On Pruning for Top-k Ranking in Uncertain Databases

Chonghai Wang, Li Yan Yuan, Jia-Huai You, Osmar R. Zaiane
University of Alberta, Canada

Jian Pei
Simon Fraser University, Canada

August 23, 2011

Outline

- Background
- A new representation of PRF^{ω}
- A general upper bound method
- Pruning for PRF^{ω}
- Pruning for PRF^{e}
- Experiments
- Conclusion

Uncertain Databases

- Uncertain databases (also called probabilistic databases) are proposed to deal with uncertainty in a variety of application domains, such as in sensor network and data cleaning
- X-tuple is a data model to describe the exclusive correlations between tuples in uncertain databases
- Possible world semantics: A possible world W is a set of tuples, such that for each generation rule r, W consists of exactly one tuple in r if $Pr(r) = 1$, and zero or one tuple in r if $Pr(r) < 1$.
- The probability of W, denoted by $Pr(W)$, is the product of the membership probabilities of all the tuples in W and all of $Pr(\bar{r})$, for each r where W contains no tuples from it.

An Example of Uncertain Database

	Time	Radar	Model	Plate No	Speed	Prob
t_1	11:45	L1	Honda	X-123	120	1.0
t_2	11:50	L2	Toyota	Y-245	130	0.7
t_3	11:35	L3	Toyota	Y-245	95	0.3
t_4	12:10	L4	Mazda	W-541	90	0.4
t_5	12:25	L5	Mazda	W-541	110	0.6
t_6	12:15	L6	Chevy	L-105	105	0.5
t_7	12:20	L7	Chevy	L-105	85	0.4

The generation rules here are $t_2 \oplus t_3$, $t_4 \oplus t_5$, $t_6 \oplus t_7$, and t_1.

Possible Worlds

World	Prob
$PW^1 = \{t_1, t_2, t_4, t_6\}$	0.14
$PW^2 = \{t_1, t_2, t_4, t_7\}$	0.112
$PW^3 = \{t_1, t_2, t_4\}$	0.028
$PW^4 = \{t_1, t_2, t_5, t_6\}$	0.21
$PW^5 = \{t_1, t_2, t_5, t_7\}$	0.168
$PW^6 = \{t_1, t_2, t_5\}$	0.042
$PW^7 = \{t_1, t_3, t_4, t_6\}$	0.06
$PW^8 = \{t_1, t_3, t_4, t_7\}$	0.048
$PW^9 = \{t_1, t_3, t_4\}$	0.012
$PW^{10} = \{t_1, t_3, t_5, t_6\}$	0.09
$PW^{11} = \{t_1, t_3, t_5, t_7\}$	0.072
$PW^{12} = \{t_1, t_3, t_5\}$	0.018

Top-k Tuple Ranking in Uncertain Databases

Top-k tuples are the best k tuples in an uncertain database.
Two factors influence top-k tuples:

- Tuple scores
- Membership probabilities

Different Semantics of Top-k Tuples

- U-Topk, U-kRanks (Soliman et al. ICDE2007)
- PT-k query answer (Hua et al. SIGMOD2008)
- Expected Rank (Yi et al. TKDE2008)
- Parameterized Ranking Functions (Li et al. VLDB2009)

Parameterized Ranking Function

PRF^{ω}: $\Upsilon(t) = \sum_{W \in PW(t)} \omega(t, \beta_W(t)) \times Pr(W)$

- $PW(t)$ is the set of all the possible worlds containing t
- $\beta_W(t)$ is the position of t in the possible world W
- $\omega(t, i)$ is a weight function

Our restrictions: We restrict $\omega(t, i)$ to $\omega(i)$ and we assume $\omega(i)$ is monotonically non-increasing.

PRF^{e}: If we set $\omega(i) = \alpha^{i}(0 < \alpha < 1)$, PRF^{ω} becomes PRF^{e}.

Algorithms to Find Top-k Tuples for PRF^{ω} and PRF^{e}

For each tuple t in an uncertain database, compute the PRF^{ω} value of t, then pick up the k tuples with highest PRF^{ω} values. Similarly for PRF^{e}.

