Probabilistic Streaming Tensor Decomposition(POST)

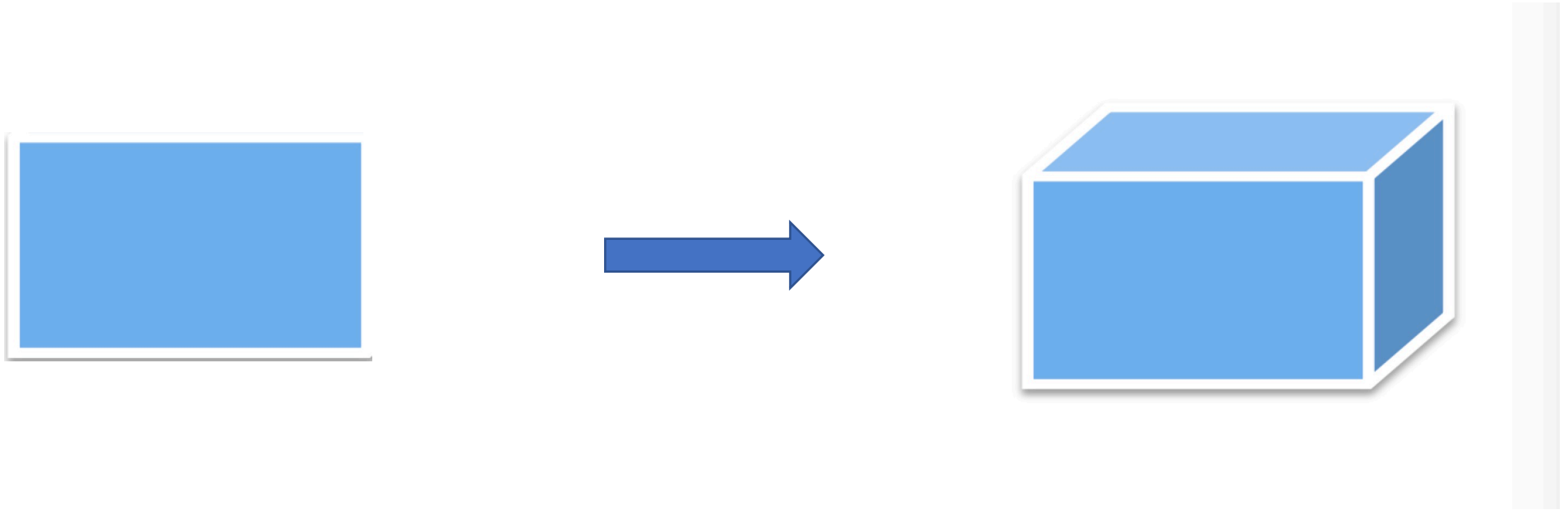
Yishuai Du, Yimin Zheng, Kuang-chih Lee, Shandian Zhe

Presenter: **Yishuai Du**

Overview

- Motivation
- Why POST
- Experiment

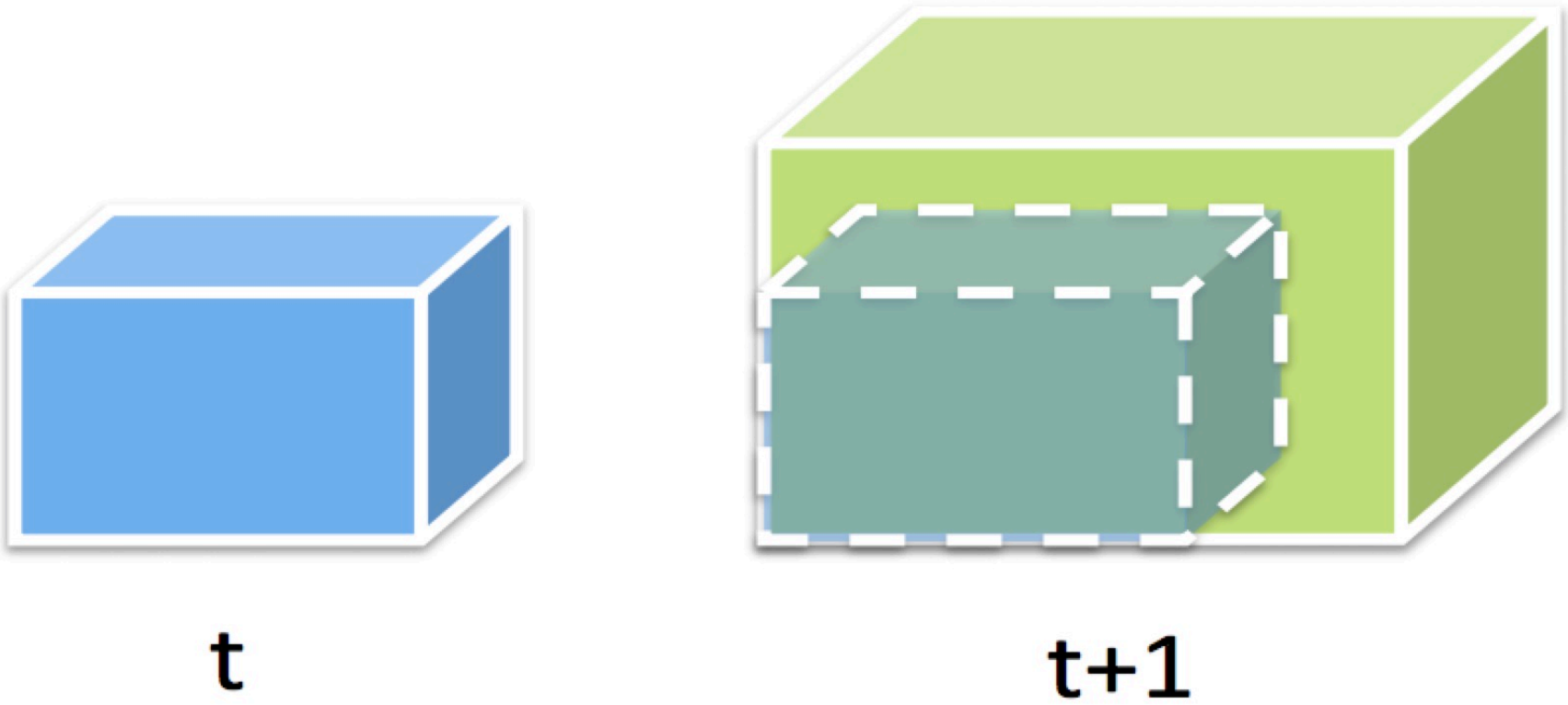
Tensor (static)



