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 ($\sigma$) (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\}
⇝ a social-aware result: D4, D2
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:

\[
\text{score}(o, t \mid \text{loc}, \alpha) = \alpha \times \text{tf}(t, o) + (1 - \alpha) \times \text{dist}(o, \text{loc})
\]

Problem (location-aware top-k retrieval)

Given a query \( Q = \{t_1, \ldots, t_n\} \), a context (e.g., location and \( \alpha \)), 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 \( o2:0.85 \)
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 $\sim$ score uncertainty, approximate top-k results.
Outline

Context-aware top-k retrieval

Uncertainty in views

View-based top-k processing

Refinements

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

The context $C^V$ of a view $V$ is a pair $(C^V.l, C^V.\alpha)$:
- **location** $C^V.l$: geo-coordinates or seeker Id in a social network
- **contextual parameter** $C^V.\alpha$: the weight of the context in scores
Focus on two applications: location-aware search, social-aware search

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

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

**Transposition:** adapt results for $(C^V.l, C^V.\alpha)$ to a new context $(C.l, C.\alpha)$
Context transposition yields uncertainty

$L0: Q$
$L1: V1$
$L2: V2, V3$

$V1=(L1, \{t1,t2\})$
$V2=(L2, \{t1\})$
$V3=(L2, \{t2\})$

<table>
<thead>
<tr>
<th></th>
<th>o</th>
<th>sc</th>
</tr>
</thead>
<tbody>
<tr>
<td>o5</td>
<td>1.062</td>
<td></td>
</tr>
<tr>
<td>o4</td>
<td>1.029</td>
<td></td>
</tr>
<tr>
<td>o2</td>
<td>1.000</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>o</th>
<th>sc</th>
</tr>
</thead>
<tbody>
<tr>
<td>o2</td>
<td>0.946</td>
<td></td>
</tr>
<tr>
<td>o3</td>
<td>0.575</td>
<td></td>
</tr>
<tr>
<td>o5</td>
<td>0.450</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>o</th>
<th>sc</th>
</tr>
</thead>
<tbody>
<tr>
<td>o4</td>
<td>0.962</td>
<td></td>
</tr>
<tr>
<td>o5</td>
<td>0.450</td>
<td></td>
</tr>
<tr>
<td>o1</td>
<td>0.437</td>
<td></td>
</tr>
</tbody>
</table>

Top-2 query $Q=\{t1,t2\}$ at location $L0$
Context transposition yields uncertainty

\[ V_1 = (L_1, \{t1, t2\}) \]
\[ V_2 = (L_2, \{t1\}) \]
\[ V_3 = (L_2, \{t2\}) \]

<table>
<thead>
<tr>
<th>o</th>
<th>sc</th>
<th>o</th>
<th>sc</th>
<th>o</th>
<th>sc</th>
</tr>
</thead>
<tbody>
<tr>
<td>o5</td>
<td>1.062</td>
<td>o2</td>
<td>0.946</td>
<td>o4</td>
<td>0.962</td>
</tr>
<tr>
<td>o4</td>
<td>1.029</td>
<td>o3</td>
<td>0.575</td>
<td>o5</td>
<td>0.450</td>
</tr>
<tr>
<td>o2</td>
<td>1.000</td>
<td>o5</td>
<td>0.450</td>
<td>o1</td>
<td>0.437</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o4</td>
<td>0.262</td>
<td>o2</td>
<td>0.246</td>
</tr>
</tbody>
</table>

Top-2 query \( Q = \{t1, t2\} \) at location \( L_0 \)
Context transposition yields uncertainty

\[ V_1 = (L_1, \{t_1, t_2\}) \]
\[ V_2 = (L_2, \{t_1\}) \]
\[ V_3 = (L_2, \{t_2\}) \]

\[
\begin{array}{c|c}
 o & sc \\
\hline
 o_5 & 1.062 \\
 o_4 & 1.029 \\
 o_2 & 1.000 \\
\end{array}
\]

\[
\begin{array}{c|c}
 o & sc \\
\hline
 o_2 & 0.946 \\
 o_3 & 0.575 \\
 o_5 & 0.450 \\
 o_4 & 0.262 \\
\end{array}
\]

\[
\begin{array}{c|c}
 o & sc \\
\hline
 o_4 & 0.962 \\
 o_5 & 0.450 \\
 o_1 & 0.437 \\
 o_2 & 0.246 \\
\end{array}
\]

