Context-Aware Top-k Processing Using Views

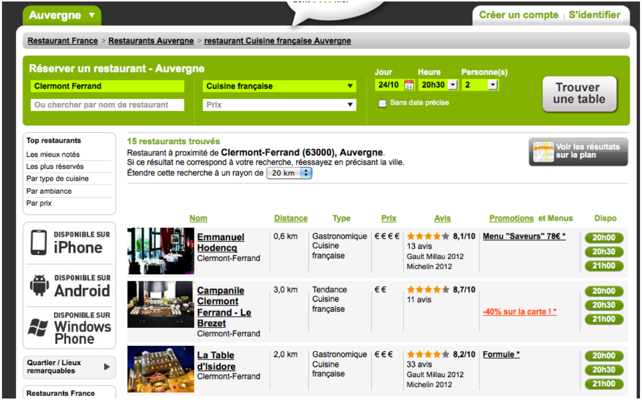
Silviu Maniu, Bogdan Cautis

University of Hong Kong & Univ. Paris-Sud / INRIA Saclay

CIKM 2013

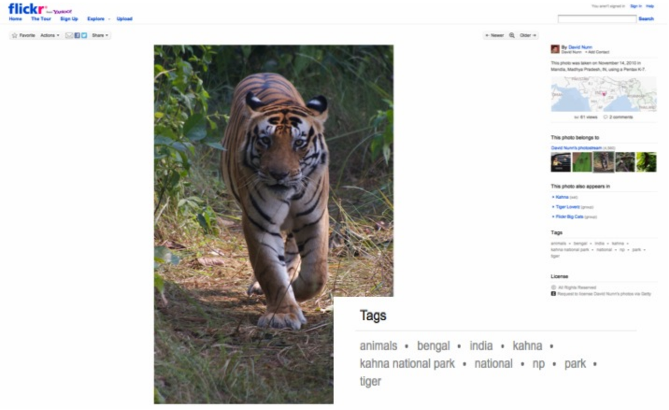
Location-aware top-k retrieval

Users search for specific types of restaurants near a given location.



Social-aware top-k retrieval

In social tagging applications (Flickr, Delicious, Twitter), users search for photos/pages/items having certain tags.



Outline

Context-aware top-k retrieval

Uncertainty in views

View-based top-k processing

Refinements

Experiments

Context-aware top-k retrieval

- Collection of objects $\mathcal{O}$, attributes $\mathcal{T}$ (e.g., keywords, tags)
- For a given context parameter $\mathcal{C}$, objects o are associated to certain attributes t, by a function $score(o, t \mid \mathcal{C})$
  - extended to a set of attributes by monotone aggregation (e.g., sum).

    $$score(o, \{t_1, \ldots, t_n\} \mid \mathcal{C}) = \sum(score(o, t_1 \mid \mathcal{C}), \ldots, score(o, t_n \mid \mathcal{C}))$$

Problem (context-aware top-k retrieval)

Given a query $Q = \{t_1, \ldots, t_n\} \subset \mathcal{T}$ and a context $\mathcal{C}$, retrieve the k objects $o \in \mathcal{O}$ having the highest values $score(o, Q \mid \mathcal{C})$.

Social-aware top-k retrieval

[Amer-Yahia et al. VLDB'08, Shenkel et al. SIGIR'08, Maniu et al. CIKM'13]

Top-k retrieval in social tagging applications:

- Collaborative tagging environment: objects (e.g., photos), users, attributes (tags), a relation `Tagged(object,user,tag)`
- Social network: associates to pairs of users a social proximity value (σ) (e.g., $[0, 1]$ similarity in tagging)
- Social score model: a seeker-dependent score (for seeker s)

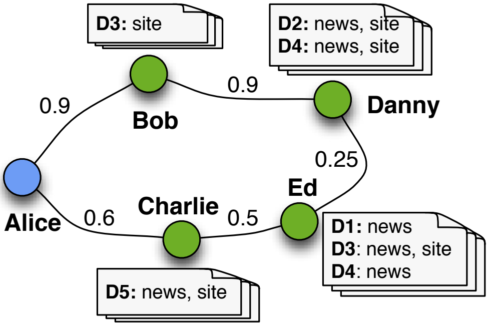
$$score(o, t \mid s) = \sum_{u \in \{v \mid Tagged(o,u,t)\}} \sigma(s, u)$$

Problem (social-aware top-k retrieval)

Given a query $Q = \{t_1, \ldots, t_n\}$ and a context (e.g., the seeker s), retrieve the k objects having the highest scores.