- Four-mode tensor (*user, advertisement, page-section, site*)

http://delivery.acm.org/10.1145/3100000/3098007/p435song.pdf?ip=67.164.251.251&id=3098007&acc=OPENTOC&key=4D4702B0C3E38B35%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35%2E054E54E275136550&__acm__=1535162723_46fdcd651e09ed0d16b1d672bbeb7d4

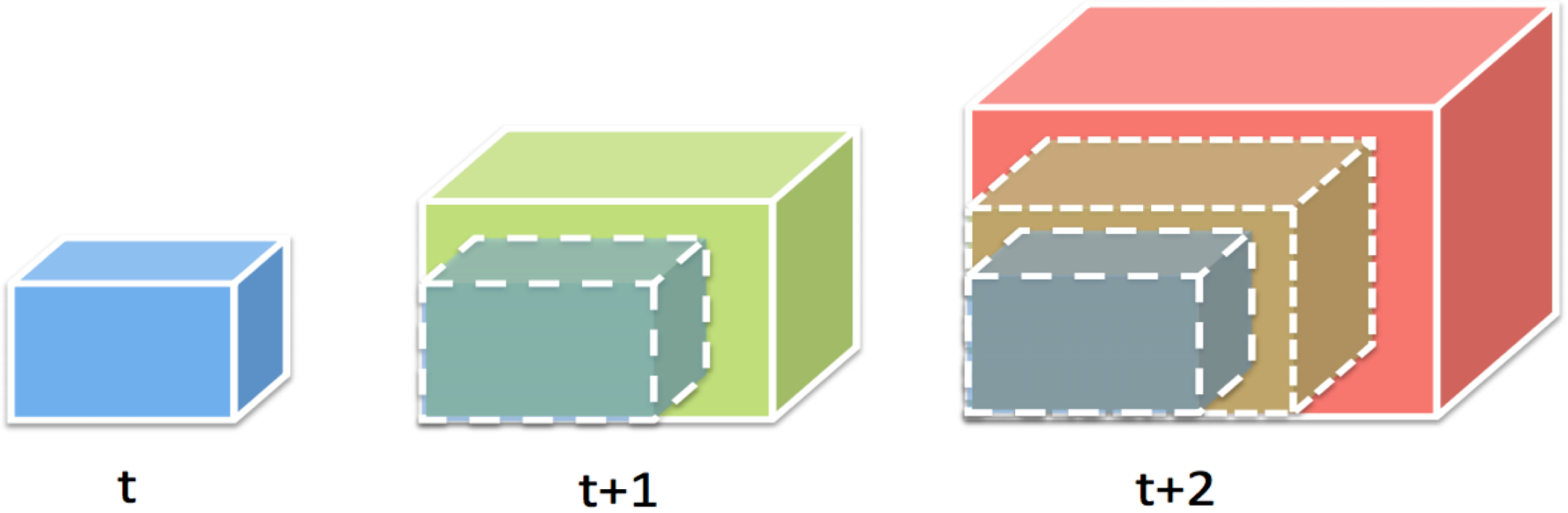
Streaming Tensor (dynamic and incrementally)



http://delivery.acm.org/10.1145/3100000/3098007/p435-song.pdf?ip=67.164.251.251&id=3098007&acc=OPENTOC&key=4D4702B0C3E38B35%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35%2E054E54E275136550&__acm__=1535162723_46fdcd651e09ed0d16b1d672bcbeb7

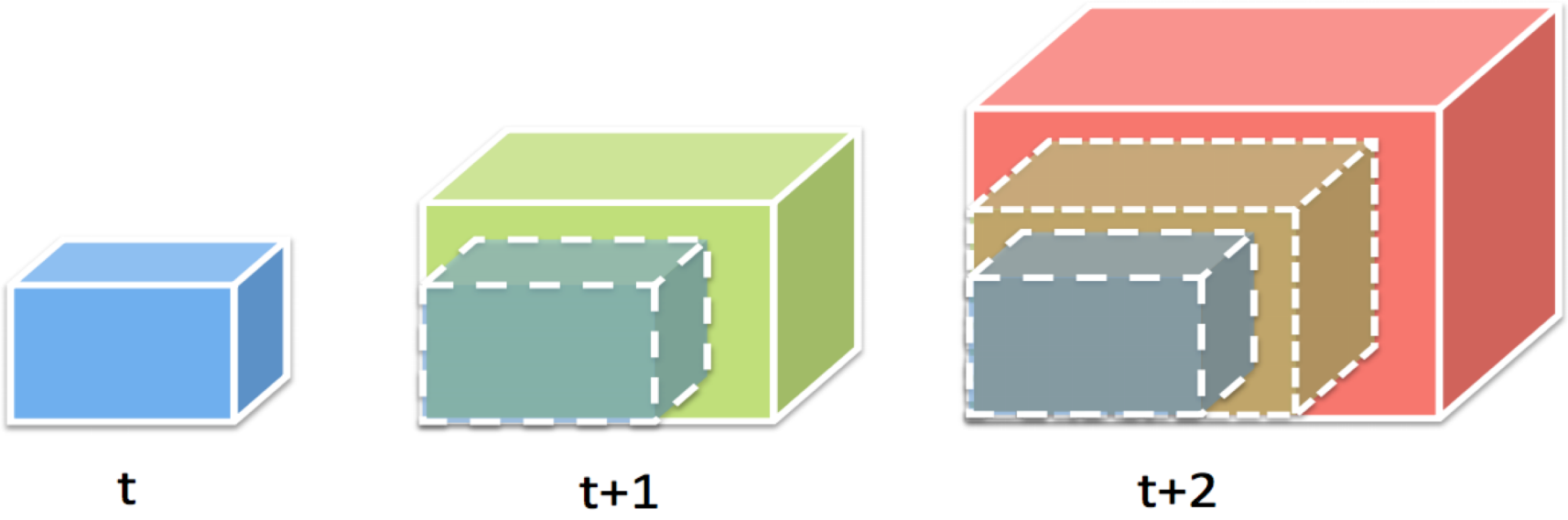
Motivation

- **Multi-Aspect Streaming Tensor Completion(MAST)**, KDD ,2017



MAST

- Limited the order of the streaming entries
- How to address this issue(a). ---- POST



Disadvantage of standard CP

- Unable to evaluate the uncertainty
- Example
- How to address this issue(b) --- POST