Distance of \( o_4 \) to \( Q \) unknown, but within \([0.987, 1.037]\) interval.
Context transposition yields uncertainty

\[ V_1 = (L_1, \{t_1, t_2\}) \quad V_2 = (L_2, \{t_1\}) \quad V_3 = (L_2, \{t_2\}) \]

<table>
<thead>
<tr>
<th></th>
<th>o</th>
<th>sc</th>
<th></th>
<th>o</th>
<th>sc</th>
<th></th>
<th>o</th>
<th>sc</th>
</tr>
</thead>
<tbody>
<tr>
<td>o5</td>
<td>1.062</td>
<td></td>
<td>o2</td>
<td>0.946</td>
<td></td>
<td>o4</td>
<td>0.962</td>
<td></td>
</tr>
<tr>
<td>o4</td>
<td>1.029</td>
<td></td>
<td>o3</td>
<td>0.575</td>
<td></td>
<td>o5</td>
<td>0.450</td>
<td></td>
</tr>
<tr>
<td>o2</td>
<td>1.000</td>
<td></td>
<td>o5</td>
<td>0.450</td>
<td></td>
<td>o1</td>
<td>0.437</td>
<td></td>
</tr>
<tr>
<td></td>
<td>o4</td>
<td>0.262</td>
<td>o2</td>
<td>0.246</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

distance of o4 to Q unknown, but within [0.987, 1.037] interval
Context transposition yields uncertainty

\[ V_1 = (L_1, \{t_1, t_2\}) \quad V_2 = (L_2, \{t_1\}) \quad V_3 = (L_2, \{t_2\}) \]

<table>
<thead>
<tr>
<th></th>
<th>o</th>
<th>sc</th>
<th></th>
<th>o</th>
<th>sc</th>
<th></th>
<th>o</th>
<th>sc</th>
</tr>
</thead>
<tbody>
<tr>
<td>o5</td>
<td>1.062</td>
<td></td>
<td>o2</td>
<td>0.946</td>
<td></td>
<td>o4</td>
<td>0.962</td>
<td></td>
</tr>
<tr>
<td>o4</td>
<td>1.029</td>
<td></td>
<td>o3</td>
<td>0.575</td>
<td></td>
<td>o5</td>
<td>0.450</td>
<td></td>
</tr>
<tr>
<td>o2</td>
<td>1.000</td>
<td></td>
<td>o5</td>
<td>0.450</td>
<td></td>
<td>o1</td>
<td>0.437</td>
<td></td>
</tr>
</tbody>
</table>

Distance of \( o_4 \) 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 $C.l \sim C^V.l$ path - worstscore
- other known paths - bestscore
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 $(Q^V, C^V)$
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:

\[
\begin{align*}
\max & \sum_{t_i \in Q} \text{sc}(o, t_i \mid C) \\
\min & \sum_{t_i \in Q} \text{sc}(o, t_i \mid C) \\
\text{wsc} & \leq \sum_{t_j \in Q^V} \text{sc}(o, t_j \mid C) \quad \forall V \in \mathcal{V} \text{ s.t. } (o, \text{wsc}, \text{bsc}) \in \text{ans}(V) \\
\sum_{t_j \in Q^V} \text{sc}(o, t_j \mid C) & \leq \text{bsc} \quad \forall V \in \mathcal{V} \text{ s.t. } (o, \text{wsc}, \text{bsc}) \in \text{ans}(V) \\
\text{sc}(o, t_l \mid C) & \geq 0, \forall t_l \in \mathcal{T}
\end{align*}
\]
Before context transposition

\[ V_1 = (L_1, \{t_1, t_2\}) \]
\[ V_2 = (L_2, \{t_1\}) \]
\[ V_3 = (L_2, \{t_2\}) \]