Question: Is it necessary to compute the PRF^{ω} and PRF^{e} value for every tuple?

We can apply pruning to avoid substantial computation - Assuming we know $\Upsilon(t_1)$, if we know that $\Upsilon(t_2) \leq \Upsilon(t_1) \leq \mathit{threshold}$, then we do not need to compute $\Upsilon(t_2)$.

Basic Idea for Generating Upper Bound

Given an uncertain database T, consider a set of q tuples $Q = \{t_1, ..., t_q\}$ and generation rules $R = \{r_1, ..., r_l\}$ associated with Q, such that every tuple in Q is in some generation rule in R and every $r_i \in R$ contains at least one tuple in Q.

For any $t \in Q$, our interest is to find an upper bound of it. For this, we want to find some real numbers c_i such that

$$\sum_{i=1}^{q} c_i \Upsilon(t_i) \geq 0 \qquad (1)$$

Let the coefficient of t be c. If $c < 0$, (1) can be transformed to

$$\Upsilon(t) \leq \sum_{t_i \in Q, t_i \neq t} -\frac{c_i}{c} \Upsilon(t_i) \qquad (2)$$

That is, the value of $\Upsilon(t)$ cannot be higher than the right hand side of (2), which is thus an upper bound of t.

A New Representation of PRF^{ω}

Let $t_i \in r_d$, for some $r_d \in R$. Consider a tuple set η of size l, such that $t_i \in \eta$ and each tuple in η is from a distinct generation rule in R. We can write it as

$$\{t_{s_1}, t_{s_2}, ..., t_{s_{d-1}}, t_i, t_{s_{d+1}}, ..., t_{s_l}\}$$

where $t_{s_j} \in r_j$.

Denote by Δ_i the set of all such tuple sets.

We divide Δ_i into l sets. Let S_{ij} be the set of tuple sets in Δ_i each of which contains j tuples which have higher scores than t_i.

Cont'd

Let $\eta \in S_{ij}$, and $PW(\eta)$ be the set of all possible worlds containing all the tuples in η. We define

$$\Upsilon_{\eta}(t_i) = \sum_{W \in PW(\eta)} \omega(\beta_W(t_i)) \times Pr(W)$$

For each non-empty S_{ij} and any two tuple sets $\eta_1, \eta_2 \in S_{ij}$, we can prove that

$$\frac{\Upsilon_{\eta_1}(t_i)}{Pr(\eta_1)} = \frac{\Upsilon_{\eta_2}(t_i)}{Pr(\eta_2)}$$

.

For each non-empty S_{ij}, we define the PRF^{ω} value ratio of S_{ij}, denoted as U_{ij}.

$$U_{ij} = \frac{\Upsilon_{\eta}(t_i)}{Pr(\eta)}$$

Cont'd

A new representation of PRF^{ω}:

$$\Upsilon(t_i) = \sum_{j=0}^{l-1} U_{ij} \times Pr(S_{ij}) \tag{3}$$

We can compute all $Pr(S_{ij})$ in $O(ql^2 + ql\tau)$ time, where τ is the maximum number of real tuples involved in a generation rule.

We have the following conclusion:

(i) if $j_1 \leq j_2$ then $U_{ij_1} \geq U_{ij_2}$, and

(ii) if $score(t_{i_1}) \geq score(t_{i_2})$ then $U_{i_1 j} \geq U_{i_2 j}$.

A General Upper Bound Method (I)

For equation (3), we can multiply both sides with a constant c_i to get

$$c_i \Upsilon(t_i) = c_i \sum_{j=0}^{l-1} U_{ij} \times Pr(S_{ij})$$

Then we add all q equations together to get

$$\sum_{i=1}^{q} c_i \Upsilon(t_i) = \sum_{i=1}^{q} \sum_{j=0}^{l-1} c_i \times U_{ij} \times Pr(S_{ij}) \tag{4}$$

A General Upper Bound Method (II)