Social-aware top-k retrieval

Alice wants the top two documents for the query $\{news, site\}$
$\rightsquigarrow$ a social-aware result: **D4, D2**



news doc	tf	*site* doc	tf
D4	1.11	D3	1.20
D2	0.81	D4	0.81
D5	0.60	D2	0.81
D3	0.30	D5	0.60
D1	0.30	D1	0.00

user	**prox.**
Bob	0.90
Danny	0.81
Charlie	0.60
Ed	0.30

Location-aware top-k retrieval

[Cong et al. VLDB'09, Christoforaki et al. CIKM'11, Cao et al. SIGMOD'11]

Top-k retrieval in spatial applications:

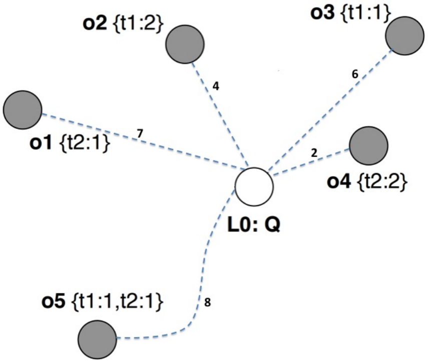
- Objects (e.g., documents) with attributes and geo-location.
- Spatial score model: combine textual and location relevance:

$$score(o, t \mid loc, \alpha) = \alpha \times tf(t, o) + (1 - \alpha) \times dist(o, loc)$$

Problem (location-aware top-k retrieval)

Given a query $Q = \{t_1, \ldots, t_n\}$, a context (e.g., location and α), retrieve the k objects having the highest scores.

Location-aware top-k retrieval



Top-2 query **Q**={t1,t2}, $\alpha = 0.7$ at **L0** : **o4:0.92** and **o2:0.85**

Query answering using views

Context-aware retrieval is inherently difficult: joint exploration of the textual and "contextual" (e.g., spatial or social) space.

Our goal: improve efficiency by materialization, exploiting results of previous searches (views).

Each view has a context: its usefulness is proportional to distance w.r.t. the new context $\rightsquigarrow$ score uncertainty, approximate top-k results.

Outline

Context-aware top-k retrieval

Uncertainty in views

View-based top-k processing

Refinements

Experiments

Context transposition

Focus on two applications: location-aware search, social-aware search

The context $\mathcal{C}^V$ of a view V is a pair $(\mathcal{C}^V.l, \mathcal{C}^V.\alpha)$:

- location $\mathcal{C}^V.l$: geo-coordinates or seeker Id in a social network
- contextual parameter $\mathcal{C}^V.\alpha$: the weight of the context in scores

Context transposition

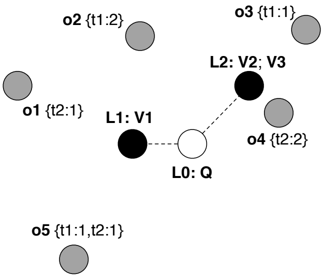
Focus on two applications: location-aware search, social-aware search

The context $\mathcal{C}^V$ of a view V is a pair $(\mathcal{C}^V.l, \mathcal{C}^V.\alpha)$:

- location $\mathcal{C}^V.l$: geo-coordinates or seeker Id in a social network
- contextual parameter $\mathcal{C}^V.\alpha$: the weight of the context in scores

Transposition: adapt results for $(\mathcal{C}^V.l, \mathcal{C}^V.\alpha)$ to a new context $(\mathcal{C}.l, \mathcal{C}.\alpha)$

Context transposition yields uncertainty



V1=(L1,{t1,t2})

o	sc
o5	1.062
o4	1.029
o2	1.000

V2=(L2,{t1})