Probabilistic Streaming Tensor Decomposition Algorithm

Probabilistic Streaming Tensor Decomposition Algorithm

- Arbitrary orders
- The uncertainty quantification of the embeddings
- More Algorithm

Preliminaries

vectors $\mathbf{u}$

matrices $\mathbf{U}$

K-mode tensor $\mathcal{Y} \in \mathbb{R}^{d_1 \times \ldots \times d_K}$

The entry value (**i** is location) $\mathbf{i} = (i_1, \ldots, i_K)$ is denoted by $y_{\mathbf{i}}$

Embedding vector $\mathbf{u}_j^k$

Embedding matrix $\mathbf{U}^k = [\mathbf{u}_1^{k\top}, \ldots, \mathbf{u}_{d_k}^{k\top}]^\top$

K latent embedding matrices $\mathcal{U} = \{\mathbf{U}^1, \ldots, \mathbf{U}^K\}$

Preliminaries

CP assumes that

$$\mathcal{Y} \approx [\![\mathbf{U}^1, \ldots, \mathbf{U}^K]\!]$$

Kruskal operator

$$\text{vec}(\mathcal{Y}) \approx \sum_{r=1}^{R} \mathbf{U}^1(:, r) \otimes \ldots \otimes \mathbf{U}^K(:, r)$$

Kronecker product

From the vectorized form, we can easily obtain that for each observed entry **i**,

$$y_{\mathbf{i}} \approx \sum_{r=1}^{R} \prod_{k=1}^{K} u^k_{i_k, r} = \mathbf{1}^\top (\mathbf{u}^1_{i_1} \circ \ldots \circ \mathbf{u}^K_{i_K}) \qquad (1)$$

Bayesian model

We first sample the embedding vectors from a Gaussian prior

$$p(\mathcal{U}) = \prod_{k=1}^{K} \prod_{s=1}^{d_k} \mathcal{N}(\mathbf{u}_s^k | \mathbf{m}_s^k, v\mathbf{I}) \tag{2}$$

Bayesian model (Cont.)

- Gaussian likelihood,

$$p(y_{\mathbf{i}}|\mathcal{U}) = \mathcal{N}(y_{\mathbf{i}}|\mathbf{1}^\top(\mathbf{u}^1_{i_1} \circ \ldots \circ \mathbf{u}^K_{i_K}), \tau^{-1}) \qquad (3)$$

- Gamma prior over τ

$$p(\tau|a_0, b_0) = \mathrm{Gam}(\tau|a_0, b_0) = \frac{b_0^{a_0}}{\Gamma(a_0)}\tau^{a_0-1}e^{-b_0\tau}. \qquad (4)$$

The joint probability (Cont.)

$$p(\{y_{\mathbf{i}}\}_{\mathbf{i}\in S}, \mathcal{U}, \tau) = \mathrm{Gam}(\tau|a_0, b_0) \prod_{k=1}^{K} \prod_{s=1}^{d_k} \mathcal{N}(\mathbf{u}_s^k|\mathbf{m}_s^k, v\mathbf{I}) \cdot \prod_{\mathbf{i}\in S} \mathcal{N}(y_{\mathbf{i}}|\mathbf{1}^\top(\mathbf{u}_{i_1}^1 \circ \ldots \circ \mathbf{u}_{i_K}^K), \tau^{-1}), \quad (6)$$

Continuous Tensor

Denote $\mathcal{D}_t = \{y_{\mathbf{i}}\}_{\mathbf{i} \in S_1 \cup \ldots \cup S_{t-1}}$

From the joint probability:

$$p(\mathcal{D}_t \cup \{y_{\mathbf{i}}\}_{\mathbf{i} \in S_t}, \mathcal{U}, \tau) = p(\mathcal{D}_t, \mathcal{U}, \tau) p(\{y_{\mathbf{i}}\}_{\mathbf{i} \in S_t} | \mathcal{U}, \tau). \quad (8)$$

Continuous Tensor

$$p(\mathcal{D}_t \cup \{y_{\mathbf{i}}\}_{\mathbf{i}\in S_t}, \mathcal{U}, \tau) = p(\mathcal{D}_t, \mathcal{U}, \tau) p(\{y_{\mathbf{i}}\}_{\mathbf{i}\in S_t} | \mathcal{U}, \tau). \quad (8)$$

Dividing both sides by the marginal probability $p(\mathcal{D}_t \cup \{y_{\mathbf{i}}\}_{\mathbf{i}\in S_t})$

$$p(\mathcal{U}, \tau | \mathcal{D}_t \cup \{y_{\mathbf{i}}\}_{\mathbf{i}\in S_t}) \propto p(\mathcal{U}, \tau | \mathcal{D}_t) p(\{y_{\mathbf{i}}\}_{\mathbf{i}\in S_t} | \mathcal{U}, \tau). \quad (9)$$

As we can see, the updated posterior is proportional to the current posterior multiplying with the likelihood of the received data batch. Hence, we can use a recursive process for the streaming inference.

Continuous Tensor

- The computation of the posterior update is intractable
- How to overcome this problem

Continuous Tensor

- Factorized variational posterior distribution

$$q(\mathcal{U}, \tau) = q(\tau) \prod_{k=1}^{K} \prod_{s=1}^{d_k} q(\mathbf{u}_s^k), \tag{10}$$

- To approximate $p(\mathcal{U}, \tau | \mathcal{D}_t)$ in $p(\mathcal{U}, \tau | \mathcal{D}_t \cup \{y_\mathbf{i}\}_{\mathbf{i} \in S_t}) \propto p(\mathcal{U}, \tau | \mathcal{D}_t) p(\{y_\mathbf{i}\}_{\mathbf{i} \in S_t} | \mathcal{U}, \tau).$ (9

- Calculate the best approximation $q^*(\mathcal{U}, \tau)$

variational inference

- Minimize the Kullback Leibler (KL) divergence between

$q^*(\mathcal{U}, \tau)$ and $\frac{1}{C} p(\{y_{\mathbf{i}}\}_{\mathbf{i} \in S_t} | \mathcal{U}, \tau) q(\mathcal{U}, \tau$

- Variational model evidence lower bound (maximizing)

$$\mathcal{L} = \int q^*(\mathcal{U}, \tau) \log \frac{p(\{y_{\mathbf{i}}\}_{\mathbf{i} \in S_t} | \mathcal{U}, \tau) q(\mathcal{U}, \tau)}{q^*(\mathcal{U}, \tau)} \mathrm{d}\mathcal{U} \mathrm{d}\tau. \quad (11)$$

