TCS: Efficient Topic Discovery over Crowd-oriented Service Data

Yongxin Tong, Caleb Chen Cao, Lei Chen

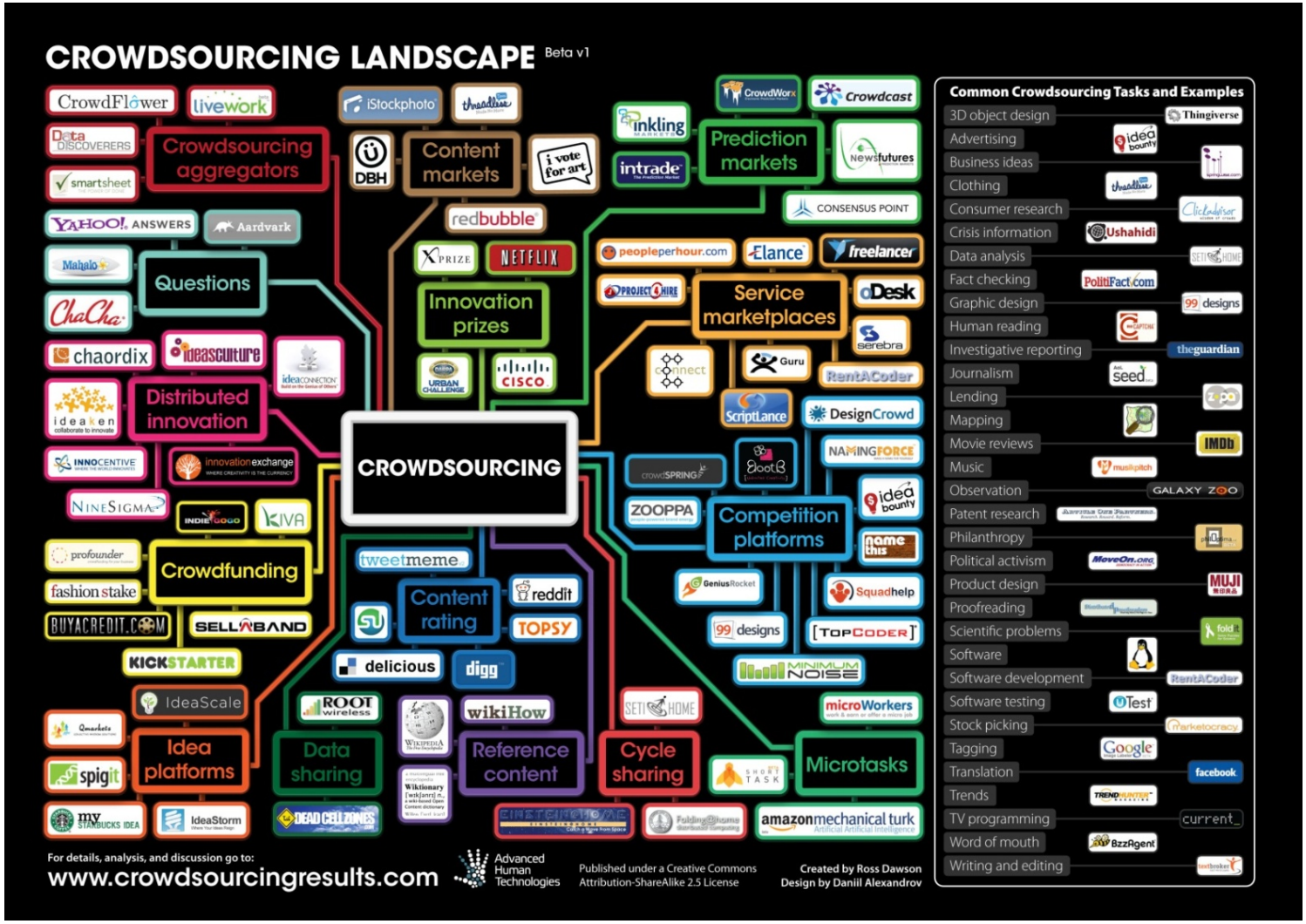
Department of Computer Science and Engineering
The Hong Kong University of Science and Technology