<table>
<thead>
<tr>
<th>o</th>
<th>sc</th>
<th>o</th>
<th>sc</th>
<th>o</th>
<th>sc</th>
</tr>
</thead>
<tbody>
<tr>
<td>o5</td>
<td>1.062</td>
<td>o2</td>
<td>0.946</td>
<td>o4</td>
<td>0.962</td>
</tr>
<tr>
<td>o4</td>
<td>1.029</td>
<td>o3</td>
<td>0.575</td>
<td>o5</td>
<td>0.450</td>
</tr>
<tr>
<td>o2</td>
<td>1.000</td>
<td>o5</td>
<td>0.450</td>
<td>o1</td>
<td>0.437</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o4</td>
<td>0.262</td>
<td>o2</td>
<td>0.246</td>
</tr>
</tbody>
</table>

Top-2 query \( Q = \{t_1, t_2\} \) at location \( L_0 \)
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 $o5$:

$$\max \quad \text{sc}(o5, t1 \mid C) + \ 	ext{sc}(o5, t2 \mid C)$$
$$\min \quad \text{sc}(o5, t1 \mid C) + \ 	ext{sc}(o5, t2 \mid C)$$

$$0.957 \leq \text{sc}(o5, t1 \mid C) + \text{sc}(o5, t2 \mid C) \leq 1.167 \quad (V1)$$
$$0.500 \leq \text{sc}(o5, t1 \mid C) \leq 0.525 \quad (V2)$$
$$0.500 \leq \text{sc}(o5, t2 \mid C) \leq 0.525 \quad (V3)$$
Using views for one object: example

Top-\(k\) using views with uncertain scores:

LP formulation to compute tightest bounds - e.g., for \(o_5\):

\[
\begin{align*}
\text{max} & \quad \text{sc}(o_5, t_1 \mid C) + \text{sc}(o_5, t_2 \mid C) \\
\text{min} & \quad \text{sc}(o_5, t_1 \mid C) + \text{sc}(o_5, t_2 \mid C) \\
0.957 & \leq \text{sc}(o_5, t_1 \mid C) + \text{sc}(o_5, t_2 \mid C) \leq 1.167 \\
0.500 & \leq \text{sc}(o_5, t_1 \mid C) \leq 0.525 \\
0.500 & \leq \text{sc}(o_5, t_2 \mid C) \leq 0.525
\end{align*}
\]

\[\Rightarrow\] score interval for \(o_5\) 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 \(\rightsquigarrow\) the SR-TA and SR-NRA algorithms.
Our approach for top-$k$ using views

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

Plug the LPs in:

- the computation of worst-score/ best-score bounds,
- the computation of the termination threshold.
In some cases, the exact top-\(k\) cannot be extracted with full confidence.

In our running example, at termination:

<table>
<thead>
<tr>
<th>Candidates</th>
<th>obj</th>
<th>wsc</th>
<th>bsc</th>
</tr>
</thead>
<tbody>
<tr>
<td>o4</td>
<td>1.174</td>
<td>1.134</td>
<td></td>
</tr>
<tr>
<td>o2</td>
<td>1.042</td>
<td>1.105</td>
<td></td>
</tr>
<tr>
<td>o5</td>
<td>1.000</td>
<td>1.050</td>
<td></td>
</tr>
<tr>
<td>o3</td>
<td>0.500</td>
<td>0.971</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>0</td>
<td>0.849</td>
<td></td>
</tr>
</tbody>
</table>
In some cases, the exact top-\( k \) cannot be extracted with full confidence.