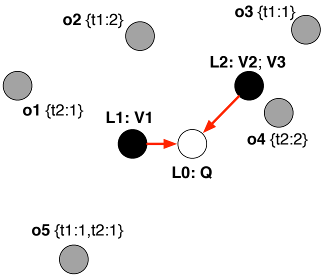
o	sc
o2	0.946
o3	0.575
o5	0.450
o4	0.262

V3=(L2,{t2})

o	sc
o4	0.962
o5	0.450
o1	0.437
o2	0.246

Top-2 query **Q**={t1,t2} at location **L0**

Context transposition yields uncertainty



V1=(L1,{t1,t2})

o	sc
o5	1.062
o4	1.029
o2	1.000

V2=(L2,{t1})

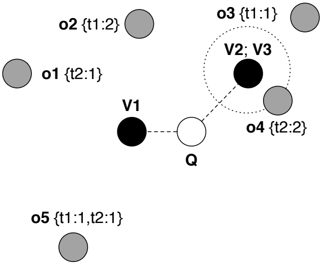
o	sc
o2	0.946
o3	0.575
o5	0.450
o4	0.262

V3=(L2,{t2})

o	sc
o4	0.962
o5	0.450
o1	0.437
o2	0.246

Top-2 query **Q**={t1,t2} at location **L0**

Context transposition yields uncertainty



V1=(L1,{t1,t2})

o	sc
o5	1.062
o4	1.029
o2	1.000

V2=(L2,{t1})

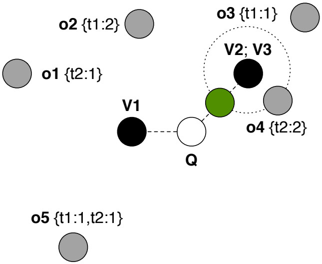
o	sc
o2	0.946
o3	0.575
o5	0.450
o4	0.262

V3=(L2,{t2})

o	sc
o4	0.962
o5	0.450
o1	0.437
o2	0.246

distance of **o4** to **Q** unknown, but within [0.987, 1.037] interval

Context transposition yields uncertainty



V1=(L1,{t1,t2})

o	sc
o5	1.062
o4	1.029
o2	1.000

V2=(L2,{t1})

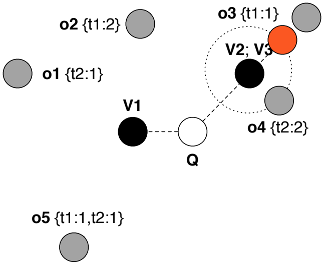
o	sc
o2	0.946
o3	0.575
o5	0.450
o4	0.262

V3=(L2,{t2})

o	sc
o4	0.962
o5	0.450
o1	0.437
o2	0.246

distance of **o4** to **Q** unknown, but within [0.987, 1.037] interval

Context transposition yields uncertainty



V1=(L1,{t1,t2})

o	sc
o5	1.062
o4	1.029
o2	1.000

V2=(L2,{t1})

o	sc
o2	0.946
o3	0.575
o5	0.450
o4	0.262

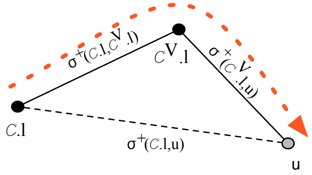
V3=(L2,{t2})

o	sc
o4	0.962
o5	0.450
o1	0.437
o2	0.246

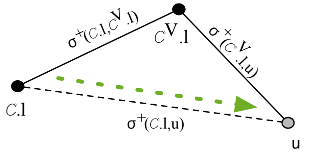
distance of **o4** to **Q** unknown, but within [0.987, 1.037] interval

Context transposition yields uncertainty

Reasoning based on shortest paths, i.e., the optimal is through:



- a path that has as prefix the $\mathcal{C}.l \rightsquigarrow \mathcal{C}^V.l$ path - worstscore



- other known paths - bestscore

Uncertain views

- For an input query Q, after context transposition (if necessary),
- A view V is composed of:
  1. a *definition* $def(V)$: a pair query-context $(Q^V, \mathcal{C}^V)$
  2. an *answer set* $ans(V)$: triples (o_i, wsc_i, bsc_i), indicating that object o_i has a score in the range $[wsc_i, bsc_i]$

Outline

- Context-aware top-k retrieval
- Uncertainty in views
- **View-based top-k processing**
- Refinements
- Experiments

Using the views for one object's bounds

Given a view set $\mathcal{V}$ and a query Q sharing the same context, compute the tightest worst-score / best-score bounds for some object o.