SIGKDD 2014

Outline

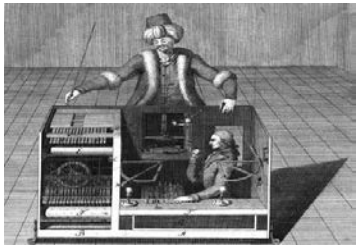
- Motivations
- Problem Definitions
- Topic Crowd Service Model
- Pairwise Sketch
- Parameter Estimation
- Experimental Study
- Conclusion

Crowdsourcing in Social Media

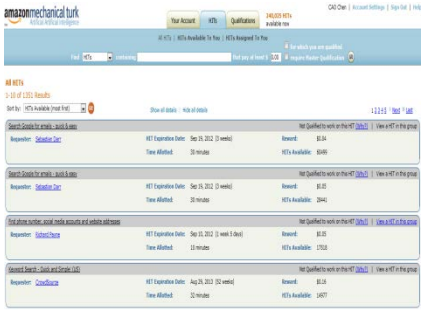


Crowdsourcing Process

Requesters



AMT



MTurk workers
(Photo By Andrian Chen)

- **The web connects the *tasks* from requestors and *responses* from workers.**

Crowd-Oriented Services

- **The information services provided by crowdsourcing usually include *massive task-response pairs*.**

Crowd-Oriented Service Data

- A snippet of crowd-oriented service from Stack Overflow

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33
T2	How to use simpleadapter with activity Android...	2014-02-02 10:31:25
T3	How to find mobile phones on hotspot networks in iPhone?	2014-02-03 21:40:01

Crowd-Oriented Service Data

- A snippet of crowd-oriented service from Stack Overflow
  - Task ID

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33
T2	How to use simpleadapter with activity Android...	2014-02-02 10:31:25
T3	How to find mobile phones on hotspot networks in iPhone?	2014-02-03 21:40:01

Crowd-Oriented Service Data

- A snippet of crowd-oriented service from Stack Overflow
  - Task ID, Task Details,

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33
T2	How to use simpleadapter with activity Android...	2014-02-02 10:31:25
T3	How to find mobile phones on hotspot networks in iPhone?	2014-02-03 21:40:01

Crowd-Oriented Service Data

- A snippet of crowd-oriented service from Stack Overflow
  - Tasks: Task ID, Task Details, Timestamp, etc.

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33
T2	How to use simpleadapter with activity Android...	2014-02-02 10:31:25
T3	How to find mobile phones on hotspot networks in iPhone?	2014-02-03 21:40:01

Crowd-Oriented Service Data

- A snippet of crowd-oriented service from Stack Overflow
  - Tasks: Task ID, Task Details, Timestamp, etc.

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33
T2	How to use simpleadapter with activity Android...	2014-02-02 10:31:25
T3	How to find mobile phones on hotspot networks in iPhone?	2014-02-03 21:40:01

  - Responses: Responses ID

Response ID	Task ID	Responses	Timestamp
R1	T1	Android SQLite database with multiple...	2014-01-31 18:23:01
R2	T1	Save the image to your sdcard. ...	2014-02-01 15:01:53
R3	T1	Storing images in your database will…	2014-02-01 16:38:17

Crowd-Oriented Service Data

- A snippet of crowd-oriented service from Stack Overflow
  - Tasks: Task ID, Task Details, Timestamp, etc.

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33
T2	How to use simpleadapter with activity Android...	2014-02-02 10:31:25
T3	How to find mobile phones on hotspot networks in iPhone?	2014-02-03 21:40:01

  - Responses: Responses ID, Task ID

Response ID	Task ID	Responses	Timestamp
R1	T1	Android SQLite database with multiple...	2014-01-31 18:23:01
R2	T1	Save the image to your sdcard. ...	2014-02-01 15:01:53
R3	T1	Storing images in your database will…	2014-02-01 16:38:17

Crowd-Oriented Service Data

- A snippet of crowd-oriented service from Stack Overflow
  - Tasks: Task ID, Task Details, Timestamp, etc.