- They are equal

Continuous Tensor

- Optimize $q^*(\mathcal{U}, \tau)$ with efficient close-form updates
- Gamma distribution multiplying with (multivariate) Gaussian distributions

$$q(\mathcal{U}, \tau) = q(\tau) \prod_{k=1}^{K} \prod_{s=1}^{d_k} q(\mathbf{u}_s^k) = \text{Gamma}(\tau | a, b) \prod_{k=1}^{K} \prod_{s=1}^{d_k} \mathcal{N}(\mathbf{u}_s^k | \boldsymbol{\mu}_s^k, \boldsymbol{\Sigma}_s^k). \quad (12)$$

Continuous Tensor

- In maximizing the variational bound

$$q^*(\mathcal{U}, \tau) = q^*(\tau) \prod_{k=1}^{K} \prod_{s=1}^{d_k} q^*(\mathbf{u}_s^k),$$

- With the batch S_t, we only need to update the posterior terms

$q^*(\tau)$ and $\{\{q^*(\mathbf{u}_{i_k}^k)\}_{k=1}^K\}_{\mathbf{i} \in S_t}$

Continuous Tensor

we can derive the following close-form updates,

$$q^*(\mathbf{u}_{i_k}^k) = \mathcal{N}(\mathbf{u}_{i_k}^k | {\boldsymbol{\mu}_{i_k}^k}^*, {\boldsymbol{\Sigma}_{i_k}^k}^*), \tag{13}$$

$$q^*(\tau) = \mathrm{Gam}(\tau | a^*, b^*). \tag{14}$$

Continuous Tensor

The sufficient statistics are given by

$$\boldsymbol{\Sigma}_{i_k}^{k\ *} = \left(\boldsymbol{\Sigma}_{i_k}^{k\ -1} + \langle\tau\rangle \sum\nolimits_{\mathbf{j}\in S_t, j_k=i_k} \langle \mathbf{t}_{\mathbf{j},\neg k}\mathbf{t}_{\mathbf{j},\neg k}^{\top}\rangle\right)^{-1}, \quad (15)$$

$$\boldsymbol{\mu}_{i_k}^{k\ *} = \boldsymbol{\Sigma}_{i_k}^{k\ *}(\boldsymbol{\Sigma}_{i_k}^{k\ -1}\boldsymbol{\mu}_{i_k}^{k} + \langle\tau\rangle \sum\nolimits_{\mathbf{j}\in S_t, j_k=i_k} y_{\mathbf{j}}\langle \mathbf{t}_{\mathbf{j},\neg k}\rangle), \quad (16)$$

$$a^* = a + \frac{1}{2}|S_t|, \quad (17)$$

$$b^* = b + \frac{1}{2}\sum\nolimits_{\mathbf{i}\in S_t}\left[y_{\mathbf{i}}^2 - 2y_{\mathbf{i}}\langle \mathbf{1}^{\top}\mathbf{t}_{\mathbf{i}}\rangle + \langle(\mathbf{1}^{\top}\mathbf{t}_{\mathbf{i}})^2\rangle\right], \quad (18)$$

where $\langle\cdot\rangle$ is the expectation w.r.t the other posteriors, $|\cdot|$ is the size of the batch, $\langle\tau\rangle = \frac{a^*}{b^*}$,

$$\mathbf{t}_{\mathbf{i}} = \mathbf{u}_{i_1}^1 \circ \ldots \mathbf{u}_{i_K}^K,$$
$$\mathbf{t}_{\mathbf{j},\neg k} = \mathbf{u}_{j_1}^1 \circ \ldots \mathbf{u}_{j_{k-1}}^{k-1} \circ \mathbf{u}_{j_{k+1}}^{k+1} \ldots \circ \mathbf{u}_{j_K}^K,$$

and

$$\langle \mathbf{t}_{\mathbf{j},\neg k}\rangle = \boldsymbol{\mu}_{j_1}^{1\ *} \circ \ldots \boldsymbol{\mu}_{j_{k-1}}^{k-1\ *} \circ \boldsymbol{\mu}_{j_{k+1}}^{k+1\ *} \ldots \circ \boldsymbol{\mu}_{j_K}^{K\ *},$$
$$\langle \mathbf{t}_{\mathbf{j},\neg k}\mathbf{t}_{\mathbf{j},\neg k}^{\top}\rangle = \langle \mathbf{u}_{j_1}^1 \mathbf{u}_{j_1}^{1\ \top}\rangle \circ \ldots \circ \langle \mathbf{u}_{j_{k-1}}^{k-1} \mathbf{u}_{j_{k-1}}^{k-1\ \top}\rangle,$$
$$\circ \langle \mathbf{u}_{j_{k+1}}^{k+1} \mathbf{u}_{j_{k+1}}^{k+1\ \top}\rangle \circ \ldots \circ \langle \mathbf{u}_{j_K}^{K} \mathbf{u}_{j_K}^{K\ \top}\rangle$$
$$\langle \mathbf{1}^{\top}\mathbf{t}_{\mathbf{i}}\rangle = \mathbf{1}^{\top}(\boldsymbol{\mu}_{i_1}^1 \circ \ldots \circ \boldsymbol{\mu}_{i_K}^K),$$
$$\langle(\mathbf{1}^{\top}\mathbf{t}_{\mathbf{i}})^2\rangle = \mathrm{tr}\big(\mathbf{1}\cdot\mathbf{1}^{\top}(\langle \mathbf{u}_{i_1}^1 \mathbf{u}_{i_1}^{1\ \top}\rangle \circ \ldots \circ \langle \mathbf{u}_{i_K}^K \mathbf{u}_{j_K}^{K\ \top}\rangle)\big)$$

Bayesian model (binary)

- Probit likelihood,

$$p(y_{\mathbf{i}}|\mathcal{U}) = \phi\big((2y_{\mathbf{i}} - 1)\mathbf{1}^\top(\mathbf{u}^1_{i_1} \circ \ldots \circ \mathbf{u}^K_{i_K})\big) \qquad (5)$$

The joint probability (binary)

$$p(\{y_{\mathbf{i}}\}_{\mathbf{i}\in S}, \mathcal{U}) = \prod_{k=1}^{K}\prod_{s=1}^{d_k} \mathcal{N}(\mathbf{u}_s^k | \mathbf{m}_s^k, v\mathbf{I}) \cdot \prod_{\mathbf{i}\in S} \phi\big((2y_{\mathbf{i}} - 1)\mathbf{1}^\top (\mathbf{u}_{i_1}^1 \circ \ldots \circ \mathbf{u}_{i_K}^K)\big). \quad (7)$$

