Query suggestions using **query-flow graphs**

Paolo Boldi
Francesco Bonchi
Carlos Castillo
Debora Donato
Sebastiano Vigna

Query suggestions using **query-flow graphs**

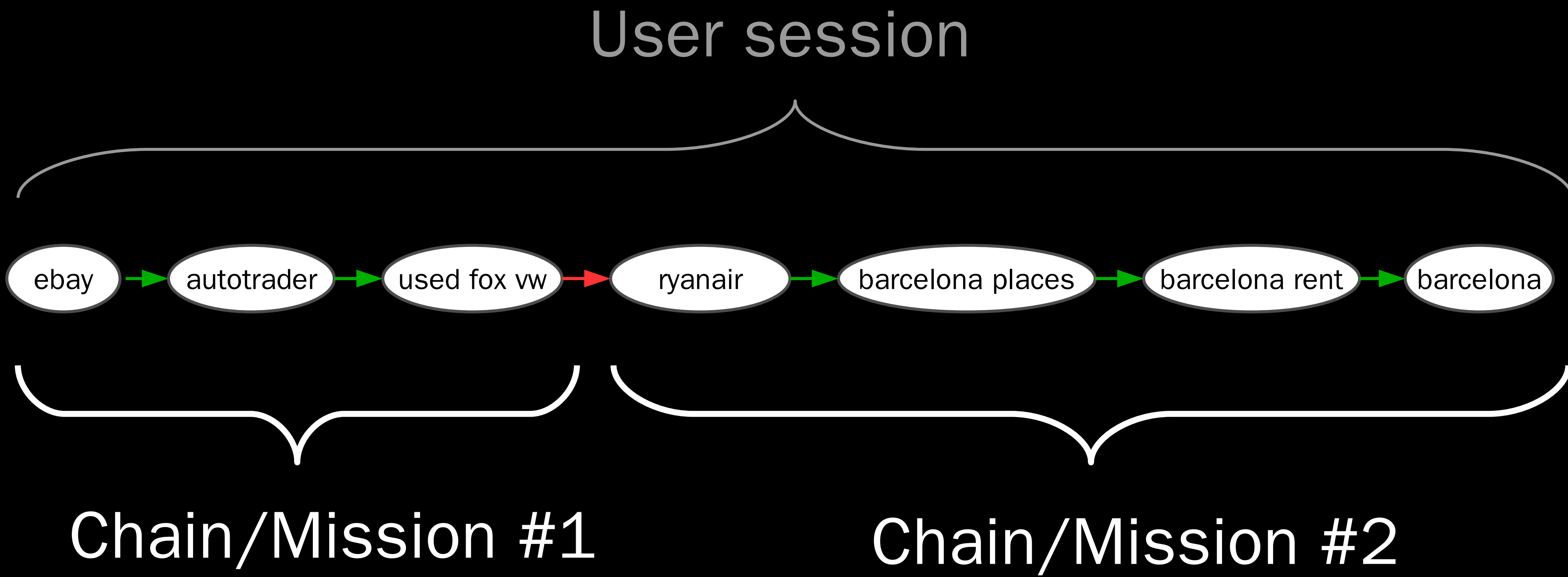
Paolo Boldi[M]
Francesco Bonchi[Y]
Carlos Castillo[Y]
Debora Donato[Y]
Sebastiano Vigna[M]

