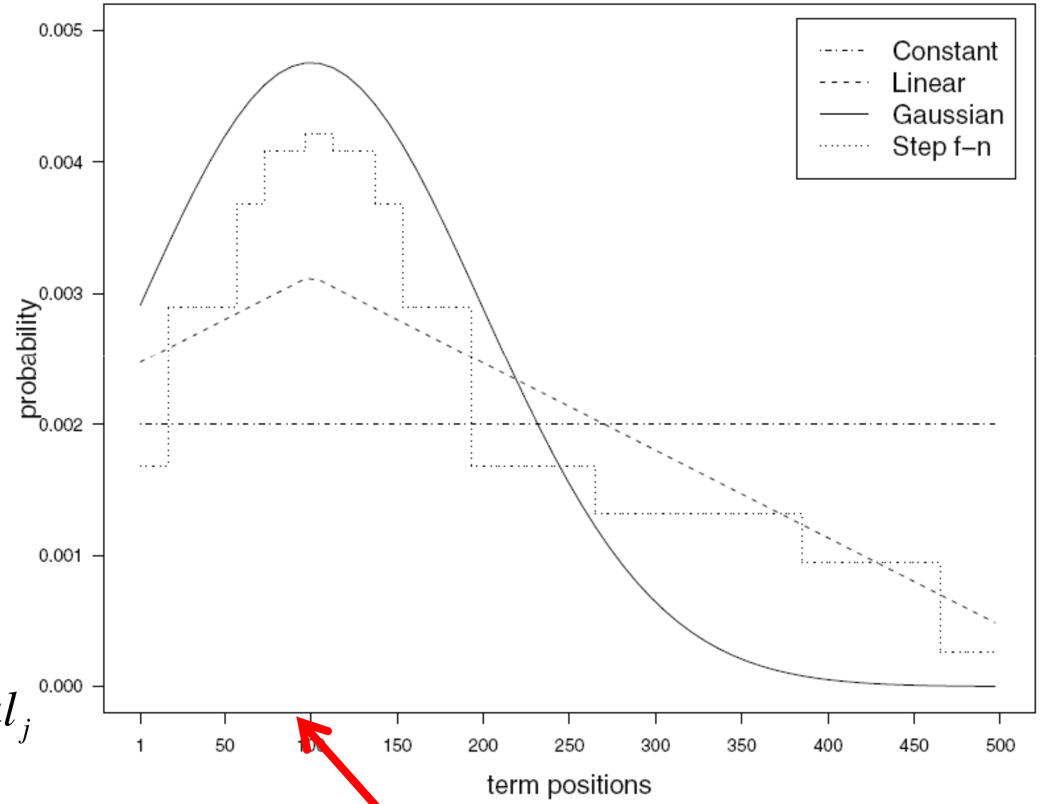
$$k(q,e) = 1 - |pos(q) - pos(e)|$$

- Gaussian function:

$$k(q,e) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[\frac{-(pos(q) - pos(e))^2}{2\sigma^2}\right]$$

- Step function:

$$k(q,e) = \alpha_j, if\ |pos(q) - pos(e)| \in Interval_j$$



Position of expert mention

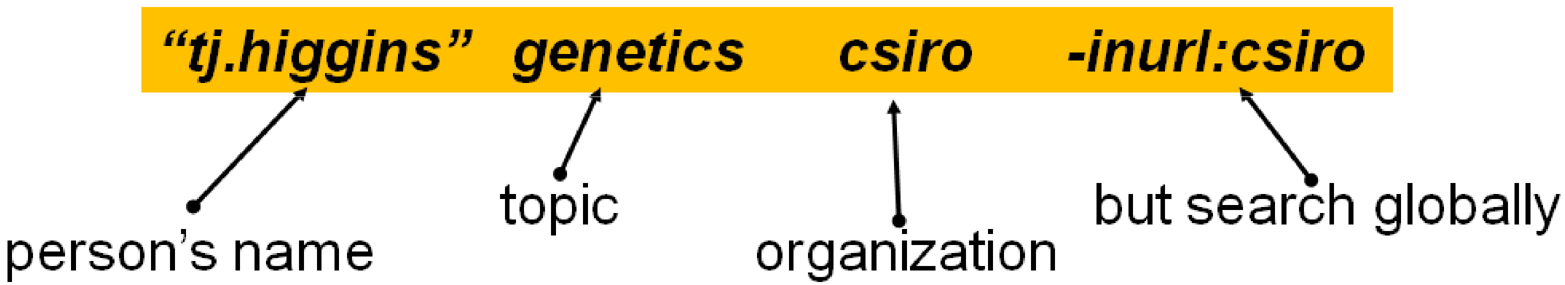
Proximity-based document representation for named entity retrieval. Petkova et. al. CIKM 2007

Going beyond the enterprise

- Why to search only in the enterprise?