Binary Tensor

- The Probit likelihood cannot derive close-form updates for $q(\mathcal{U})$,

- Overcome this problem

Binary Tensor

we augment the Probit likelihood with a latent continuous random variable $z_{\mathbf{i}}$ for each observed entry $\mathbf{i}$, and we have

$$p(y_{\mathbf{i}}, z_{\mathbf{i}}|\mathcal{U}, \tau) = \mathcal{N}\big(z_{\mathbf{i}}|\mathbf{1}^\top(\mathbf{u}^1_{i_1} \circ \ldots \circ \mathbf{u}^K_{i_K}), 1\big)p(y_{\mathbf{i}}|z_{\mathbf{i}}) \quad (19)$$

where $p(y_{\mathbf{i}}|z_{\mathbf{i}}) = \mathbb{1}(y_{\mathbf{i}} = 1)\mathbb{1}(z_{\mathbf{i}} \geq 0) + \mathbb{1}(y_{\mathbf{i}} = 0)\mathbb{1}(z_{\mathbf{i}} < 0)$ and $\mathbb{1}(\cdot)$ is the indicator function. It is easy to show that by marginalizing out $z_{\mathbf{i}}$ in (19), we can recover the original Probit likelihood (5).

Binary Tensor

- Factorized posterior distribution $q(\mathcal{U}) = \prod_{k=1}^{K} \prod_{s=1}^{d_k} q(\mathbf{u}_s^k)$
- The latent variables $\mathbf{z} = \{z_{\mathbf{i}}\}_{\mathbf{i} \in S_t}$
- Derive a variational lower bound

$$\mathcal{L} = \int q^*(\mathcal{U}) q(\mathbf{z}) \log \frac{p(\{z_{\mathbf{i}}, y_{\mathbf{i}}\}_{\mathbf{i} \in S_t} | \mathcal{U}, \tau) q(\mathcal{U})}{q^*(\mathcal{U}) q(\mathbf{z})} \mathrm{d}\mathcal{U} \mathrm{d}\mathbf{z}. \quad (20)$$

- Factorized form for the posterior of **z** , $q(\mathbf{z}) = \prod_{\mathbf{i} \in S_t} q(z_{\mathbf{i}})$

Binary Tensor

- To maximizne L , we then update the $q(\mathbf{u}_\mathbf{i}^k)$ and $q(z_\mathbf{i})$ in the batch.

- To update $q(z_\mathbf{i})$, is a truncated Gaussian distribution:

$$q(z_\mathbf{i}) \propto \mathcal{N}\big(z_\mathbf{i}|\langle \mathbf{1}^\top \mathbf{t}_\mathbf{i}\rangle, 1\big)\mathbb{1}\big((2y_\mathbf{i}-1)z_\mathbf{i} \geq 0\big). \qquad (21)$$

- To update. $q(\mathbf{u}_\mathbf{i}^k)$, is a Gaussian distribution. Same as the continuous case .

Binary Tensor

- But we need to remove ⟨τ⟩ and replace y_j by $\langle z_j \rangle$

We can calculate each $\langle z_i \rangle$ through

$$\langle z_i \rangle = \langle \mathbf{1}^\top \mathbf{t}_i \rangle + \frac{(2y_i - 1)\mathcal{N}(\langle \mathbf{1}^\top \mathbf{t}_i \rangle | 0, 1)}{\phi\big((2y_i - 1)\langle \mathbf{1}^\top \mathbf{t}_i \rangle\big)}. \qquad (22)$$

Prediction

- For continuous tensor data :

$$p(y_{\mathbf{i}}|D_t) = \int \mathcal{N}(y_{\mathbf{i}}|\mathbf{1}^\top \mathbf{t}_{\mathbf{i}}, \tau^{-1}) q(\tau) \prod_k q(\mathbf{u}_{i_k}^k) \mathrm{d}\tau \prod_k \mathrm{d}\mathbf{u}_{i_k}^k$$

- Mean and Variance(cont.)

$$\mathrm{E}(y_{\mathbf{i}}) = \langle \mathbf{1}^\top \mathbf{t}_{\mathbf{i}} \rangle, \quad \mathrm{Var}(y_{\mathbf{i}}) = \langle (\mathbf{1}^\top \mathbf{t}_{\mathbf{i}})^2 \rangle + \frac{b}{a-1} - \mathrm{E}^2(y_{\mathbf{i}}).$$

- For binary tensor data:

$$p(y_{\mathbf{i}}|D_t) = \int \phi\big((2y_{\mathbf{i}} - 1)\mathbf{1}^\top \mathbf{t}_{\mathbf{i}}\big) \prod_k q(\mathbf{u}_{i_k}^k) \prod_k \mathrm{d}\mathbf{u}_{i_k}^k. \qquad g_{\mathbf{i}} = \mathbf{1}^\top \mathbf{t}_{\mathbf{i}}.$$

- Mean and Variance(binary.)

$$\mathrm{E}(y_{\mathbf{i}}) = p(y_{\mathbf{i}} = 1|D_t) \approx \phi\Big(\frac{\mathrm{E}(g_{\mathbf{i}})}{\sqrt{1 + \mathrm{Var}(g_{\mathbf{i}})}}\Big), \qquad (23)$$

$$\mathrm{Var}(y_{\mathbf{i}}) = \mathrm{E}(y_{\mathbf{i}}) - \mathrm{E}^2(y_{\mathbf{i}}). \qquad (24)$$

EXPERIMENT

- **Q1.** How does POST compare with MAST, in the case that new entries always occur in tensor increments and increase the full tensor size?
- **Q2.** How does POST perform, when tensor entries stream in arbitrary orders, as compared with the traditional batch decomposition approaches?
- **Q3.** What does the uncertainty information produced by POST reveal and how does it potentially benefit practical applications, such as online advertising?