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33
T2	How to use simpleadapter with activity Android...	2014-02-02 10:31:25
T3	How to find mobile phones on hotspot networks in iPhone?	2014-02-03 21:40:01

  - Responses: Responses ID, Task ID, Response Details,

Response ID	Task ID	Responses	Timestamp
R1	T1	Android SQLite database with multiple...	2014-01-31 18:23:01
R2	T1	Save the image to your sdcard. ...	2014-02-01 15:01:53
R3	T1	Storing images in your database will…	2014-02-01 16:38:17

Crowd-Oriented Service Data

- A snippet of crowd-oriented service from Stack Overflow
  - Tasks: Task ID, Task Details, Timestamp, etc.

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33
T2	How to use simpleadapter with activity Android...	2014-02-02 10:31:25
T3	How to find mobile phones on hotspot networks in iPhone?	2014-02-03 21:40:01

  - Responses: Responses ID, Task ID, Response Details, Timestamp, etc.

Response ID	Task ID	Responses	Timestamp
R1	T1	Android SQLite database with multiple...	2014-01-31 18:23:01
R2	T1	Save the image to your sdcard. ...	2014-02-01 15:01:53
R3	T1	Storing images in your database will…	2014-02-01 16:38:17

Characteristic of Crowd-Oriented Service Data-I

- Task-Response Pairs
  - Task-Response Correlation

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33
T2	How to use simpleadapter with activity Android...	2014-02-02 10:31:25
T3	How to find mobile phones on hotspot networks in iPhone?	2014-02-03 21:40:01

Response ID	Task ID	Responses	Timestamp
R1	T1	Android SQLite database with multiple...	2014-01-31 18:23:01
R2	T1	Save the image to your sdcard. ...	2014-02-01 15:01:53
R3	T1	Storing images in your database will…	2014-02-01 16:38:17

Characteristic of Crowd-Oriented Service Data-II

- Big volume
  - Each task may have large amount of responses

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33
…	…	…

Response ID	Task ID	Responses	Timestamp
R1	T1	Android SQLite database with multiple...	2014-01-31 18:23:01
…	…	…	…
R100	T1	Storing images in your database will…	2014-02-04 11:36:02
…	…	…	…

Characteristic of Crowd-Oriented Service Data-III

- Dynamic Evolution with Time

Timeline

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33

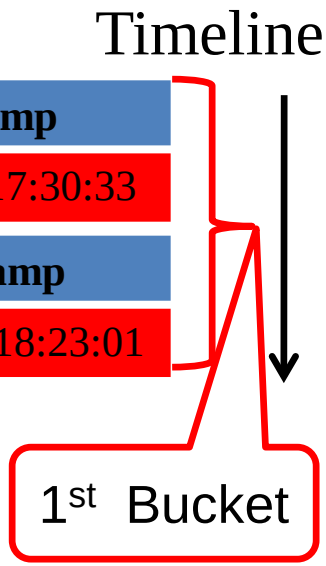
1st Bucket

Characteristic of Crowd-Oriented Service Data-III

- Dynamic Evolution with Time
  - Accumulates as Multiple Consecutive Buckets

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33

Response ID	Task ID	Responses	Timestamp
R1	T1	Android SQLite database with multiple...	2014-01-31 18:23:01