Via a linear program:

$$\max \sum_{t_i \in Q} \mathbf{sc}(\mathbf{o}, \mathbf{t_i} \mid \mathcal{C}) \tag{1}$$

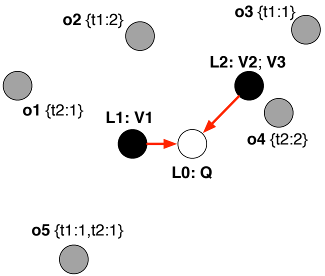
$$\min \sum_{t_i \in Q} \mathbf{sc}(\mathbf{o}, \mathbf{t_i} \mid \mathcal{C}) \tag{2}$$

$$wsc \leq \sum_{t_j \in Q^V} \mathbf{sc}(\mathbf{o}, \mathbf{t_j} \mid \mathcal{C}),\ \forall V \in \mathcal{V}\ s.t.\ (o, wsc, bsc) \in ans(V)$$

$$\sum_{t_j \in Q^V} \mathbf{sc}(\mathbf{o}, \mathbf{t_j} \mid \mathcal{C}) \leq bsc,\ \forall V \in \mathcal{V}\ s.t.\ (o, wsc, bsc) \in ans(V)$$

$$\mathbf{sc}(\mathbf{o}, \mathbf{t_l} \mid \mathcal{C}) \geq 0, \forall t_l \in \mathcal{T}$$

Before context transposition



V1=(L1,{t1,t2})

o	sc
o5	1.062
o4	1.029
o2	1.000

V2=(L2,{t1})

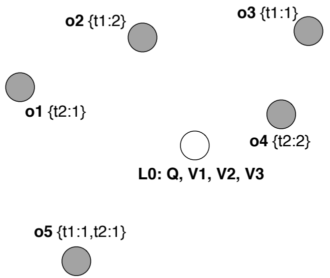
o	sc
o2	0.946
o3	0.575
o5	0.450
o4	0.262

V3=(L2,{t2})

o	sc
o4	0.962
o5	0.450
o1	0.437
o2	0.246

Top-2 query **Q**={t1,t2} at location **L0**

After context transposition



V1=(L0,{t1,t2})

o	wsc	bsc
o5	0.957	1.167
o4	0.924	1.134
o2	0.895	1.105

V2=(L0,{t1})

o	wsc	bsc
o2	0.871	1.000
o3	0.500	0.650
o5	0.500	0.525
o4	0.187	0.337

V3=(L0,{t2})

o	wsc	bsc
o4	0.987	1.037
o5	0.500	0.525
o1	0.362	0.512
o2	0.171	0.321

How can we use the views to compute the top-2 for Q?

Using views for one object: example

Top-k using views with uncertain scores:

LP formulation to compute tightest bounds - e.g., for **o5**:

$$\begin{aligned}
\max \quad & \mathbf{sc}(\mathbf{o5}, \mathbf{t1} \mid \mathcal{C}) + \mathbf{sc}(\mathbf{o5}, \mathbf{t2} \mid \mathcal{C}) && (3)\\
\min \quad & \mathbf{sc}(\mathbf{o5}, \mathbf{t1} \mid \mathcal{C}) + \mathbf{sc}(\mathbf{o5}, \mathbf{t2} \mid \mathcal{C}) && (4)\\
0.957 \leq \; & \mathbf{sc}(\mathbf{o5}, \mathbf{t1} \mid \mathcal{C}) + \mathbf{sc}(\mathbf{o5}, \mathbf{t2} \mid \mathcal{C}) \leq 1.167 && (V1)\\
0.500 \leq \; & \mathbf{sc}(\mathbf{o5}, \mathbf{t1} \mid \mathcal{C}) \leq 0.525 && (V2)\\
0.500 \leq \; & \mathbf{sc}(\mathbf{o5}, \mathbf{t2} \mid \mathcal{C}) \leq 0.525 && (V3)
\end{aligned}$$