Data

- *Twitter Topic* (user,expert,topic) , 500×500×20
- *MovieLens* (user, movie, week) , 400 × 400 × 31
- Compared with MAST , CP-ALS and TNCP
- randomly chose {50%, 80%, 90%} as missing data
- the dimension of embedding vectors, from {3,5,8,10}

Q1

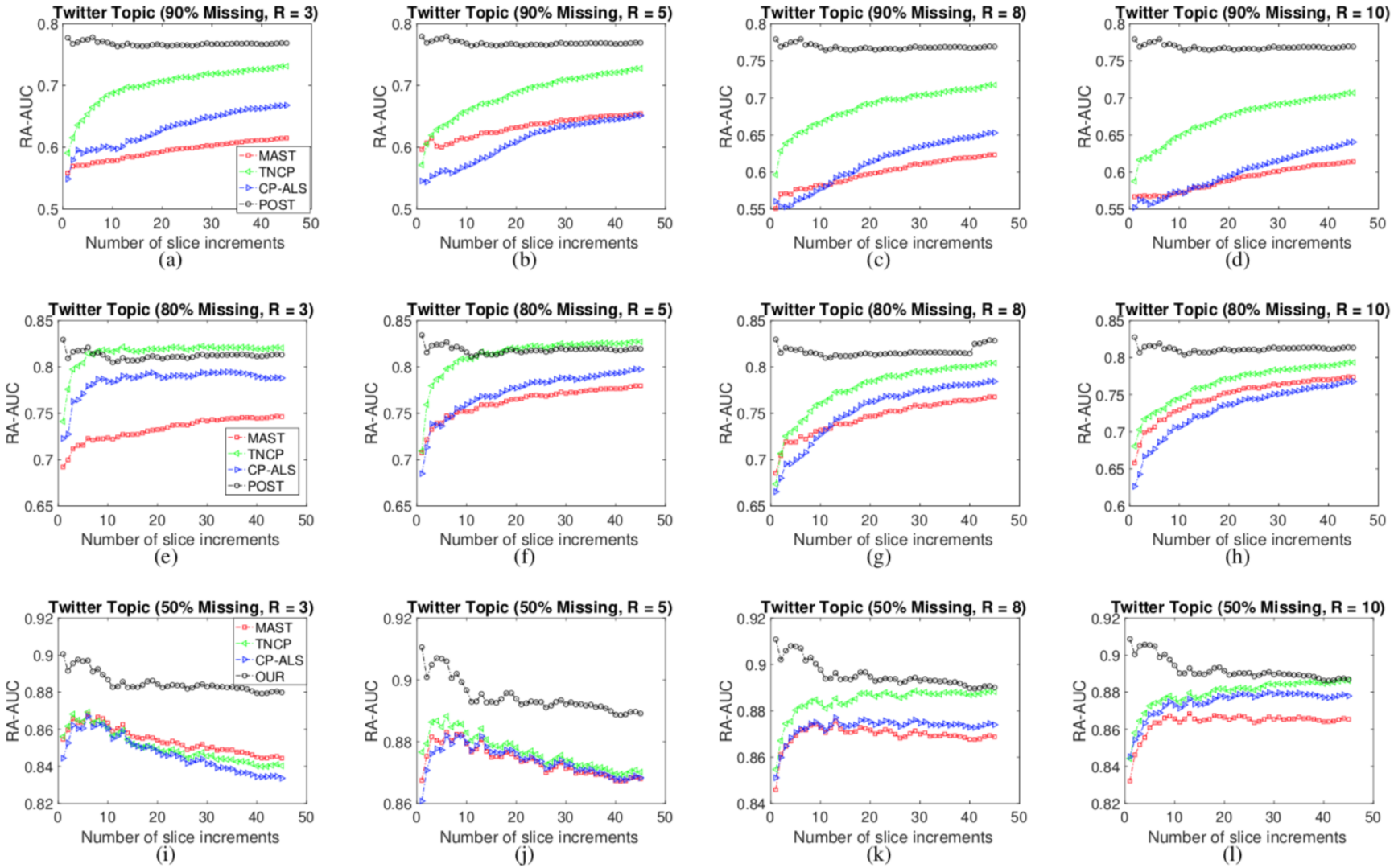


Fig. 1. The running average AUC on *Twitter Topic* dataset. Each time, 10 $user \times expert$ slices were appended. The results were averaged over 5 runs.

Q1

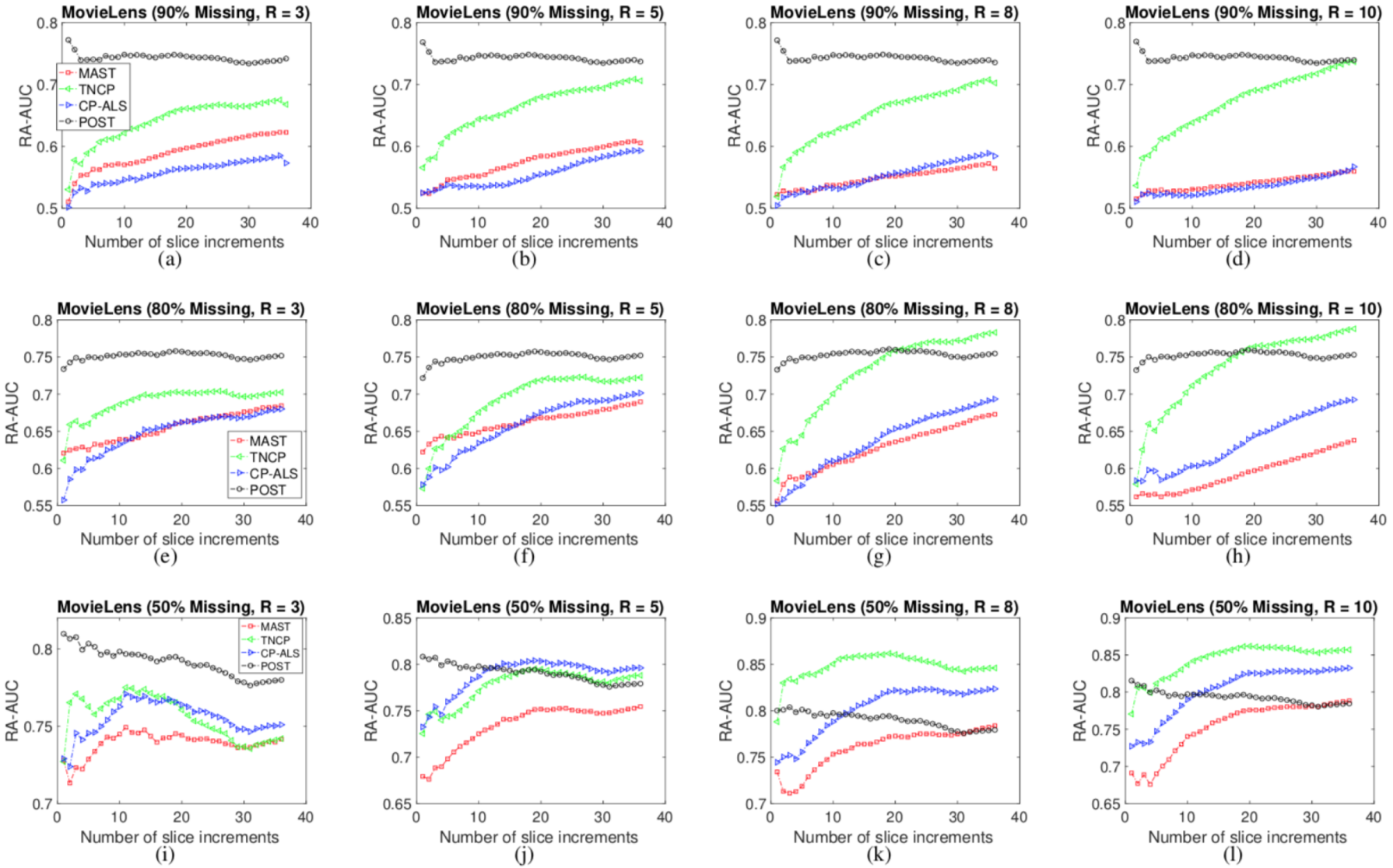
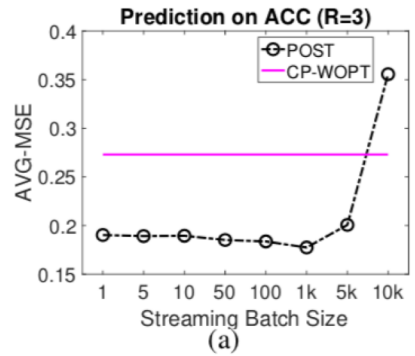