In our running example, at termination:

<table>
<thead>
<tr>
<th>Candidates</th>
<th>obj</th>
<th>wsc</th>
<th>bsc</th>
</tr>
</thead>
<tbody>
<tr>
<td>o4</td>
<td>1.174</td>
<td>1.134</td>
<td></td>
</tr>
<tr>
<td>o2</td>
<td>1.042</td>
<td>1.105</td>
<td></td>
</tr>
<tr>
<td>o5</td>
<td>1.000</td>
<td>1.050</td>
<td></td>
</tr>
<tr>
<td>o3</td>
<td>0.500</td>
<td>0.971</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>0</td>
<td>0.849</td>
<td></td>
</tr>
</tbody>
</table>

- 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:

<table>
<thead>
<tr>
<th>Candidates</th>
<th>obj</th>
<th>wsc</th>
<th>bsc</th>
</tr>
</thead>
<tbody>
<tr>
<td>o4</td>
<td>1.174</td>
<td>1.134</td>
<td></td>
</tr>
<tr>
<td>o2</td>
<td>1.042</td>
<td>1.105</td>
<td></td>
</tr>
<tr>
<td>o5</td>
<td>1.000</td>
<td>1.050</td>
<td></td>
</tr>
<tr>
<td>o3</td>
<td>0.500</td>
<td>0.971</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>0</td>
<td>0.849</td>
<td></td>
</tr>
</tbody>
</table>

- 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:

<table>
<thead>
<tr>
<th>Candidates</th>
<th>wsc</th>
<th>bsc</th>
</tr>
</thead>
<tbody>
<tr>
<td>obj</td>
<td></td>
<td></td>
</tr>
<tr>
<td>o4</td>
<td>1.174</td>
<td>1.134</td>
</tr>
<tr>
<td>o2</td>
<td>1.042</td>
<td>1.105</td>
</tr>
<tr>
<td>o5</td>
<td>1.000</td>
<td>1.050</td>
</tr>
<tr>
<td>o3</td>
<td>0.500</td>
<td>0.971</td>
</tr>
<tr>
<td>*</td>
<td>0</td>
<td>0.849</td>
</tr>
</tbody>
</table>

- 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-\textit{k} using uncertain views

Problem (Top-\textit{k} retrieval using uncertain views)

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

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

Context-aware top-k retrieval

Uncertainty in views

View-based top-k processing

Refinements

Experiments
Beyond the most informative top-\(k\) answer

Estimating the most likely top-\(k\) answer:

In the example: \(P = \{o_2 \in [1.042, 1.105], o_5 \in [1.000, 1.050]\}\).

If we assume a uniform distribution in the intervals:

\[ P[o_2 \geq o_5] = 0.989 \]
\[ P[o_5 > o_2] = 0.011 \]

\(\Rightarrow\) the most likely top-\(k\) is \(G \cup \{o_2\}\).

Ways to evaluate:

- naive enumeration: good if \(|P|\) is small,
- sampling or probabilistic top-\(k\) [Soliman et al, VLDBJ10]
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: $P = \{o_2 \in [1.042, 1.105], \ o_5 \in [1.000, 1.050]\}$.

If we assume a uniform distribution in the intervals:

\[
\begin{align*}
P[o_2 \geq o_5] &= 0.989 \\
P[o_5 > o_2] &= 0.011
\end{align*}
\]
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:

\[
P[o2 \geq o5] = 0.989 \\
P[o5 > o2] = 0.011
\]

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

\[
\begin{align*}
P[o2 \geq o5] &= 0.989 \\
P[o5 > o2] &= 0.011
\end{align*}
\]

\[\implies \text{the most likely top-}\(k\) \text{ is } G \cup \{o2\}: \ 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 $\tilde{V} \subset V$

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

- trade-off between size of $\tilde{V}$ and “quality” of the resulting $(\tilde{G}, \tilde{P})$ pair, in terms of distance to $(G, P)$:

  $$\Delta = \left( \frac{|\tilde{P}|}{k - |\tilde{G}|} \right) - \left( \frac{|P|}{k - |G|} \right)$$

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

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

$$\Delta = \left( \frac{|\tilde{P}|}{k - |\tilde{G}|} \right) - \left( \frac{|P|}{k - |G|} \right)$$