[M] *Università degli studi di Milano, Italy*

[Y] *Yahoo! Research Barcelona, Spain*

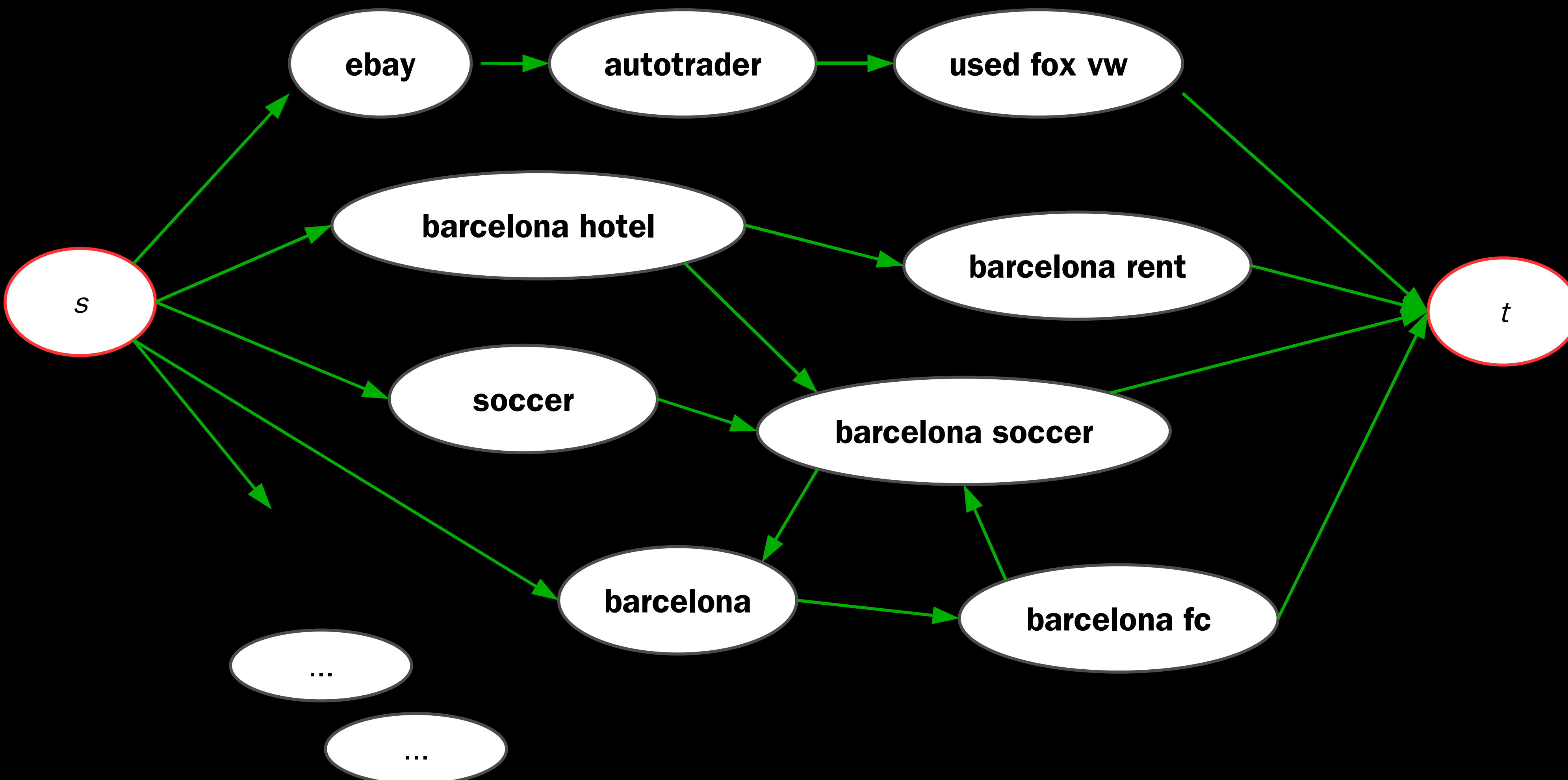
Thanks to: Aris Gionis, Marco Rosa

Notation: sessions and missions

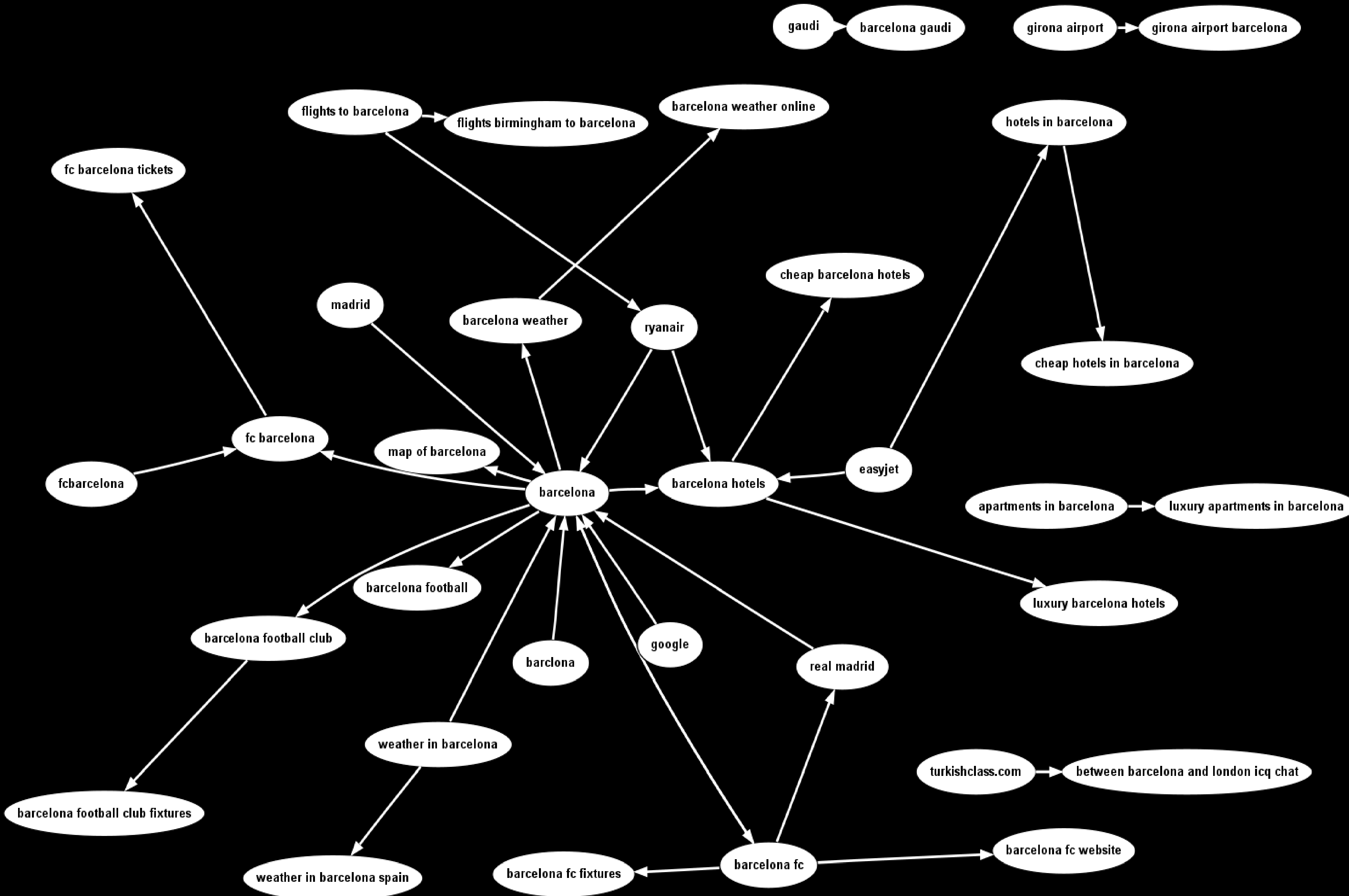


F. Radlinksi & T. Joachims: “Query chains: learning to rank from implicit feedback”. KDD 2005.

Query-Flow Graph



P. Boldi, F. Bonchi, C. Castillo, D. Donato, A. Gionis, S. Vigna: “The Query-Flow Graph”. CIKM 2008.



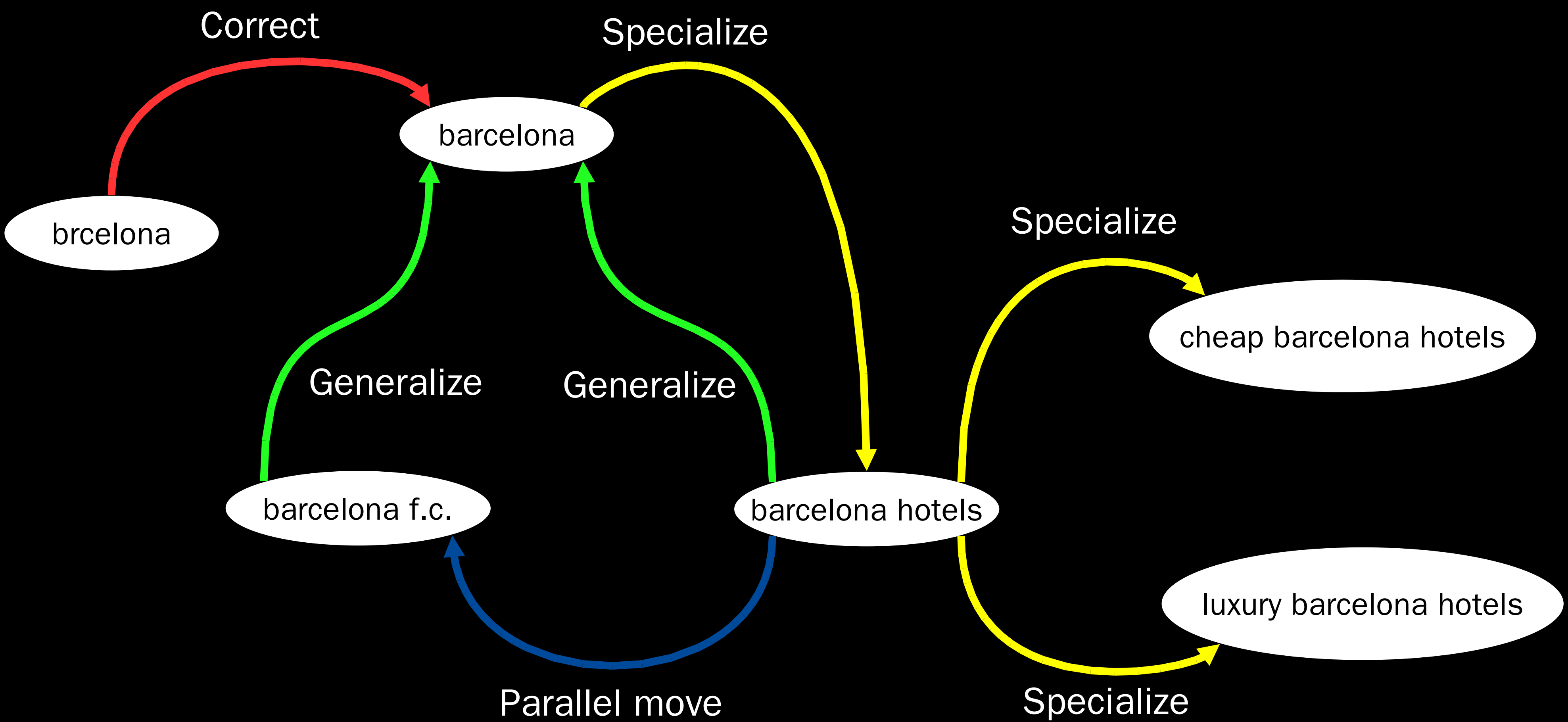
R. Baeza-Yates: “Graphs from search engine queries”. SOFSEM 2007.

The query-flow graph

- **Directed** graph
- Nodes are queries
- Arcs are reformulations