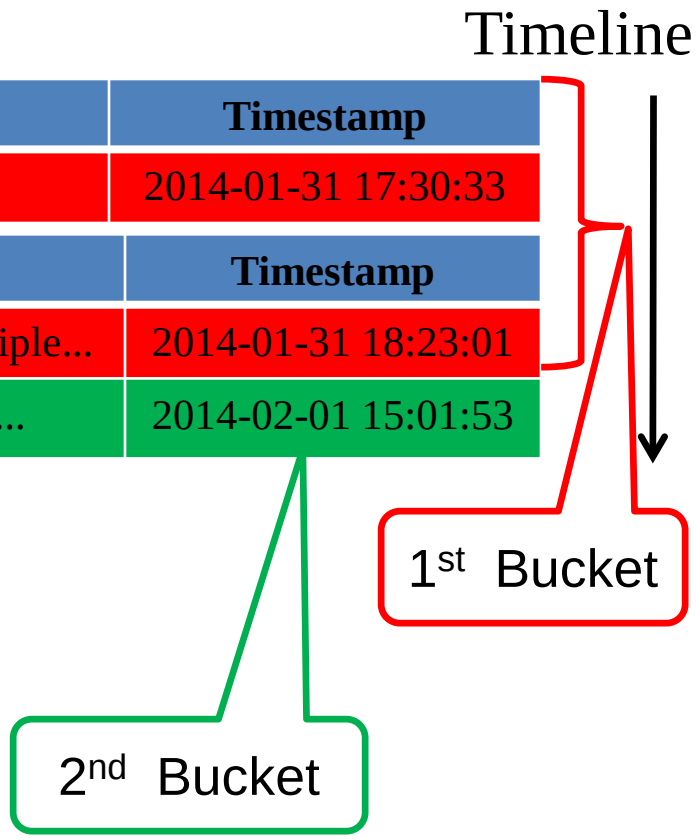


Characteristic of Crowd-Oriented Service Data-III

- Dynamic Evolution with Time

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33

Response ID	Task ID	Responses	Timestamp
R1	T1	Android SQLite database with multiple...	2014-01-31 18:23:01
R2	T1	Save the image to your sdcard. ...	2014-02-01 15:01:53



Characteristic of Crowd-Oriented Service Data-III

- Dynamic Evolution with Time

Timeline

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33

Response ID	Task ID	Responses	Timestamp
R1	T1	Android SQLite database with multiple...	2014-01-31 17:30:33
R2	T1	Save the image to your sdcard. ...	2014-02-01 10:31:25
R3	T1	Storing images in your database will…	2014-02-01 21:40:01

2nd Bucket

1st Bucket

Characteristic of Crowd-Oriented Service Data-III

- Dynamic Evolution with Time

Timeline

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33

Response ID	Task ID	Responses	Timestamp
R1	T1	Android SQLite database with multiple...	2014-01-31 17:30:33
R2	T1	Save the image to your sdcard. ...	2014-02-01 10:31:25
R3	T1	Storing images in your database will…	2014-02-01 21:40:01

Task ID	Tasks	Timestamp
T2	How to use simpleadapter with activity Android...	2014-02-02 10:31:25

2nd Bucket

3rd Bucket

1st Bucket

Characteristic of Crowd-Oriented Service Data-III

- Dynamic Evolution with Time
  - Accumulates as Multiple Consecutive Buckets

Timeline

Task ID	Tasks	Timestamp
T1	Android application database to save images ...	2014-01-31 17:30:33

Response ID	Task ID	Responses	Timestamp
R1	T1	Android SQLite database with multiple...	2014-01-31 17:30:33
R2	T1	Save the image to your sdcard. ...	2014-02-01 10:31:25
R3	T1	Storing images in your database will…	2014-02-01 21:40:01

Task ID	Tasks	Timestamp
T2	How to use simpleadapter with activity Android...	2014-02-02 10:31:25
…	…….	….

Applications of Topics in Crowd-Oriented Service Data

- **Accuracy of Crowdsourcing Systems**
  - Quality Control
- **Users Personalization**
  - Task Requestors
  - Crowd Workers
- **Task Recommendation**
  - Task Assignment in Crowdsourcing Platforms

Challenges

- **How to model crowd-oriented service data containing**
  - Task-Response Correlation

Responses ID	Task ID	Tasks	Timestamp
R1	T1	Android SQLite database with multiple...	2014-01-31 17:30:33
R2	T1	Save the image to your sdcard. ...	2014-02-01 10:31:25
R3	T1	Storing images in your database will…	2014-02-01 21:40:01

- **High training efficiency is important**

- **Topics over crowd-oriented service data are evolving**

Our Contributions

- We design the New probabilistic topic model to crowd-oriented service data.
  - Topic Crowd Service Model
- We propose an efficient solution to discover latent topics from crowd-oriented service data.
  - Pairwise Sketch (pSketch)
  - Bucket Parameter Estimation (BPE) Algorithm
- We verify the effectiveness and efficiency of the proposed methods in extensive experimental results.

