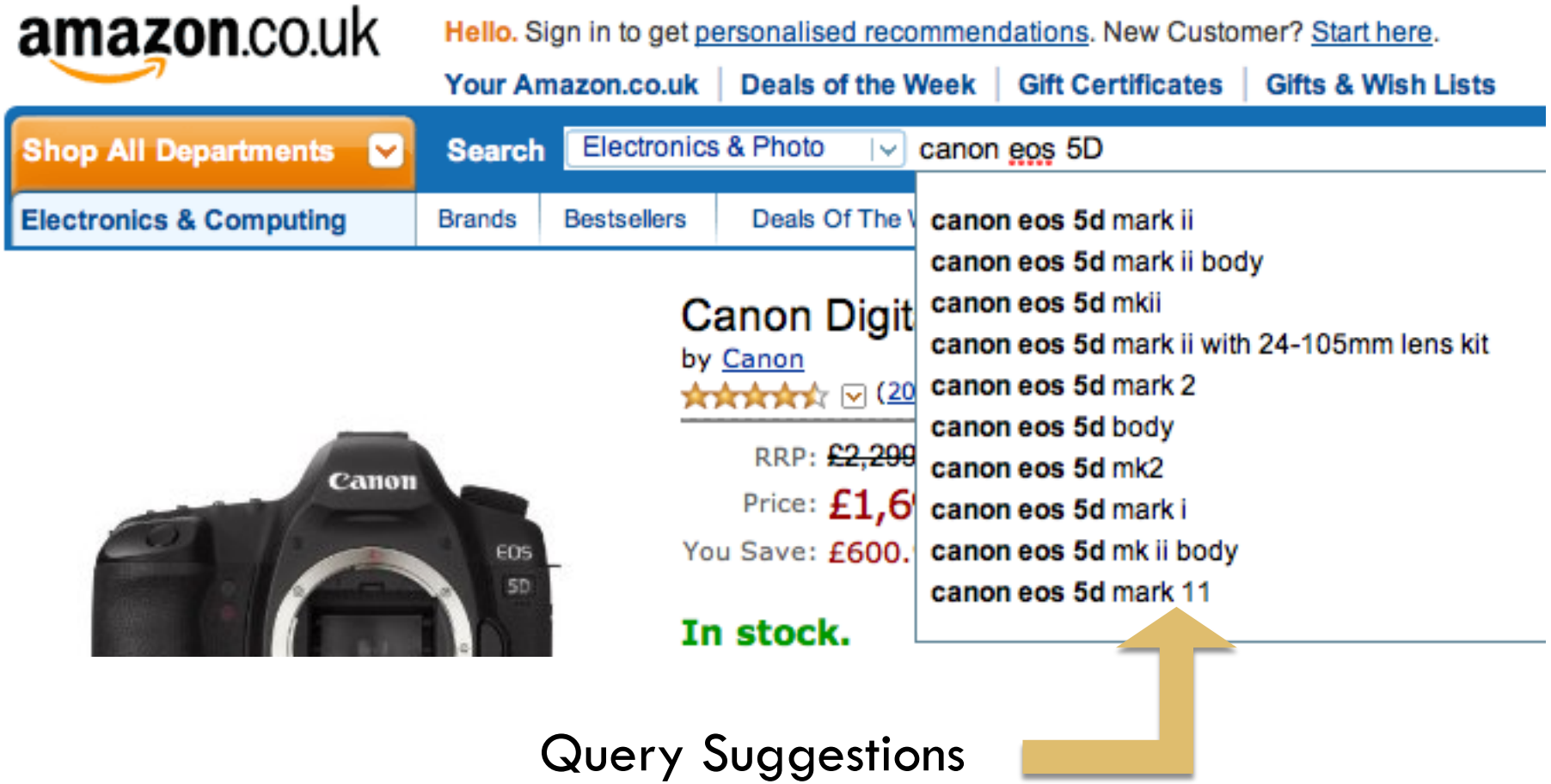
E-COMMERCE: QUERY RECOMMENDATION (AND EXPANSION)

Claudio Silvestri

Intro to the problem



Query Suggestions
(related to query expansion)

Algo 1

Concept-Based Query Expansion

- Idea:
  - Expand a query by adding similar terms
  - Could be done by counting co-occurrences
- Nice idea:
  - Compare postings-lists to compute term similarity