  - non-symmetrical
- Arcs have annotations
  - frequencies, similarities, etc.

P. Boldi, F. Bonchi, C. Castillo, D. Donato, A. Gionis, S. Vigna: “The Query-Flow Graph”. CIKM 2008.

Query-reformulation types



Rieh, S. Y. and Xie, H: “Analysis of multiple query reformulations on the web”. IPM 32 (3) 2006.

Reformulation types

- Correction
  - `startford cinema` → `stratford cinema`
- Generalization (“zoom out”)
  - `barcelona hotels` → `barcelona`
- Specialization (“zoom in”)
  - `barcelona soccer` → `barcelona camp nou`

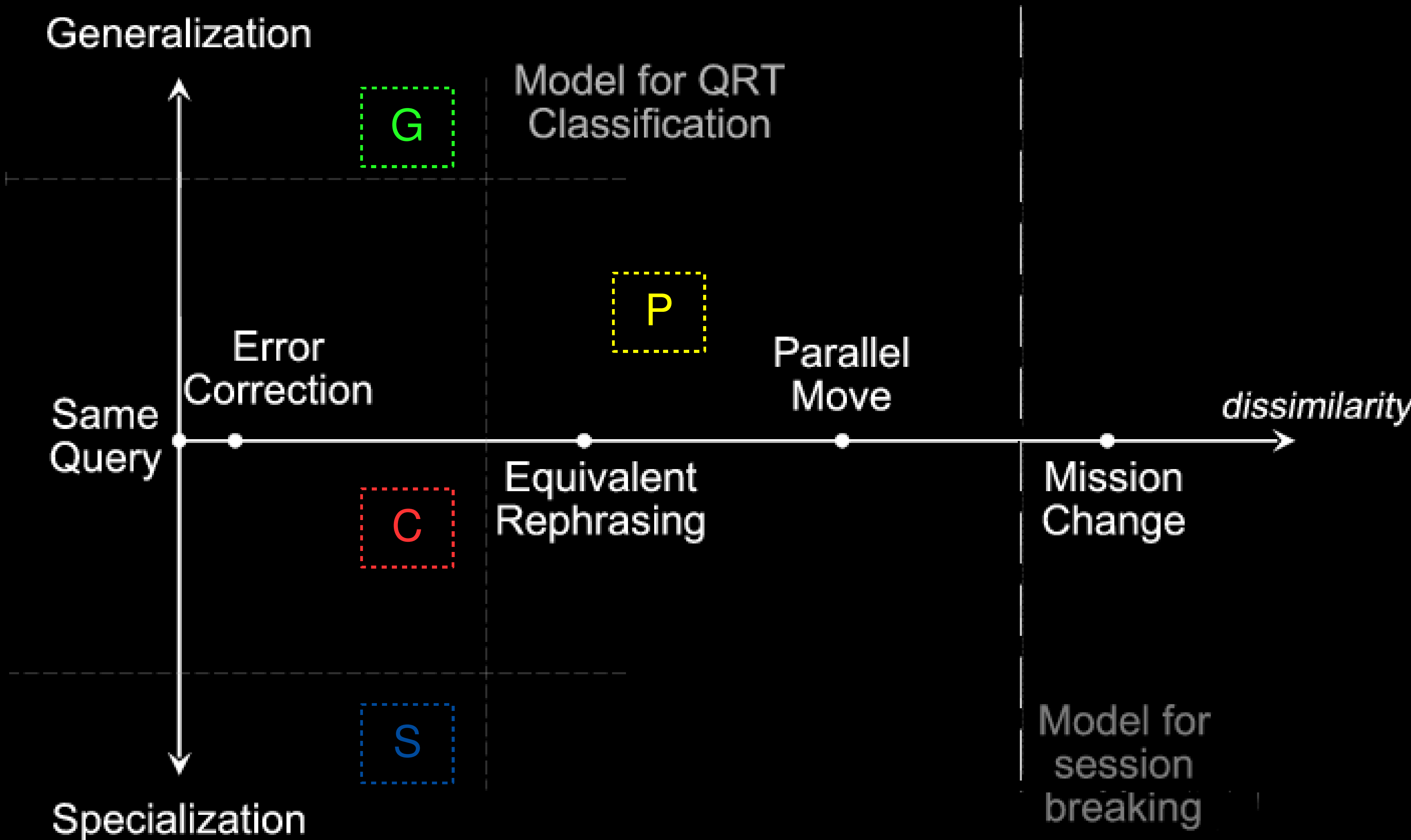
Rieh and Xie: “Analysis of multiple query reformulations”. IPM 2006.