If we can transform the right hand side of the equation (4) to the following formats:

$$\sum_{k=1}^{m} a_k(U_{i_k j_k} - U_{i'_k j'_k}) \tag{5}$$

or

$$\sum_{k=1}^{m_1} a_k(U_{i_k j_k} - U_{i'_k j'_k}) + \sum_{k'=1}^{m_2} b_{k'} U_{i_{k'} j_{k'}} \tag{6}$$

Then we can get

$$\sum_{i=1}^{q} c_i \Upsilon(t_i) \geq 0$$

so we get

$$\Upsilon(t) \leq \sum_{t_i \in Q, t_i \neq t} -\frac{c_i}{c} \Upsilon(t_i)$$

A General Upper Bound Method (III)

Theorem: Let $Q = \{t_1, ..., t_q\}$. Assume $t \in Q$ and there exists a tuple $s \in Q$ such that $s \neq t$ and $score(s) \geq score(t)$. Then, there exists at least one assignment θ of c_i such that the right hand side of (4) can be transformed to an expression in the form of (5), and if not, to an expression in the form of (6).

Theorem: Let T be an uncertain table, $Q = \{t', t\}$ be a set of tuples from T. The upper bound u of t, induced by any assignment w.r.t. Q, satisfies $u \geq \frac{Pr(t)}{Pr(t')} \Upsilon(t')$.

If we want to improve the upper bound of t, we may consider adding more tuples in Q. When the size of Q becomes larger, we may get better upper bound.

Practical Pruning Method for PRF^{ω}

For any two tuples t_1 and t_2 such that $score(t_1) \geq score(t_2)$

- If they are involved in one generation rule, we have

$$\Upsilon(t_2) \leq \frac{Pr(t_2)}{Pr(t_1)} \Upsilon(t_1)$$

- If they are involved in two different generation rules, we have
  - If $\frac{Pr(S_{10})}{Pr(t_1)} \geq \frac{Pr(S_{20})}{Pr(t_2)}$, we have $\Upsilon(t_2) \leq \frac{Pr(t_2)}{Pr(t_1)} \Upsilon(t_1)$.
  - If $\frac{Pr(S_{10})}{Pr(t_1)} < \frac{Pr(S_{20})}{Pr(t_2)}$ and the weight function is non-negative, we have $\Upsilon(t_2) \leq \frac{Pr(S_{20})}{Pr(S_{10})} \Upsilon(t_1)$. And we can also add one more tuple into Q such that it is possible to get $\Upsilon(t_2) \leq \frac{Pr(t_2)}{Pr(t_1)} \Upsilon(t_1)$.

Pruning for PRF^e

PRF^e is a special case of PRF^ω, it has some special properties.

For any two tuples t_1 and t_2 (score$(t_1) \geq$ score(t_2)), we can get

$$\Upsilon(t_2) \leq \frac{1}{\alpha} \times \frac{1}{Pr(t_1)} \Upsilon(t_1)$$

.

The time complexity for pruning is $O(1)$.

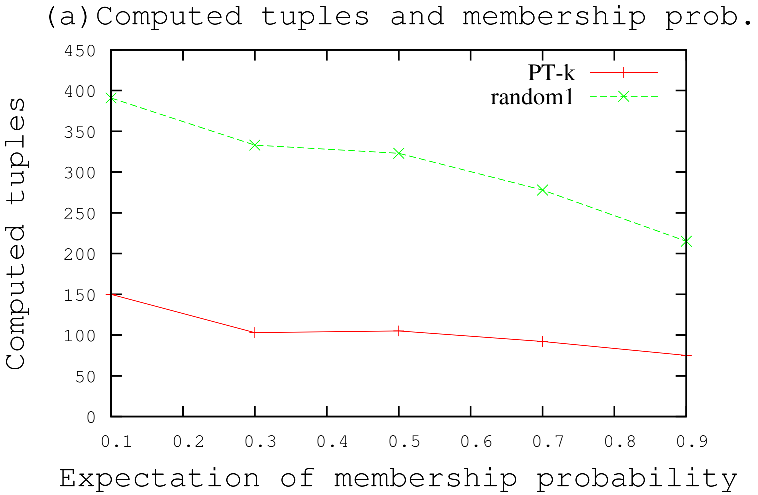
Datasets:

- Normal Datasets: The number of tuples involved in each multi-tuple generation rules follows the normal distribution, so does the probabilities of independent tuple and multi-tuple generation rules
- Special Datasets: The scores of tuples are in a descending order and their membership probabilities are in an ascending order
- Real Dataset: A real data set is generated from International Ice Patrol Iceberg Sighting Datasets

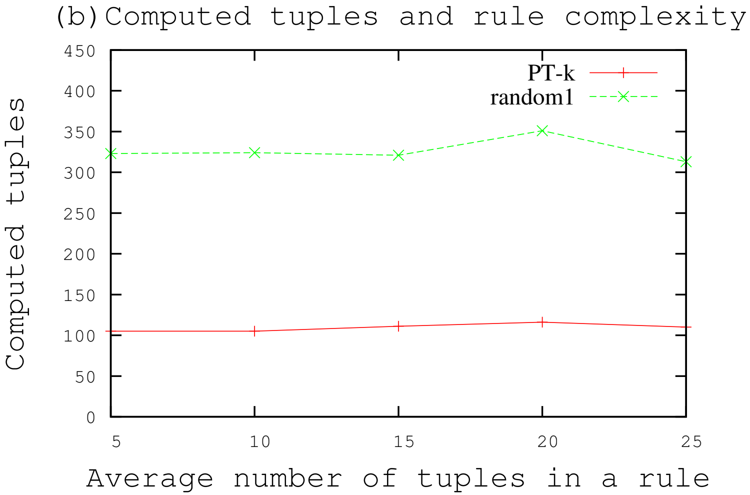
Weight Functions:

- Randomly generated weight functions
- $\omega(i) = n - i$
- PT-k query answer

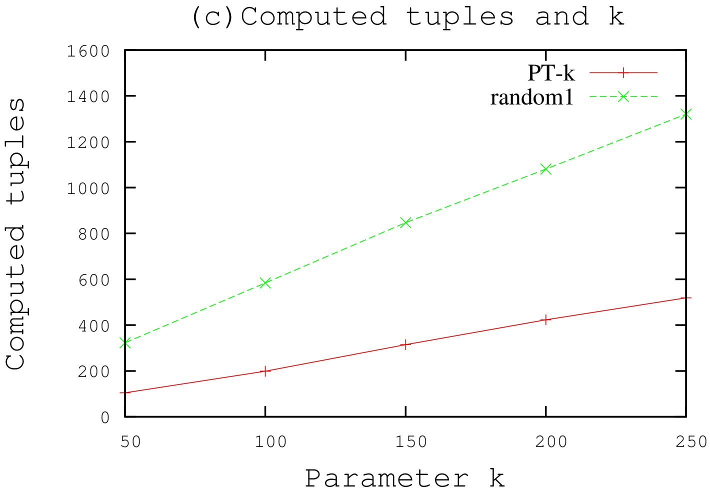
Computed Tuples for PRF^{ω} on Normal Data Sets (I)



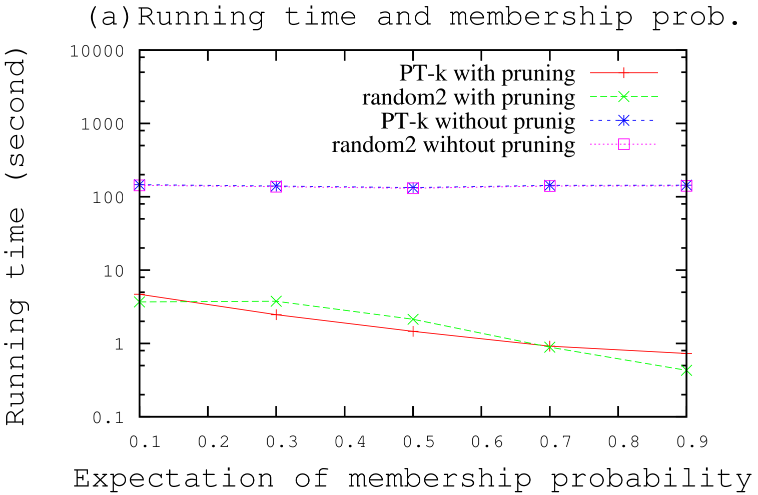
Computed Tuples for PRF^{ω} on Normal Data Sets (II)