Concept-Based Query Expansion

- A term t_i is modeled as a vector:
  - $t_i = (d_{i1}, d_{i2}, \ldots, d_{in})$
  - Where d_{ik} signifies the importance of document k for term i

$$d_{ik} = \frac{\left(0.5 + 0.5\frac{\text{ff}(d_k, t_i)}{\text{maxff}(t_i)}\right) \cdot \text{iif}(d_k)}{\sqrt{\sum_j (d_{ij})^2}}$$

  - $ff(d_i,t_i)$ = # time occurs t_i in d_k
  - $maxff(d_i,t_i)$ = max of $ff(d_k,t_i)$ for every d_k
  - $iff(d_k)$ = log (# terms / # of distinct terms in d_k)

Concept-Based Query Expansion

- Similarity between two terms is defined as:

$$\text{SIM}(t_i, t_j) = t_i \cdot t_j = \sum_k d_{ik} \cdot d_{jk}$$

- Wrap up:
  - A term is a set of meanings (= documents)
  - Each meaning (=document) has different weight
  - Two terms are similar if they are represented by the same documents/meaning/concepts

Concept-Based Query Expansion

- What about the query ??
- A query ***q*** is modeled as a vector:
  - $q = (q_1, q_2, \ldots, q_m)$
  - where q_i is the importance of term *i* for the query *q*
  - could be *1/m*, or *idf*, or explicit, or others …
- Move into the *document space*:

$$\text{SIM-QT}(q,t) = \frac{\sum_{t_i \in q} q_i \cdot \text{SIM}(t_i,t)}{\sum q_j}$$

Concept-Based Query Expansion

- This allows to compute a score for every term *t*
- Get the ***r*** top-ranked terms
- Create a new expanded query in the document space
  -
  -

Concept-Based Query Expansion

- Results on three *corpora*:

Collection	MED	CACM	NPL
avg. precision of original queries	0.5446	0.2718	0.1818
Number of additional terms	80	100	800
avg. precision of expanded queries	0.6443	0.3339	0.2349
Improvement	+ 18.31 %	+ 22.85 %	+ 29.21 %

How to apply this to Query Suggestion

- We have seen an algorithm for query expansion
- It provides a ranked list of relevant terms t_1, …, t_n
- Given a query q, we can suggest new queries like:
  - $q + t_1$
  - $q + t_2$
  - …
  - $q + t_n$

Algo 2

An Association Thesaurus for Information Retrieval

- Two main assumptions:
  - Terms that co-occur frequently and close to each other are likely to be related
  - Every term can be tagged as being:
    - N: Noun, J: Adjective, R: Adverb, V: Verb, I: Cardinal number, A: Article, others…
- Goal:
  - Provide query suggestions according to a given template or *phrase rule*:
    - {AN} or {NNN, JNN, JJN, NN, JN, N}
  - NB: query suggestions was originally used to run an expanded query

An Association Thesaurus for Information Retrieval

- A *document* is a set of paragraphs
- A *paragraph* is a set of sentences
- A *sentence* is a set of phrases
- A *phrase* is a set of consecutive *terms*, according to the given *phrase rule*

- Goal:
  - Find terms that co-occur frequently in phrases

An Association Thesaurus for Information Retrieval

- Given a text collection, *for each phrase and term* in a *paragraph* generate an association:
  - <term_id, phrase_id>
- Count occurrences over the whole collection to compute triples:
  - <term_id, phrase_id, frequency>
- All the triples regarding a given *phrase_id* are gathered to form *pseudo-document*:
  - $term\_id^1$ ($frequency^1$), …, $term\_id^N$ ($frequency^N$).

An Association Thesaurus for Information Retrieval

- The pseudo-documents are indexed by some information retrieval system
- A query *q* is submitted to such system, which will return a set of pseudo-documents
  - Each pseudo-document corresponds to a phrase
  - The top ranked phrases are returned to the user

An Association Thesaurus for Information Retrieval

- Results:

```
Query:115.1 : Impact of the 1986 Immigration Law  - will report
specific consequence consequences of the U.S.'s Immigration Reform
and Control Act of 1986.

    0.511462        illegal immigration
    0.501936        illegals
    0.499120        undocumented aliens
    0.498964        amnesty program
    0.498054        immigration reform law
    0.492453        editorial-page article
    0.490993        naturalization service
    0.489448        civil fines
    0.488754        new immigration law
    0.487762        legal immigration
    0.487187        employer sanctions
    0.483245        simpson-mazzoli immigration reform
```