Zoom-in, zoom-out, pan, names comes from Y!SAMA

Reformulation types

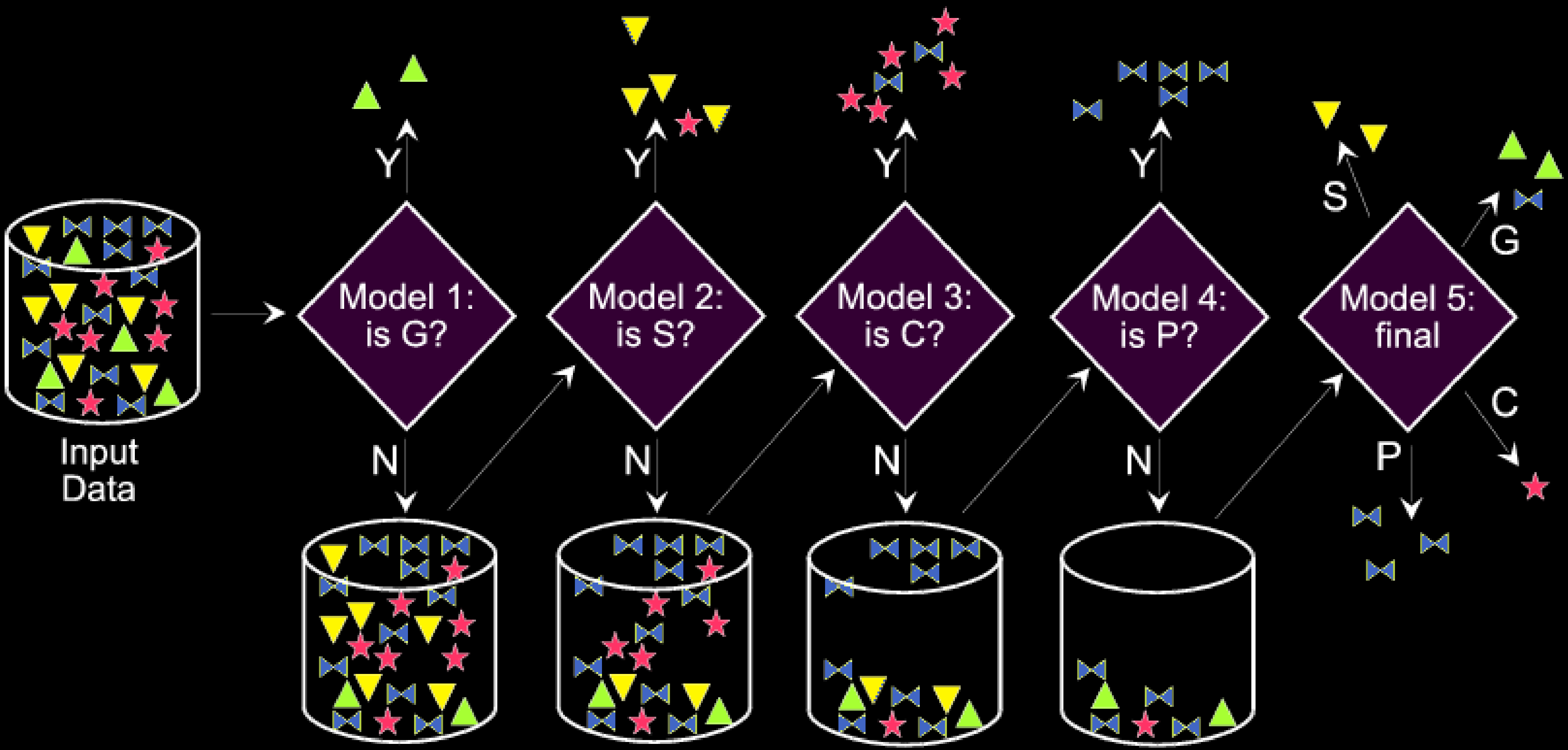
- Rephrasing
  - `wikipedia english` → `english wikipedia`
  - `robbs celebrities` → `robbs celebs`
- Parallel move
  - `barcelona` → `rome`

Rieh and Xie: “Analysis of multiple query reformulations”. IPM 2006.