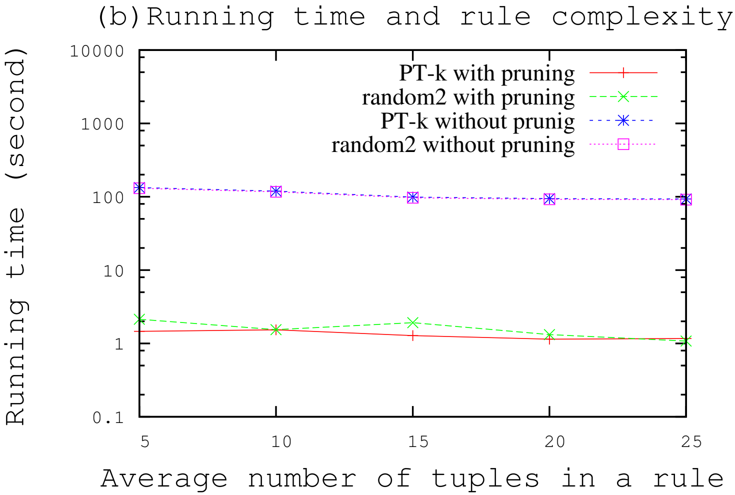
Computed Tuples for PRF^{ω} on Normal Data Sets (III)



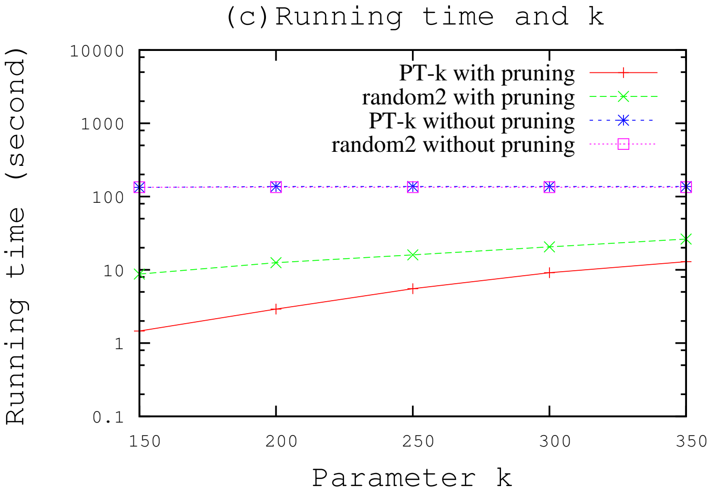
Running Times for PRF^{ω} on Normal Data Sets (I)



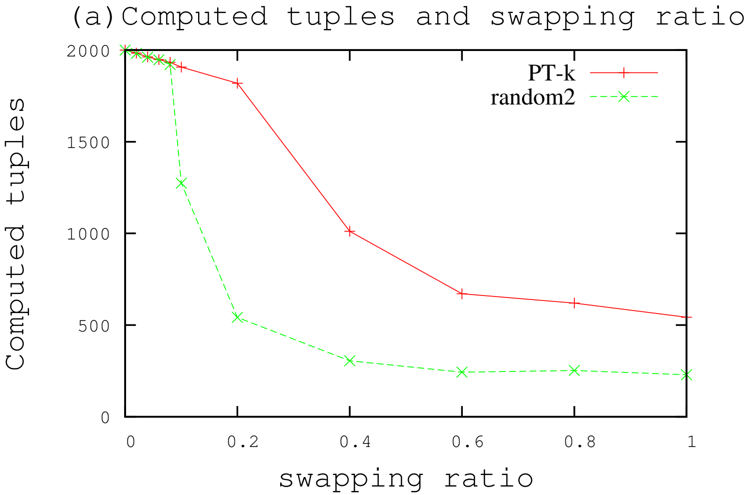
Running Times for PRF^{ω} on Normal Data Sets (II)