Using views for one object: example

Top-k using views with uncertain scores:

LP formulation to compute tightest bounds - e.g., for **o5**:

$$
\begin{aligned}
\max \quad & \mathbf{sc}(\mathbf{o5},\mathbf{t1} \mid \mathcal{C}) + \mathbf{sc}(\mathbf{o5},\mathbf{t2} \mid \mathcal{C}) && (3)\\
\min \quad & \mathbf{sc}(\mathbf{o5},\mathbf{t1} \mid \mathcal{C}) + \mathbf{sc}(\mathbf{o5},\mathbf{t2} \mid \mathcal{C}) && (4)\\
0.957 \leq\ & \mathbf{sc}(\mathbf{o5},\mathbf{t1} \mid \mathcal{C}) + \mathbf{sc}(\mathbf{o5},\mathbf{t2} \mid \mathcal{C}) \leq 1.167 && (V1)\\
0.500 \leq\ & \mathbf{sc}(\mathbf{o5},\mathbf{t1} \mid \mathcal{C}) \leq 0.525 && (V2)\\
0.500 \leq\ & \mathbf{sc}(\mathbf{o5},\mathbf{t2} \mid \mathcal{C}) \leq 0.525 && (V3)
\end{aligned}
$$

$\leadsto$ score interval for **o5** between [1.000,1.050]

Our approach for top-k using views

Adapt the TA/NRA early-termination algorithms to the case of uncertain scores $\leadsto$ the $\mathrm{SR\text{-}TA}$ and $\mathrm{SR\text{-}NRA}$ algorithms.

Our approach for top-k using views

Adapt the TA/NRA early-termination algorithms to the case of uncertain scores $\rightsquigarrow$ the $\mathrm{SR\text{-}TA}$ and $\mathrm{SR\text{-}NRA}$ algorithms.

Plug the LPs in:

- the computation of worst-score/ best-score bounds,
- the computation of the termination threshold.

Most informative answer

In some cases, the exact top-k cannot be extracted with full confidence.

In our running example, at termination:

Candidates

obj	wsc	bsc
o4	1.174	1.134
o2	1.042	1.105
o5	1.000	1.050
o3	0.500	0.971
*	0	0.849

Most informative answer

In some cases, the exact top-k cannot be extracted with full confidence.

In our running example, at termination:

Candidates

obj	wsc	bsc
o4	1.174	1.134
o2	1.042	1.105
o5	1.000	1.050
o3	0.500	0.971
*	0	0.849

- one object guaranteed in the top-2: $G = \{o4\}$

Most informative answer

In some cases, the exact top-k cannot be extracted with full confidence.

In our running example, at termination:

Candidates

obj	wsc	bsc
o4	1.174	1.134
o2	1.042	1.105
o5	1.000	1.050
o3	0.500	0.971
*	0	0.849

- one object guaranteed in the top-2: $G = \{o4\}$
- objects that may be in the top-2: $P = \{o2, o5\}$

Most informative answer

In some cases, the exact top-k cannot be extracted with full confidence.

In our running example, at termination:

Candidates

obj	wsc	bsc
o4	1.174	1.134
o2	1.042	1.105
o5	1.000	1.050
o3	0.500	0.971
*	0	0.849

- one object guaranteed in the top-2: $G = \{o4\}$
- objects that may be in the top-2: $P = \{o2, o5\}$
- all other objects cannot be in the top-2

Top-k using uncertain views

Problem (Top-k retrieval using uncertain views)

Given a query $Q = \{t_1, \ldots, t_n\} \subset \mathcal{T}$ and a context $\mathcal{C}$, given a set of views $\mathcal{V}$, retrieve from $\mathcal{V}$ the most informative answer (G, P), with