P. Boldi, F. Bonchi, C. Castillo, S. Vigna: “From 'dango' to 'japanese cakes'”. 2009.

Query-reformulation classifier



P. Boldi, F. Bonchi, C. Castillo, S. Vigna: “From 'dango' to 'japanese cakes'”. 2009.

Classifier: 92% accuracy

Rule 1 of model 1: $is\_G?$	**Rule 1 of model 2:** $is\_S?$
if $terms.cosine > 0.47$ **and** $deltaLenRel \leq -0.37$ **then** $is\_G? = Y$	**if** $ngrams.cosine > 0.42$ **and** $terms.deltaLen > 1$ **then** $is\_S? = Y$

Rule 1 of model 3: $is\_C?$	**Rule 1 of model 4:** $is\_P?$
if $avgSessPosition \leq 1.91$ **and** $levenshtein \leq 3$ **then** $is\_C? = Y$	**if** $avgRelPosition > 0.65$ **and** $terms.jaccard \leq 0.25$ **and** $deltaLen \leq 5$ **and** $terms.deltaLen > 0$ **then** $is\_P? = Y$

Example 1/4: “cat”

cat Query

cat	picture of cat	cute picture of dog and cat
picture of cat 16.7%	cat 18.4%	picture of cat 11.7%
dog 5.4%	picture of funny cat and dog 13.9%	cat 10.7%
funny cat 4.1%	cute picture of dog and cat 6.2%	picture of dog 9.3%
cat picture 3.1%	picture of dog 4.8%	cute picture of dog 6.1%
cat list 2%	funny cat 3.6%	dog 3.7%
funny cat video 1.8%	dog 3.6%	cute picture of cat 3.7%
cat and dog 1.2%	cat breed 1.6%	funny cat 3.3%
chat 1.1%	picture of cute cat and kitten 1.2%	picture of funny cat and dog 3.3%
arctic cat 1%	funny picture of cat 1.1%	cute dog 2.3%
caterpillar 1%	youtube 1%	cute picture of dog and cat puppy 2.3%
cat breed 1%	kitten 0.9%	dog and cat 2.3%
cat musical 0.9%	cat and dog 0.9%	picture of dog and cat 1.9%
glow in the dark cat 0.9%	picture of kitten 0.8%	dog picture 1.9%
kitten 0.8%	picture of christmas cat 0.8%	google 1.9%
google 0.8%	type of cat 0.8%	cute picture of dog and cat cute pet 1.9%
46029 transitions, 49.6% terminal	5856 transitions, 57.6% terminal	584 transitions, 63.4% terminal

Legend: Generalization Specialization ErrorCorrection ParallelMove DifferentChain

Example 2/4: “peanut”

peanut	peanut character	charlie brown
peanut character 11.6%	peanut gang character 31.7%	charlie brown christmas 14.4%
jeff dunham video clip peanut 6.5%	peanut 13.1%	charlie brown character 14.3%
planter peanut 5.8%	peanut cartoon character 13.1%	snoopy 3.3%
jeff dunham 2.2%	charlie brown 5.5%	charlie brown kwanzaa 3.1%
peanut butter 2.2%	weather channel 4.8%	charlie browns 2.8%
garfield 1.8%	snoopy 4.8%	charlie brown picture 2.7%
archie 1.4%	planter peanut 2.8%	charlie brown song 2.5%
george washington carver 1.4%	peanut plant 2.1%	charlie brown theme song 1.8%
growing peanut 1.4%	peanut character snoopy 1.4%	charlie brown movie 1.7%
for better or for worse 1.3%	peanut gang 1.4%	google 1.7%
peanut nutrition 1.2%	peanut poster 1.4%	charlie brown restaurant 1.6%
boiled peanut 1.2%	peanut woodstock 1.4%	youtube 1.3%
jeff dunham peanut 1.1%	the peanut character 1.4%	charlie brown dance 1.3%
cashew 1.1%	the simpsons 1.4%	charlie brown comic 1.2%
nut 1.1%	wal mart 1.4%	charlie brown image 1.1%
3383 transitions, 56.7% terminal	592 transitions, 75.5% terminal	2374 transitions, 65.1% terminal

Legend: Generalization Specialization ErrorCorrection ParallelMove DifferentChain

Example 3/4: “surf board”

surf board	ron jon surf shop	o neill
surfboard 18.5%	ron jon surf shop cocoa beach 23%	o neill clothing 24.6%
rusty surf board 11.5%	surf shop 9.7%	billabong 7.5%
lost surf board 7.8%	ron jon surf shop orlando 6.4%	hurley 5%
channel island surf board 7.3%	pac sun 2%	roxy 4.4%
surf board for sale 3.1%	billabong 1.8%	volcom 3.5%
used surf board 3.1%	roxy 1.5%	pac sun 2.3%
surf boards 2.1%	hollister 1.5%	quicksilver 2.1%
ebay 1.8%	roxy swim wear 1.5%	quiksilver 2.1%
ron jon surf shop 1.6%	ron jon surf shop florida 1.5%	o neil 2.1%
cheap surf board 1.4%	surf board 1.3%	rip curl 1.9%
buy surf board 1.4%	o neill 1.3%	o neill wetsuits 1.5%
snowboard 1.3%	jacks surf shop 1.3%	o neill bag 1.3%
surfboard design 1.3%	victorias secret 1.3%	google 1%
surf shop 1.2%	quicksilver 1.3%	ebay 1%
ron jon 1%	zumiez 1%	oneill.com 0.8%
1512 transitions, 49.2% terminal	1172 transitions, 66.6% terminal	997 transitions, 52% terminal

Legend: Generalization Specialization ErrorCorrection ParallelMove DifferentChain