**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

\[
\begin{align*}
\text{SR-NRA}^{\text{sel}} & \preceq \text{SR-NRA}^{\text{nosel}} & \preceq \text{SR-NRA}^{\text{sel}}. \\
\text{SR-TA}^{\text{sel}} & \preceq \text{SR-TA}^{\text{nosel}} & \preceq \text{SR-TA}^{\text{sel}}.
\end{align*}
\]

Theorem

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

- SR-TA\(^{sel}\) is instance optimal over \( \mathbf{A} \).
- SR-NRA\(^{sel}\) is instance optimal over \( \mathbf{A} \) (when only sequential accesses are allowed).
Putting it all together

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

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

\begin{enumerate}
\item for \(V \in \mathcal{V}\) do
\item transpose the context \(C^V\) to \(C\)
\item end for
\item \(\tilde{V}\) \leftarrow\text{view selection on} \(\mathcal{V}\) for \(Q\)
\item \((\tilde{G}, \tilde{P}) \leftarrow \text{SR-TA}(Q, k, \tilde{V})\) or \(\text{SR-NRA}(Q, k, \tilde{V})\)
\item \((G, P) \leftarrow \text{Refine}(\tilde{G}, \tilde{P})\)
\item \(E = \text{Estimate}(P, k - |G|)\)
\item return \(G \cup E\)
\end{enumerate}
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 $\text{SR-TA}^{sel}$ versus exact early-termination algorithm ($\text{IR-TREE}$) (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\textsuperscript{sel} versus \textsc{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 $\alpha \in \{0, 0.1, 0.2, 0.3\}$
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)

1: $D = \emptyset$
2: **loop**
3: for each view $V \in \mathcal{V}$ in turn do
4: $(o_i, wsc_i, bsc_i) \leftarrow$ next tuple by sequential access in $V$
5: read by random-accesses all other lists $V' \in \mathcal{V}$ for tuples $(o_j, wsc_j, bsc_j)$ s.t. $o_i = o_j$
6: $wsc \leftarrow$ solution to the MP in Eq. (1) for $o_i$
7: $bsc \leftarrow$ solution to the MP in Eq. (2) for $o_i$
8: add the tuple $(o_i, wsc, bsc)$ to $D$
9: end for
10: $\tau \leftarrow$ maximal possible score of objects not encountered
11: $wsc_t \leftarrow$ lower-bound score of $k$th candidate in $D$
12: if $\tau \leq wsc_t$ then
13: break
14: end if
15: end loop
16: $(G, P) = \text{Partition}(D, k)$
17: return $(G, P)$
Threshold algorithms: \texttt{Partition}(D, k)

\textbf{Require:} candidate list $D$, parameter $k$

1: $G \leftarrow \emptyset$ the objects guaranteed to be in the top-$k$
2: $P \leftarrow \emptyset$ the objects that might enter the top-$k$
3: \textbf{for} each tuple $(o, bsc, wsc) \in D$, $o \neq \ast$ \textbf{do}
4: \hspace{1em} $x \leftarrow |\{(o', bsc', wsc') \in D | o' \neq o, bsc' > wsc\}|$
5: \hspace{1em} $wsc_t \leftarrow$ lower-bound score of $k$th candidate in $D$
6: \hspace{1em} \textbf{if} $x \leq k$ and for $(\ast, wsc_\ast, bsc_\ast) \in D$, $bsc_\ast \leq wsc$ \textbf{then}
7: \hspace{2em} add $o$ to $G$
8: \hspace{1em} \textbf{else if} $bsc > wsc_t$ \textbf{then}
9: \hspace{2em} add $o$ to $P$
10: \hspace{1em} \textbf{end if}
11: \textbf{end for}
12: \textbf{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

Relative running-time of view selection

Sequential accesses

Random accesses

Uniform distribution

Exponential distribution

 nosel 5  avg 5  max 5  def 5  nosel 10  avg 10  max 10  def 10
### 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.