- *$G \subset \mathcal{O}$ consisting of all guaranteed objects; i.e., in any data instance, they are in the top-k for Q and $\mathcal{C}$.*
- *and $P \subset \mathcal{O}$ consisting of all possible objects outside G; i.e., there exist data instances where these are in the top-k for Q and $\mathcal{C}$.*

Outline

Beyond the most informative top-k answer

Estimating the most likely top-k answer:

Beyond the most informative top-k answer

Estimating the most likely top-k answer:

In the example: $P = \{o2 \in [1.042, 1.105],\ o5 \in [1.000, 1.050]\}$.

Beyond the most informative top-k answer

Estimating the most likely top-k answer:

In the example: $P = \{o2 \in [1.042, 1.105],\ o5 \in [1.000, 1.050]\}$.

If we assume a uniform distribution in the intervals:

$$\mathbf{P}[o2 \geq o5] = 0.989$$
$$\mathbf{P}[o5 > o2] = 0.011$$

Beyond the most informative top-k answer

Estimating the most likely top-k answer:

In the example: $P = \{o2 \in [1.042, 1.105],\ o5 \in [1.000, 1.050]\}$.

If we assume a uniform distribution in the intervals:

$$\begin{aligned}\mathbf{P}[o2 \geq o5] =& 0.989\\ \mathbf{P}[o5 > o2] =& 0.011\end{aligned}$$

$\implies$ the most likely top-k is $G \cup \{o2\}$: $\mathbf{P}[\{o4, o2\}] = 0.989$

Beyond the most informative top-k answer

Estimating the most likely top-k answer:

In the example: $P = \{o2 \in [1.042, 1.105],\ o5 \in [1.000, 1.050]\}$.

If we assume a uniform distribution in the intervals:

$$\mathbf{P}[o2 \geq o5] = 0.989$$
$$\mathbf{P}[o5 > o2] = 0.011$$

$\implies$ the most likely top-k is $G \cup \{o2\}$: $\mathbf{P}[\{o4, o2\}] = 0.989$

Ways to evaluate:

- naive enumeration: good if $|P|$ is small,
- sampling or probabilistic top-k [Soliman et. al, VLDBJ10]

View selection

The P and G sets might be too expensive to compute, if the view set is very large, even using early-termination algorithms.

Solution: select few most relevant views, i.e., a subset $\tilde{\mathcal{V}} \subset \mathcal{V}$

- based on view definition, result statistics (see paper)

View selection

The P and G sets might be too expensive to compute, if the view set is very large, even using early-termination algorithms.

Solution: select few most relevant views, i.e., a subset $\tilde{\mathcal{V}} \subset \mathcal{V}$

- based on view definition, result statistics (see paper)
- trade-off between size of $\tilde{\mathcal{V}}$ and "quality" of the resulting $(\tilde{G}, \tilde{P})$ pair, in terms of distance to (G, P):

$$\Delta = \binom{|\tilde{P}|}{k - |\tilde{G}|} - \binom{|P|}{k - |G|}$$

View selection

The P and G sets might be too expensive to compute, if the view set is very large, even using early-termination algorithms.

Solution: select few most relevant views, i.e., a subset $\tilde{\mathcal{V}} \subset \mathcal{V}$

- based on view definition, result statistics (see paper)
- trade-off between size of $\tilde{\mathcal{V}}$ and "quality" of the resulting $(\tilde{G}, \tilde{P})$ pair, in terms of distance to (G, P):

$$\Delta = \binom{|\tilde{P}|}{k - |\tilde{G}|} - \binom{|P|}{k - |G|}$$

Final refinement: compute tightest bounds only for objects in $\tilde{G} \cup \tilde{P}$

Formal results

Instance optimality: For $A_i \in A$ and $A_2 \in A$, write $A_1 \preceq A_1$ iff for all sets of views $\mathcal{V}$ and all data instance $\mathbf{D}$, A_2 costs at least as much as A_1.