Alg 3

Query-flow graph

- A few consecutive queries:
  - "Brake pads"
  - "Auto repair"
  - "Auto body shop"
  - "Batteries"
  - "Car batteries"
  - "Buy car batteries online"
- Observation:
  - *Users tend to rephrase queries until their information need is satisfied*.
  - These "query-chains" can be used for query suggestion

What is a query log

- A query log *L* is a set of records:
  - $<q_i, u_i, t_i, V_i, C_i>$
  - Respectively: query, anonymized user identifier, timestamp, returned documents and clicked documents
- Users interact in sessions *S*:
  - A session is a (maximal) set of queries with inter-arrival time smaller that a given threshold (e.g. 30 mins)
- A *super session* is composed by all the sessions of a given user

What is the Query-flow graph

- G=(V,E,W)
  - **V** is the set of nodes,
    every node corresponds to a query in the query log,
    plus a "fake" starting point *s* and ending point *t*.
  - **E** is a set of *directed* edges between the nodes.
  - ***w*** is a weighting function,
    that assigns an importance to an edge between to nodes.

- Building the Query-flow graph means to build find the weight of each edge in the graph.

Finding the weights

- Task 1:
  - Are two queries q_1 and q_2 in the same chain ??
- Task 2:
  - How many times a query q_1 is reformulated with a query q_2 ??

Same-chain discovery

- When are two queries in the same chain ??
  - Def: if they satisfy the same information need.
  - Heuristic: if they are “similar”.
- When are two queries similar ?
  - TEXT: common terms, common trigrams, edit distance
  - SESSION: how many times they occur in the same session
  - TIME: distance in time

Same-chain discovery

- Take a small number of consecutive queries q_1, q_2
  - Manually label whether they are in the same chain
  - Use the some *data mining* (or genetic) algorithm to learn the impact of the various features: time distance, text similarity, etc.
  - The output is a *classifier* (function) with the following output:
    - q_1, q_2 are <u>not</u> in the same chain
    - q_1, q_2 are in the same chain
  - Use the classifier is used to set/remove edges.

Reformulation probability

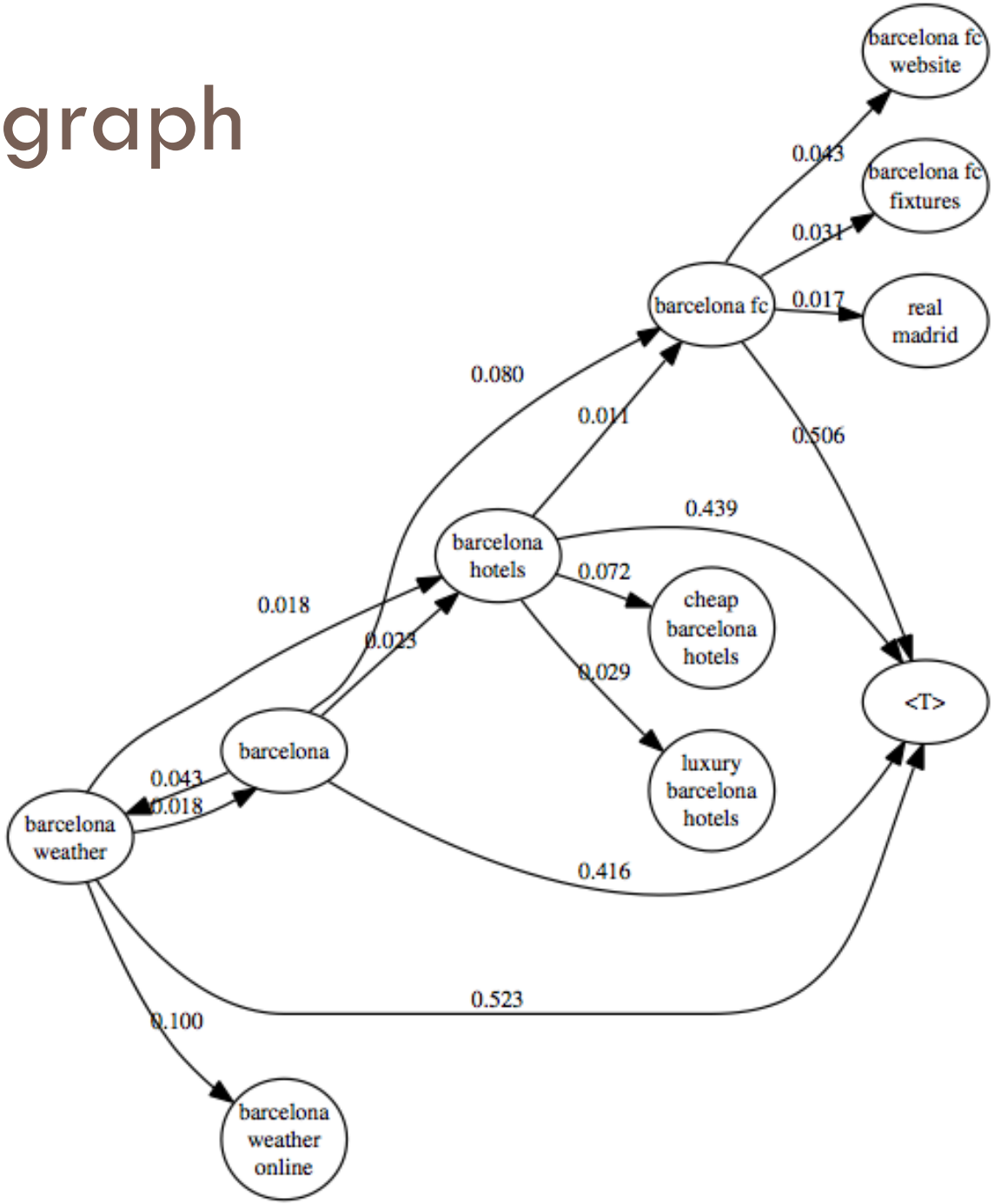
- For every "survived" edge assign weight:

$$w(q_1, q_2) = f(q_1, q_2) \,/\, f(q_1)$$

  - where $f(q_1, q_2)$ is the number of times q_1 occurs immediately after q_2,
    and $f(q_1)$ is the number of occurrences of q_1
  - … a conditional probability

Query-flow graph

- Result



How to build recommendations

- Three methods:
  - Maximum Weight
  - Random Walk
  - Random Walk with History

Maximum Weight

- Given a query q_1,
  return the query q_* with maximum $w(q_1,q_*)$

- Problem:
  - Promotes queries that are frequent anyway

Max. weight
t
apple ipod
apple store
apple trailers
amazon
apple mac
itunes
pc world
argos
currys

t
jeep cherokee
jeep grand ...
jeep wrangler
land rover
landrover
ebay
chrysler
bmw
nissan

Random Walk

- Since the Query-flow graph looks like the Web graph:
  - use PageRank
  - change the personalization vector such that the walk always restarts from the input query q_1

Random Walk Results

Max. weight	s_q	$\hat{s}_q$	$\bar{s}_q$
t	t	apple	apple
apple ipod	apple	apple fruit	apple ipod
apple store	apple ipod	apple ipod	apple trailers
apple trailers	apple store	apple belgium	apple store
amazon	apple trailers	eating apple	apple mac
apple mac	google	apple.nl	apple fruit
itunes	amazon	apple monitor	apple usa
pc world	argos	apple usa	apple ipod nano
argos	itunes	apple jobs	apple.com/ipod...
currys	pc world	apple movie ...	t
t	t	jeep	jeep
jeep cherokee	jeep	jeep trails	jeep cherokee
jeep grand ...	jeep cherokee	jeep kinderk...	jeep trails
jeep wrangler	jeep grand ...	jeep compass	jeep compass
land rover	bmw	jeep cherokee	jeep kinderkled...
landrover	jeep wrangler	swain and jon...	jeep grand ...
ebay	land rover	jeep bag	jeep wrangler
chrysler	landrover	country living ...	chryslar
bmw	chrysler	buy range rov...	jeepcj7
nissan	google	craviotto snare	buses to Knowl...

Random Walk with History

- Given the recent queries q_1, q_2, …, q_k
  - Change the personalization vector to allow restarts from any query in the recent history, such that the probability of older queries is larger

music	facebook → gabriella → music
music	music
yahoo music	gabriella
music videos	yahoo music
music downloads	music videos
free music	music downloads
yahoo music videos	free music
music yahoo	gabriella sweet like me
free music videos	lighting bug rotherham
yahoo music launch	ccp npa ndf
free music downloads	gabriela lighting