Example 4/4: “bruce springsteen”

bruce springsteen	bon jovi	def leppard
ticketmaster 4.8%	ticketmaster 4.3%	ticketmaster 2.8%
bruce springsteen lyric 4.5%	jon bon jovi 3.6%	def leppard lyric 2.7%
bruce springsteen ticket 2.2%	bon jovi lyric 3.4%	styx 2.6%
youtube 1.7%	bon jovi ticket 2%	reo speedwagon 2.1%
bruce springsteen song 1.2%	youtube 1.9%	youtube 1.7%
ebay 1.1%	ebay 1.8%	journey 1.5%
myspace 1.1%	daughtry 1.6%	whitesnake 1.5%
bon jovi 1.1%	def leppard 1.2%	van halen 1.5%
bruce springsteen biography 1%	bon jovi song 1.1%	bon jovi 1.4%
john mellencamp 0.8%	greensboro coliseum 1%	poison 1.3%
van halen 0.8%	myspace 1%	ac dc 1.2%
billy joel 0.8%	richie sambora 0.9%	myspace 1.1%
google 0.8%	aerosmith 0.9%	metallica 1.1%
stubhub 0.8%	hotmail 0.8%	ebay 1%
backstreets 0.8%	chris daughtry 0.8%	iron maiden 1%
9954 transitions, 70.4% terminal	18380 transitions, 65.8% terminal	12695 transitions, 59.7% terminal

Legend: Generalization Specialization ErrorCorrection ParallelMove DifferentChain

Reformulation types

- Parallel moves (50%-60%)
  - The most frequent class
- Specializations (30%-40%)
- Generalizations (5%-10%)
  - Frequently appear together in alternating order
- Corrections (5%-10%)
  - More frequent at the beginning or end of a chain

Query recommendation setting

Dataset: “Spring 2006 Microsoft data”

- Provided by WSCD'09 workshop organizers
- 15M queries from a 1-month period

Automatic chains and reformulations labels

- Obtained with models from previous works

Recommendations based on **random walks**

`WebGraph` (graph) `SUX4J` (hashing)

Implemented systems

Slices

Queryflow-{G, S, P, C, SP, GSP, GSPC, ...}

Composed graphs

Queryflow-{(SS^T), (SG), ...}

Weight is weight of heaviest length-2 path

Random walks

Number of steps: 1, 5, 10 (2, 6, 12)

Scoring: PPR (absolute) or PPR/PR (relative)

Self-transition with prob. 0.9, no random jumps

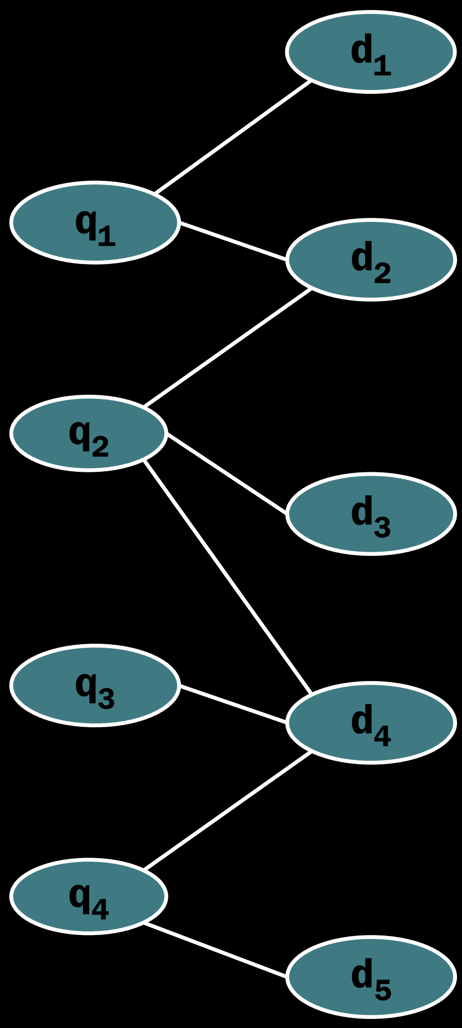
Click graphs

Weights based on clicks

Two weighting schemes:

$$w_f(i,j) = \frac{c(i,j)}{\sum_{k:(i,k)\in E} c(i,k)}$$

$$w_b(i,j) = \frac{w_f(j,i)}{\sum_{k:(j,k)\in E} w_f(j,k)}$$



N. Craswell and M. Szummer: "Random walks on the click graph" SIGIR 2007.

Evaluation method

Sampled 114 queries (freq 700 - 15,000)

Run all systems for each query

Create a pool: union of top-5 recommendations

- ~ 6,000 query recommendations in total

Evaluate each recommendation in the pool

- Useful, somewhat useful, not useful

Assessment

Example, query “cnn news”:

Useful	**Somewhat usf.**	**Not useful**
cnn world news	abc7chicagonews	CNN
msnbc news	nba scores	cnn.com
fox news	cnnfyi	verizon e-mail

Distribution of assessments

$n = 6\ 093$

Assessment	Probability
Useful	25.1%
Somewhat useful	11.6%
Not useful	62.1%
Can not assess	1.2%