Lemma

$\mathrm{SR\text{-}NRA}^{sel} \not\preceq \mathrm{SR\text{-}NRA}^{nosel} \not\preceq \mathrm{SR\text{-}NRA}^{sel}$.

$\mathrm{SR\text{-}TA}^{sel} \not\preceq \mathrm{SR\text{-}TA}^{nosel} \not\preceq \mathrm{SR\text{-}TA}^{sel}$.

Theorem

When we restrict the class of views to pairwise disjoint views:

- $\mathrm{SR\text{-}TA}^{sel}$ *is instance optimal over* $\mathbf{A}$.
- $\mathrm{SR\text{-}NRA}^{sel}$ *is instance optimal over* $\mathbf{A}$ *(when only sequential accesses are allowed).*

Putting it all together

ProcessQueryUsingViews$(\boldsymbol{\mathcal{V}}, Q, \boldsymbol{\mathcal{C}}, k)$

Require: query Q, views $\mathcal{V}$, context $\mathcal{C}$, top k required

for $V \in \mathcal{V}$ **do**
    transpose the context $\mathcal{C}^V$ to $\mathcal{C}$
end for
$\tilde{\mathcal{V}} \leftarrow$ view selection on $\mathcal{V}$ for Q
$(\tilde{G}, \tilde{P}) \leftarrow \mathrm{SR\text{-}TA}(Q, k, \tilde{\mathcal{V}})$ or $\mathrm{SR\text{-}NRA}(Q, k, \tilde{\mathcal{V}})$
$(G, P) \leftarrow \textsc{Refine}(\tilde{G}, \tilde{P})$
$E = \textsc{Estimate}(P, k - |G|)$
return $G \cup E$

Outline

Context-aware top-k retrieval

Uncertainty in views

View-based top-k processing

Refinements

Experiments

Experiments: location-aware search

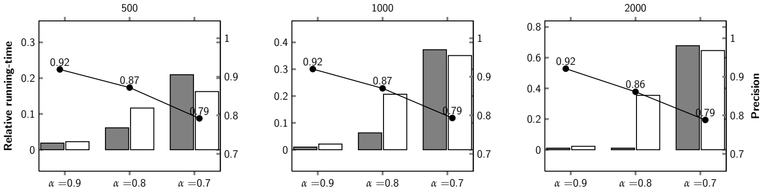


Figure : Performance and precision of $\mathrm{SR\text{-}TA}^{sel}$ versus exact early-termination algorithm (IR-TREE) (grey=top-10, white=top-20).

- PolyBot dataset: 6,115,264 objects and 1,876 attributes
- Views: 20 views of 2-term queries at 5 random locations, various list sizes
- Test: 10 queries at 5 locations and $\alpha \in \{0.7, 0.8, 0.9\}$

Experiments: social-aware search

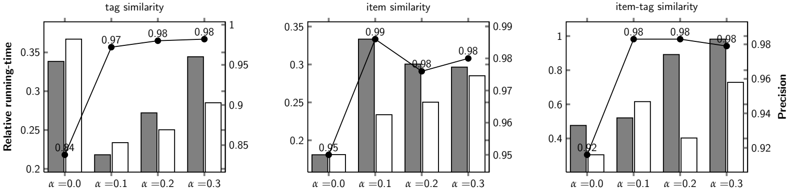


Figure : Social-aware search: performance and precision of SR-TA^{sel} versus ContextMerge(grey=top-10, white=top-20).

- Delicious data: 80000 users, 595811 objects, 198080 attributes
- Social network: 3 similarity networks (tag, item, item-tag)
- Views: 10 users each having 40 views of 1 and 2 tag queries
- Test: 10 3-tag queries for 5 seekers and $\alpha \in \{0, 0.1, 0.2, 0.3\}$

Summary

We formalize and study the problem of context-aware top-k processing based on (possibly uncertain) views.

- Context transposition, exemplified in two application scenarios
- New semantics based on views: most informative result
- Sound and complete adaptation of TA / NRA
- Probabilistic refinement: most likely top-k result
- Further efficiency: view selection
  - instance optimality under restrictions

Thank you.

Threshold algorithms: SR-TA

Adaptation of TA algorithm[Fagin01], SR-NRA similar.