Fig. 2. The running average AUC on *MovieLen* data. Each time, 10 $user \times movie$ slices were appended. The results were averaged over 5 runs.

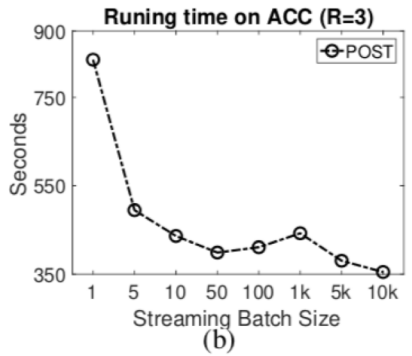
Data

- *ACC ,* (user, action, resource) , 3K × 150 × 30K, **cont.**
- *DBLP ,* (author, conf erence, keyword) , 3K × 150 × 30K , **binay**
- Compared CP-WOPT
- Rank R from {3, 5, 8}.
- Batch {1,5,10,50,100,1K,5K,10K}

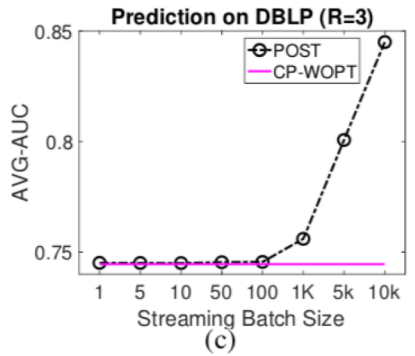
Q2



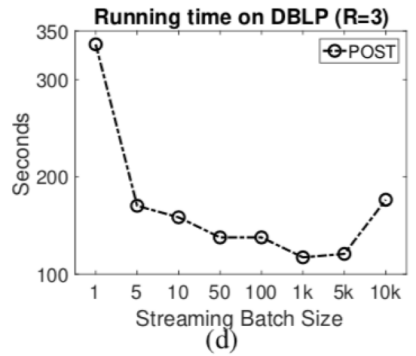
(a)



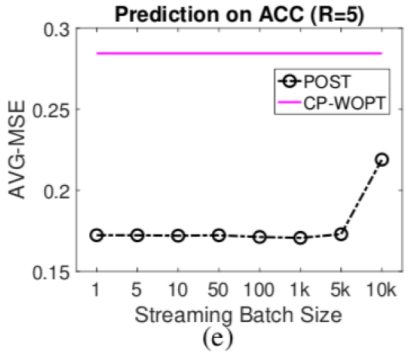
(b)



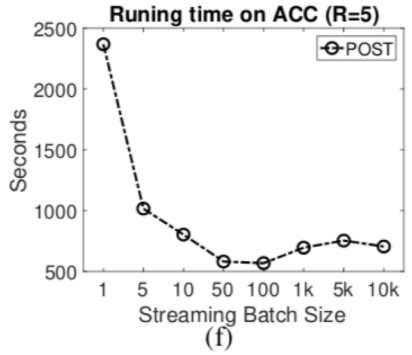
(c)



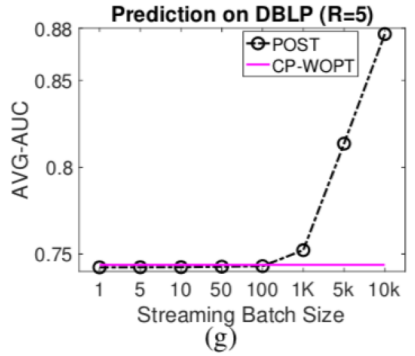
(d)



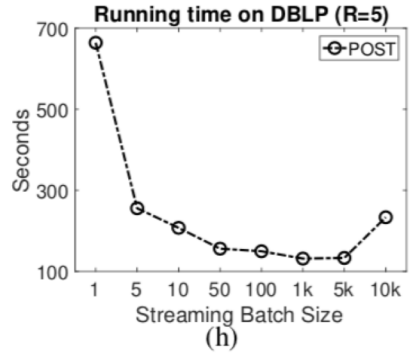
(e)



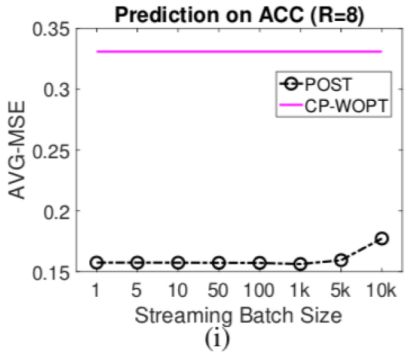
(f)



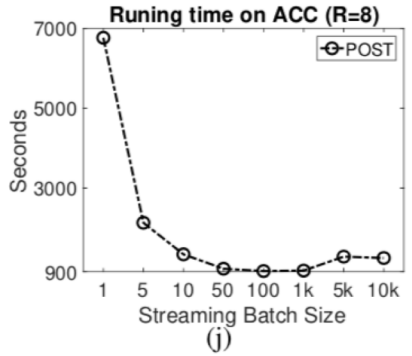
(g)