Acquiring data via Search APIs

- Retrieve all pages with person name?
  - But APIs return at most 1000 results
- Build a query consisting of:



- The number of returned results is a **rough estimate of expertise**

In **P. Serdyukov and D. Hiemstra**. Being Omnipresent to Be Almighty: The Importance of the Global Web Evidence for Organizational Expert Finding. In **FCHER 2008 (SIGIR 2008 Workshop).**

Where to start?

- Issue 3500 queries to APIs for each topic?
  - Takes about 30 minutes
- Some pre-selection stage for candidates?
  - Experts should be within some **Top-*K***
- We are making Enterprise run anyway
  - And it is very fast
- We have full access to the Enterprise data
  - It should be the primary evidence

Web Search evidence

- We need precise estimates for the number of results:
  - Estimates of “total results” are very imprecise
  - Their precision depends on starting position

1st Yahoo! page: 1 - 10 of 273 for **genetic modification" tj. higgins" csiro- inurl: csiro**

Last Yahoo! page: 71 - 73 of 111 for **genetic modification" tj. higgins" csiro- inurl: csiro**

- Worst estimate
- Better estimate
- The best estimate

- Google API returns only 32 search items
  - And its estimates are less reliable

Results **21** - **30** of about **1,020** for **genetic modification "tj.higgins" csiro -inurl:csiro**

News evidence

- Good experts are often news-makers
  - Make discoveries
  - Receive awards
- Every engine has a News Search API !
  - But all of them allow to search only in the news from the past month
  - Google News Archives allows to search even in 19th century news, but has no API
- But, let's simulate it
  - By adding ***inurl:news*** clause

Blog evidence

- Blogs are knowledge marketplaces
- Even most corporate blogs are public
- Quoting is a social recommendation

Kevin Rose writes that Digg is launching a recommendation engine
that "uses your past digging activity to identify what we call Diggers

Amit Singhal, the head of the Core Ranking team at Google has a post
on Google's philosophy of ranking.

John Langford just posted a list of seven ICML '08 papers that he
found interesting. I appreciate his taste in papers, and I particularly

- Two blog search engines have the best coverage:
  - Technorati API: almost not supported
  - Google Blog Search API: returns only 8 results

Academic search evidence

- Strong academic record is a must
  - Especially for R&D companies
- Big academic search engines have no API
  - Live Search Academic
  - Google Scholar (recommends experts itself!)

Results **1** - **2** of about **528,000** for **web retrieval**.

Key authors: G Salton - D Hawking - N Craswell - P Bailey - W Grosky

- But Google Book Search API is available!
  - It’s like a crippled Google Scholar with only books indexed

Combining evidences

- Why we need so many sources?
- Good expert is not only a local winner
  - Should be “omnipresent”
- Normalization of absolute values is hard
  - Vary a lot over queries and search engines
- Rank aggregation is a convenient solution

$$Expertise(e) = \sum_{Rankings} -Rank(e)$$

Considering URL quality

- What about result set quality?
  - Considering only its **size** is too naive
- We should measure the quality of each result item(*URL, Title, Summary*):

$$Expertise(e) = \sum_{Item \in WebResultSet} Quality(Item)$$

- Two types of quality measures:
  - **Query-independent**
  - **Query-dependent**

Future challenges for expert finding

- Modeling dependencies within a document
  - More complex topic models?
- Relevance propagation
  - Introduce new entities? Relevance sources? Search for organizational units?
- Utilize more web sources

References: expert finding

- Expert finding in industry:
  - **Expert finding systems. Survey.** M. Maybury. 2006

  free at: http://www.mitre.org/work/tech_papers/tech_papers_06/
- Expert finding in academia:
  - **People Search in the Enterprise.**

    K. Balog. PhD Thesis. 2008
  - **The Voting Model for People Search**.

    C. Macdonald. PhD Thesis . 2009
  - **Search for expertise: going beyond direct evidence.**

    P. Serdyukov. PhD Thesis. 2009