Inter-assessor agreement

$n = 560$

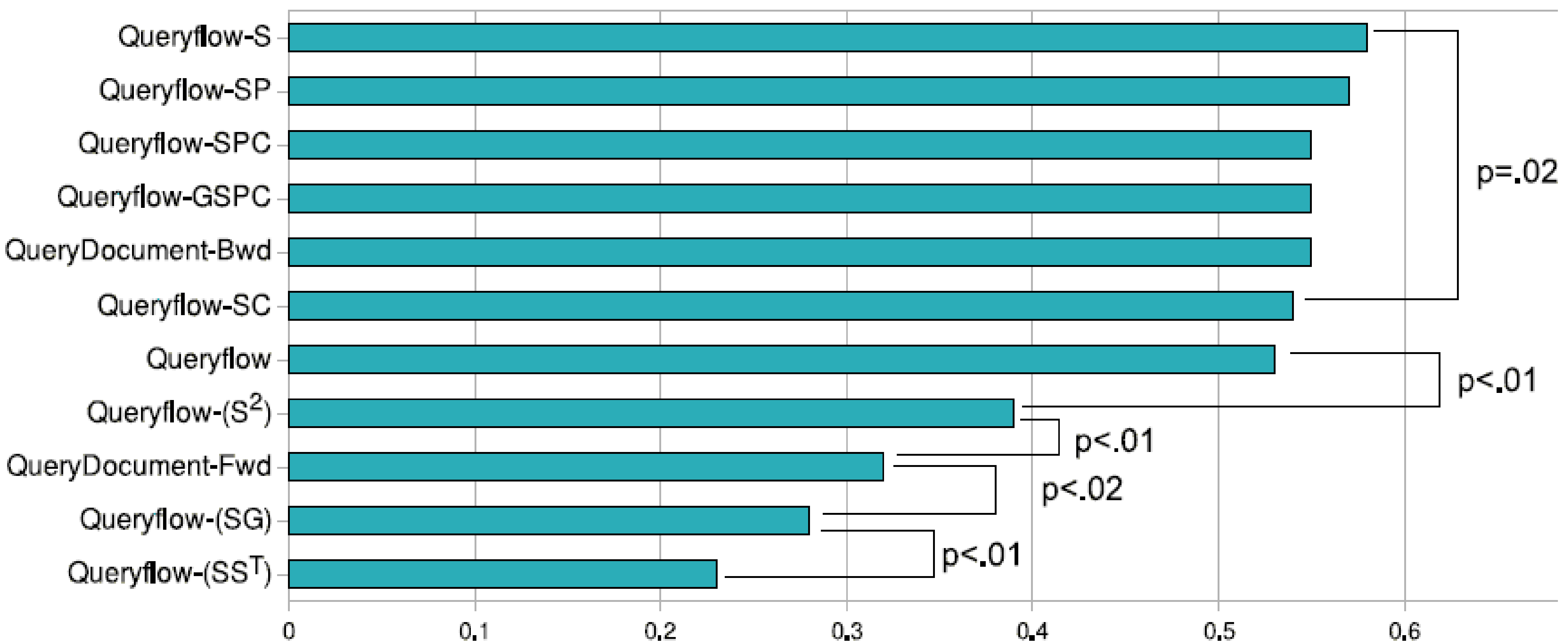
	Scenario	P_a	κ
A.	Useful vs Sw.useful vs Not useful	68%	0.43
B.	Useful vs (Sw.useful or Not useful)	86%	0.46
C.	(Useful or Sw.useful) vs Not useful	77%	0.59

Inter-assessor agreement

$$n = 560$$

	Scenario	P_a	κ
A.	Useful vs Sw.useful vs Not useful	68%	0.43
B.	Useful vs (Sw.useful or Not useful)	86%	0.46
C.	$\underbrace{\text{(Useful or Sw.useful)}}_{\text{Good}}$ vs $\underbrace{\text{Not useful}}_{\text{Bad}}$	77%	0.59

Precision @ 5



Queryflow-S
Queryflow-SP
Queryflow-SPC
Queryflow-GSPC
QueryDocument-Bwd
Queryflow-SC
Queryflow
Queryflow-(S²)
QueryDocument-Fwd
Queryflow-(SG)
Queryflow-(SSᵀ)
p=.02
p<.01
p<.01
p<.02
p<.01
0
0.1
0.2
0.3
0.4
0.5
0.6

Precision @ 5

Uscore	p-value	System	Iter.	Scoring
0.58		Queryflow-S	10	Abs.
0.58		Queryflow-S	5	Abs.
0.57		Queryflow-SP	1	Abs.
0.56		Queryflow-SP	10	Abs.
0.56		Queryflow-SP	5	Abs.
0.55	0.10	Queryflow-SPC	1	Abs.
0.55	0.06	Queryflow-GSPC	1	Abs.
0.55	0.06	Queryflow-S	1	Abs.
0.55	0.07	Queryflow-SPC	5	Abs.
0.55	0.06	Queryflow-SPC	10	Abs.
0.55	0.07	QueryDocument-Bwd	6	Rel.
0.55	0.06	QueryDocument-Bwd	12	Rel.
0.54	0.04	Queryflow-S	1	Rel.
0.54	0.03	Queryflow-S	10	Rel.
0.54	0.02	Queryflow-SC	5	Abs.
0.54	0.02	Queryflow-S	5	Rel.
0.54	0.02	QueryDocument-Bwd	2	Rel.
0.54	0.04	QueryDocument-Bwd	12	Abs.
0.54	0.02	QueryDocument-Bwd	6	Abs.
0.53	0.01	Queryflow	1	Abs.
0.53	0.01	Queryflow-GSPC	5	Abs.
0.53	0.01	Queryflow-GSPC	10	Abs.
0.53	0.01	Queryflow-SC	10	Abs.
0.52	$< .01$	Queryflow	5	Abs.
0.52	$< .01$	QueryDocument-Bwd	2	Abs.

Diversity score

Take one query

Get top-5 recommendations that are **good**

Get top-5 **search results** for each of them

Count how many **distinct URLs** appear

- Discard URLs from initial query, this is in [0,25]

Diversity scores

Dscore	p-value	System	Iter.	Scoring
13.49		Queryflow-S	10	Abs.
13.44		Queryflow-S	5	Abs.
13.20		Queryflow-SP	1	Abs.
13.04		Queryflow-SP	10	Abs.
12.99		Queryflow-SP	5	Abs.
12.84		Queryflow-SCP	1	Abs.
12.73		Queryflow-SCP	5	Abs.
12.70		Queryflow-GSPC	1	Abs.
12.70		Queryflow-SCP	10	Abs.
12.52		Queryflow-S	1	Rel.
12.42		Queryflow-S	1	Abs.
12.40		Queryflow-S	10	Rel.
12.38		Queryflow	1	Abs.
12.38		Queryflow-GSPC	5	Abs.
12.37		Queryflow-S	5	Rel.
12.33		Queryflow-SC	5	Abs.
12.33		Queryflow-GSPC	10	Abs.
12.28	0.10	QueryDocument-Bwd	6	Rel.
12.25	0.10	Queryflow-SC	10	Abs.
12.21	0.08	QueryDocument-Bwd	12	Rel.
12.21	0.08	QueryDocument-Bwd	12	Abs.
12.16	0.08	QueryDocument-Bwd	2	Rel.
12.11	0.06	QueryDocument-Bwd	6	Abs.
12.01	0.08	Queryflow	5	Abs.
11.97	0.05	QueryDocument-Bwd	2	Abs.
11.92	0.05	Queryflow-SC	1	Rel.
11.89	0.07	Queryflow	10	Abs.
11.77	0.04	Queryflow-SC	1	Abs.
11.64	0.03	Queryflow-SC	10	Rel.
11.60	0.03	Queryflow-SC	5	Rel.
11.13	0.01	Queryflow-SP	1	Rel.
11.04	0.01	Queryflow-SP	5	Rel.
10.94	0.01	Queryflow-SP	10	Rel.