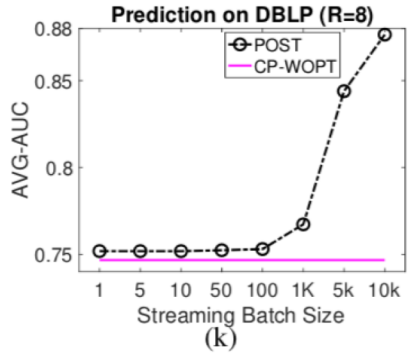
(h)



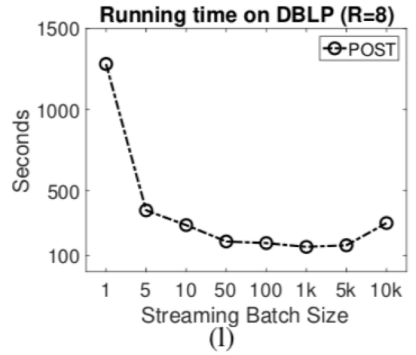
(i)



(j)



(k)



(l)

Data

- click- through-rate (CTR) {banner_pos,site_id,app_id,device_model} from Avazu Inc

- 7×2854×4114×6061

- AUC of the POST is 0.74
- AUC of the CP_WOPT is 0.49

Q3

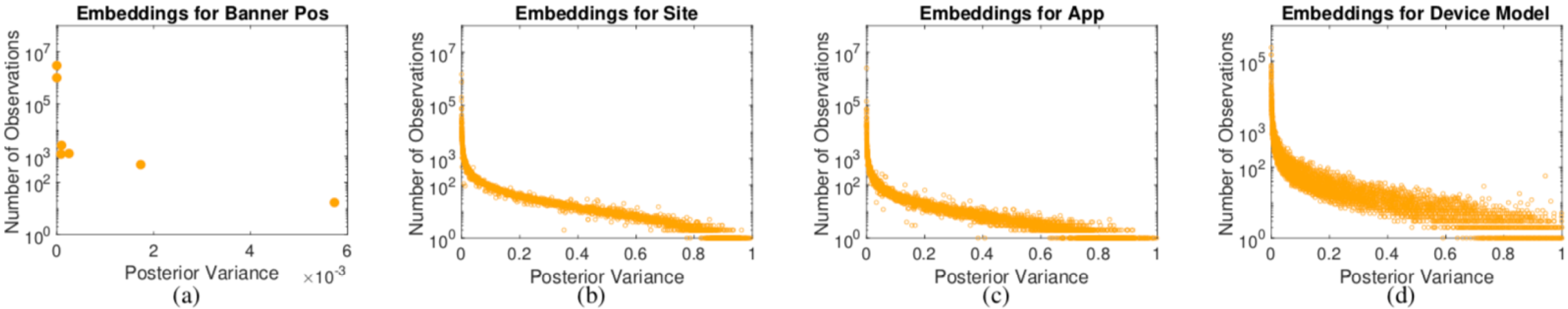


Fig. 4. The posterior variances of the embeddings ($R = 1$) *vs.* the number of observations for the corresponding objects in each mode.

Q3

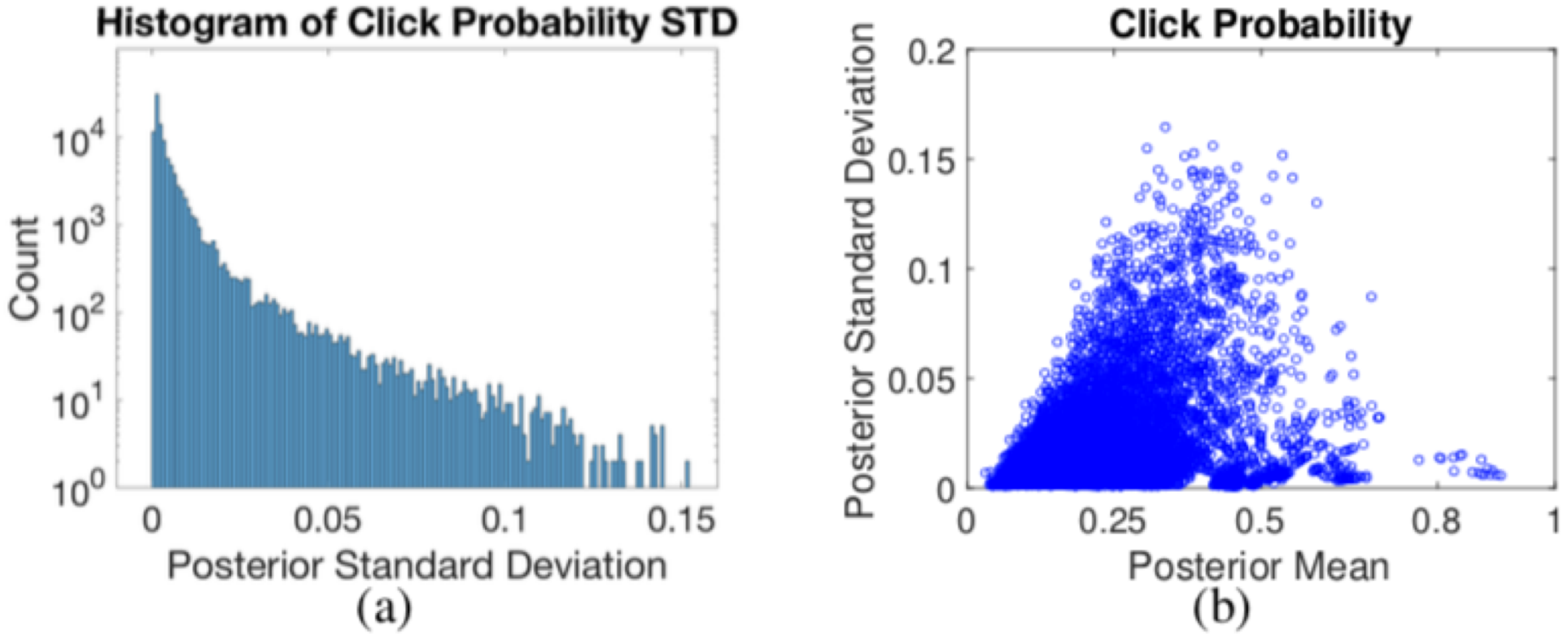


Fig. 5. The uncertainty of the click probabilities for $100K$ test entries.