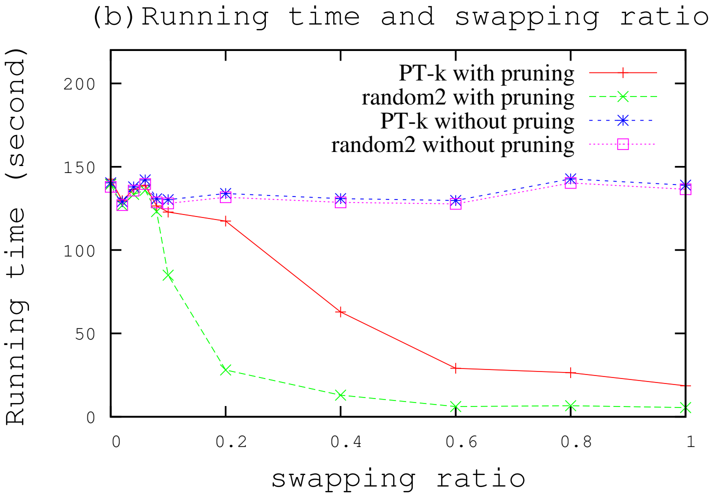
Running Times for PRF^ω on Normal Data Sets (III)



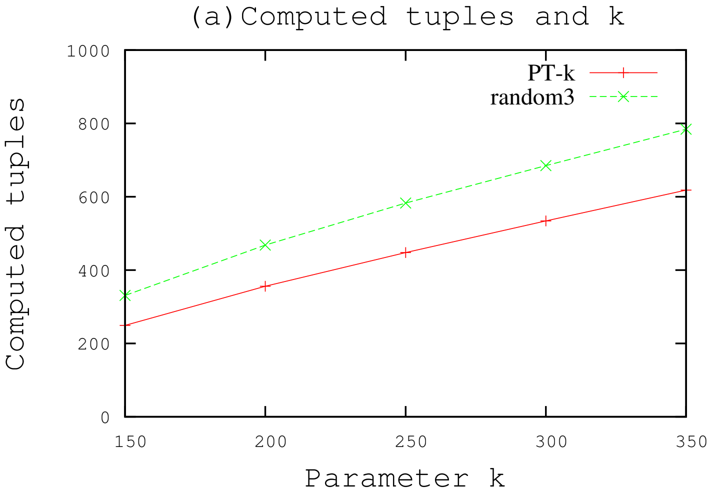
Computed Tuples for PRF^{ω} on Special Data Sets



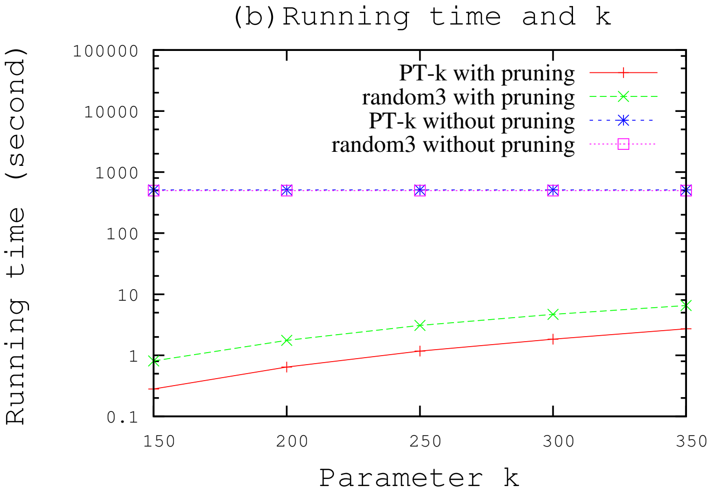
Running Times for PRF^{ω} on Special Data Sets