Conclusions and future work (1/3)

Recognizing chains is necessary to match the baseline

Fewer slices is better than more

Few iterations are enough

Conclusions and future work (2/3)

Why 5 recommendations?
- Let the system decide

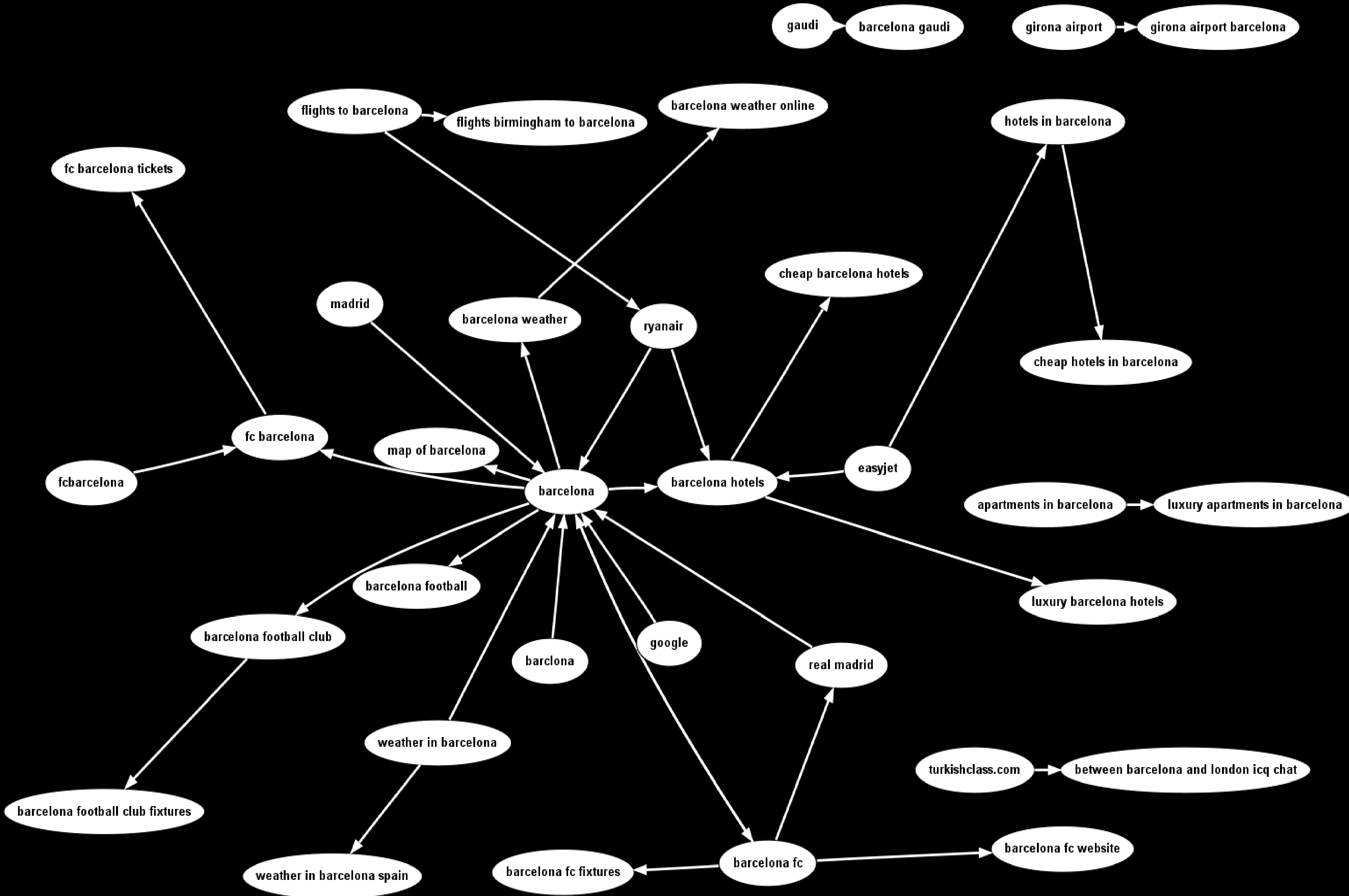
Diverse recommendations
- A good mix of parallel moves, specializations, etc.

Parallel moves are useful and valuable
- But sometimes they are a risky bet

Improve query reformulation classifier
- e.g. by learning simultaneously the query topic

Conclusions and future work (3/3) (history-based recommendation)

banana → apple → ?	**beatles → apple → ?**	**apple → ?**
banana	beatles	apple
apple	apple	apple ipod
usb no	apple ipod	apple trailers
banana cs	scarring	apple store
giant chocolate bar	srg peppers artwork	apple mac
where is the seed in a nut	ill get you	apple fruit
banana shoe	bashles	apple usa
fruit banana	dundee folk songs	apple ipod nano
banana cloths	the beatles love album	apple.com/ipod
eating bugs	place lyrics beatles	*t*



gaudi
barcelona gaudi
girona airport
girona airport barcelona
flights to barcelona
flights birmingham to barcelona
barcelona weather online
hotels in barcelona
fc barcelona tickets
cheap barcelona hotels
madrid
barcelona weather
ryanair
cheap hotels in barcelona
fc barcelona
map of barcelona
easyjet
fcbarcelona
barcelona
barcelona hotels
apartments in barcelona
luxury apartments in barcelona
barcelona football
luxury barcelona hotels
barcelona football club
google
barclona
real madrid
weather in barcelona
turkishclass.com
between barcelona and london icq chat
barcelona football club fixtures
barcelona fc website
barcelona fc
barcelona fc fixtures
weather in barcelona spain

Q&A