Require: query Q, size k, views $\mathcal{V}$ (after transposition)

$D = \emptyset$
loop
    for each view $V \in \mathcal{V}$ in turn **do**
        $(o_i, wsc_i, bsc_i) \leftarrow$ next tuple by sequential access in V
        read by random-accesses all other lists $V' \in \mathcal{V}$ for tuples (o_j, wsc_j, bsc_j) s.t. $o_i = o_j$
        $wsc \leftarrow$ solution to the MP in Eq. (1) for o_i
        $bsc \leftarrow$ solution to the MP in Eq. (2) for o_i
        add the tuple (o_i, wsc, bsc) to D
    end for
    $\tau \leftarrow$ maximal possible score of objects not encountered
    $wsc_t \leftarrow$ lower-bound score of kth candidate in D
    if $\tau \leq wsc_t$ **then**
        break
    end if
end loop
$(G, P) = \textsc{Partition}(D, k)$
return (G, P)

Threshold algorithms: PARTITION(D, k)

Require: candidate list D, parameter k

$G \leftarrow \emptyset$ *the objects guaranteed to be in the top-k*
$P \leftarrow \emptyset$ *the objects that might enter the top-k*
for each tuple $(o, bsc, wsc) \in D$, $o \neq *$ **do**
    $x \leftarrow |\{(o', bsc', wsc') \in D \mid o' \neq o,\ bsc' > wsc\}|$
    $wsc_t \leftarrow$ lower-bound score of kth candidate in D
    if $x \leq k$ and for $(*, wsc_*, bsc_*) \in D$, $bsc_* \leq wsc$ **then**
        add o to G
    else if $bsc > wsc_t$ **then**
        add o to P
    end if
end for
return G, P

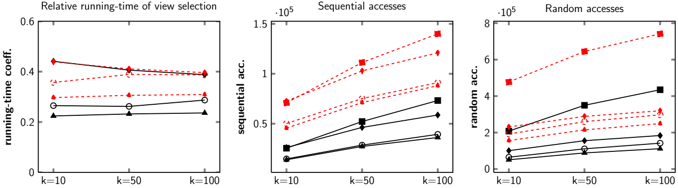
Experiments: context-agnostic setting

Input data:

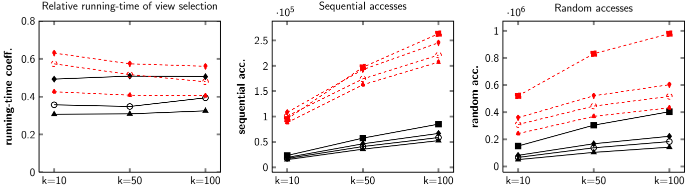
- synthetic: 100,000 objects and 10 attributes, scores in [0,100]
- views: all possible combinations of 2 and 3 attributes
- uncertain data: replace each score with a score range (Gaussian distribution, $\sigma \in \{5, 10\}$)

Test: 100 randomly-generated queries of 5 attributes

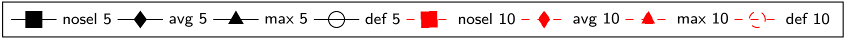
Experiments: context-agnostic setting



Uniform distribution



Exponential distribution



Experiments: context-agnostic setting

Sel. + Dist.	**Rel. running-time**			**Min. precision**			$\|P\|$		
	10	50	100	10	50	100	10	50	100
`avg` + uni	0.576	0.676	0.712	0.57	0.69	0.72	10	36	64
`def` + uni	0.350	0.446	0.544	0.57	0.69	0.72	10	36	64
`max` + uni	0.296	0.395	0.446	0.57	0.69	0.72	10	36	64
`avg` + exp	0.732	1.128	1.287	0.60	0.63	0.64	10	46	86
`def` + exp	0.531	0.771	1.003	0.60	0.63	0.64	10	46	86
`max` + exp	0.456	0.684	0.827	0.60	0.63	0.64	10	46	86

Table : Comparison between SR-TA and TA (exact scores), for uniform and exponential distributions, for std 5.