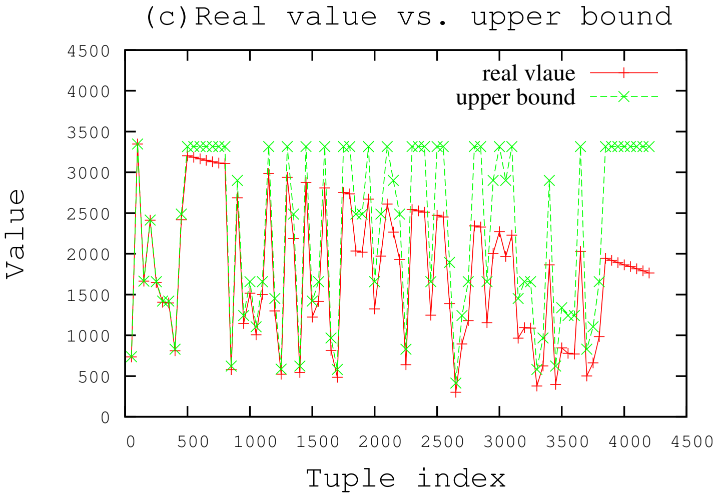
Computed Tuples for PRF^{ω} on Real Data Set



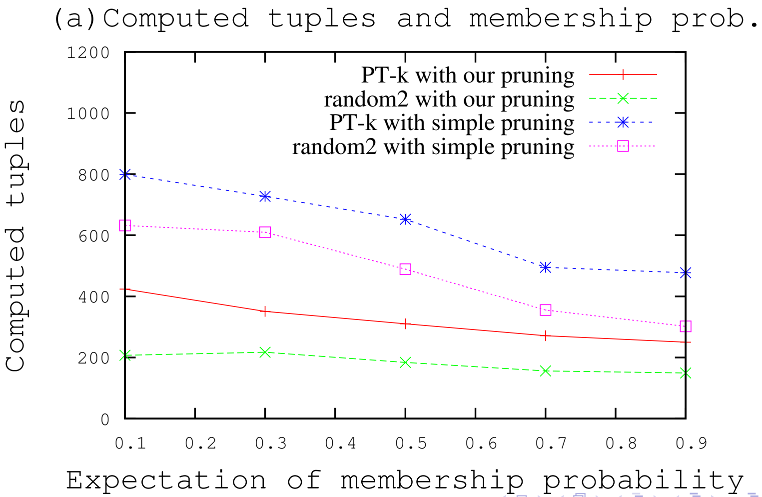
Running Times for PRF^{ω} on Real Data Set



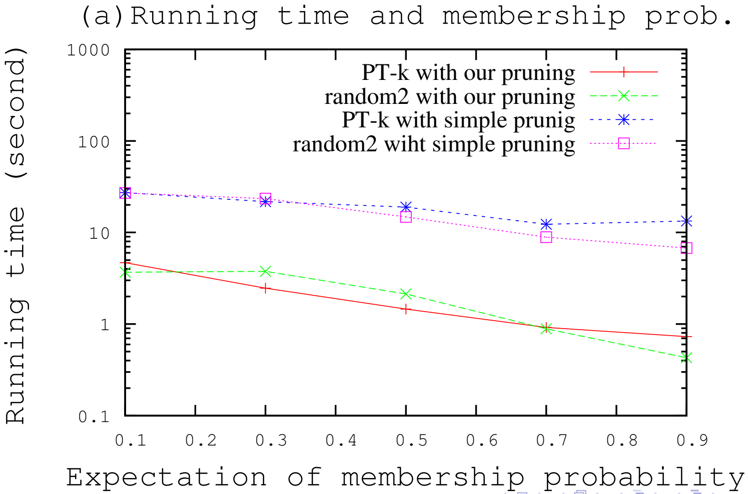
Tightness of upper bounds on Real Data Set



Comparison with (Hua et al. SIGMOD-08): Computed Tuples



Comparison with (Hua et al. SIGMOD-08): Running Times



Conclusion

- We derived a new representation of PRF^ω values
- We formulated a general framework to generate upper bounds of PRF^ω values
- We developed practical pruning methods for computing top-k tuples for PRF^ω
- We derived an early termination condition for PRF^e
- We showed experimentally that our pruning methods generated significant improvements in the computation of top-k tuples