Outline

- Motivations
- Problem Definitions
- Topic Crowd Service Model
- Pairwise Sketch
- Parameter Estimation
- Experimental Study
- Conclusion

Problem Definitions

- **Task-Response Pairs**
  - Given a crowd-oriented task T_i, a set of corresponding responses $\{R_{i,1},\ldots, R_{i,m}\}$, the arbitrary pair $(T_i, R_{i,j})$ where $j \in [1, m]$ is called a task-response pair.

- **Crowd-oriented Service Data**
  - $CS=\{(T_1, R_{1,1}),\ldots,(T_n, R_{n,1}),\ldots,(T_n, R_{n,m})\}$ be a set of task-response pairs, where each task and response is a document. Each document *d* is represented by a subset of the collection of words. Given an arbitrary task-response pair, a word-pair includes two words, one word is from the document of the task, the other word is from the document of the response.

Problem Definitions

Task ID	Tasks	Timestamp
T_1	Android application database to save images ...	2014-01-31 17:30:33
T_2	How to use simpleadapter with activity Android...	2014-02-02 10:31:25
T_3	How to find mobile phones on hotspot networks in iPhone?	2014-02-03 21:40:01

Response ID	Task ID	Responses	Timestamp
$R_{1,1}$	T_1	Android SQLite database with multiple...	2014-01-31 18:23:01
$R_{1,2}$	T_1	Save the image to your sdcard. ...	2014-02-01 15:01:53
$R_{1,3}$	T_1	Storing images in your database will…	2014-02-01 16:38:17
$R_{3,1}$	T_3	iOS 7 system of Apple devices provide...	2014-02-03 22:14:27

- $(T_1, R_{1,1})$ is a task-response pair.
- In this example, CS={$(T_1, R_{1,1})$, $(T_1, R_{1,2})$, $(T_1, R_{1,3})$, $(T_3, R_{3,1})$}.
- (iPhone, iOS7) is a word-pair in $(T_3, R_{3,1})$.

Problem Definitions

- **Topic**
  - A semantically coherent topic ϕ is a multinomial distribution of words $\{p(w|\phi)\}_{w \in W}$ with the constraint $\sum_{w \in W} p(w|\phi) = 1$.

- **Topic Discovery in Crowd-oriented Service Data**
  - Given the input of a crowd-oriented service data *CS*, we are required to infer the latent topics ϕ over in *CS*.

Outline

- Motivations
- Problem Definitions
- Topic Crowd Service Model
- Pairwise Sketch
- Parameter Estimation
- Experimental Study
- Conclusion

Generative Process of TCS Model

- Each task or response is viewed as a document, respectively.
- TCS shares ingredients with Latent Dirichlet Allocation (LDA):
  - Each topic has a distribution over words;
  - Each document has a distribution over topics;

Algorithm 1: Generative process of TCS

```
for each topic k ∈ {1, ··· , K} do
    draw a word distribution φ_k ~ Dirichlet(β);
for each document d ∈ {1, ··· , D} do
    draw topic distribution θ_d ~ Dirichlet(α);
    if d is a task then
        sample a response d' with regard to the number
        of sketch pairs between d and d';
    if d is a response then
        select the corresponding task d';
    generate new document topic distribution θ' by
    combining θ_d and θ_d';
    for each sentence s ∈ d do
        choose a topic z ~ Multinomial(θ');
        generate words w ~ Multinomial(φ_z);
```

  - If a document *d* is a task, sample a response from the set of task-response pairs;
  - Otherwise, *d* is a response and select its corresponding task;
  - Combine the task and response as a new document and generate the new distribution over topics;
  - Each sentence is the basic unit for topic assignment.

Challenges of TCS Model

- It is infeasible to count and store frequencies of all word pairs due to the excessively high cost.
  - Our Solution: Only storing significant (frequent) word-pairs and removing extremely infrequent word-pairs.

- How to training the TCS model efficiently when the correlation of task-response pair is considered?
  - Our Solution: Speeding up the training and belief updating process according to significant word-pairs.

Outline

- Motivations
- Problem Definitions
- Topic Crowd Service Model
- Pairwise Sketch
- Parameter Estimation
- Experimental Study
- Conclusion

Key ideas of Pairwise Sketch

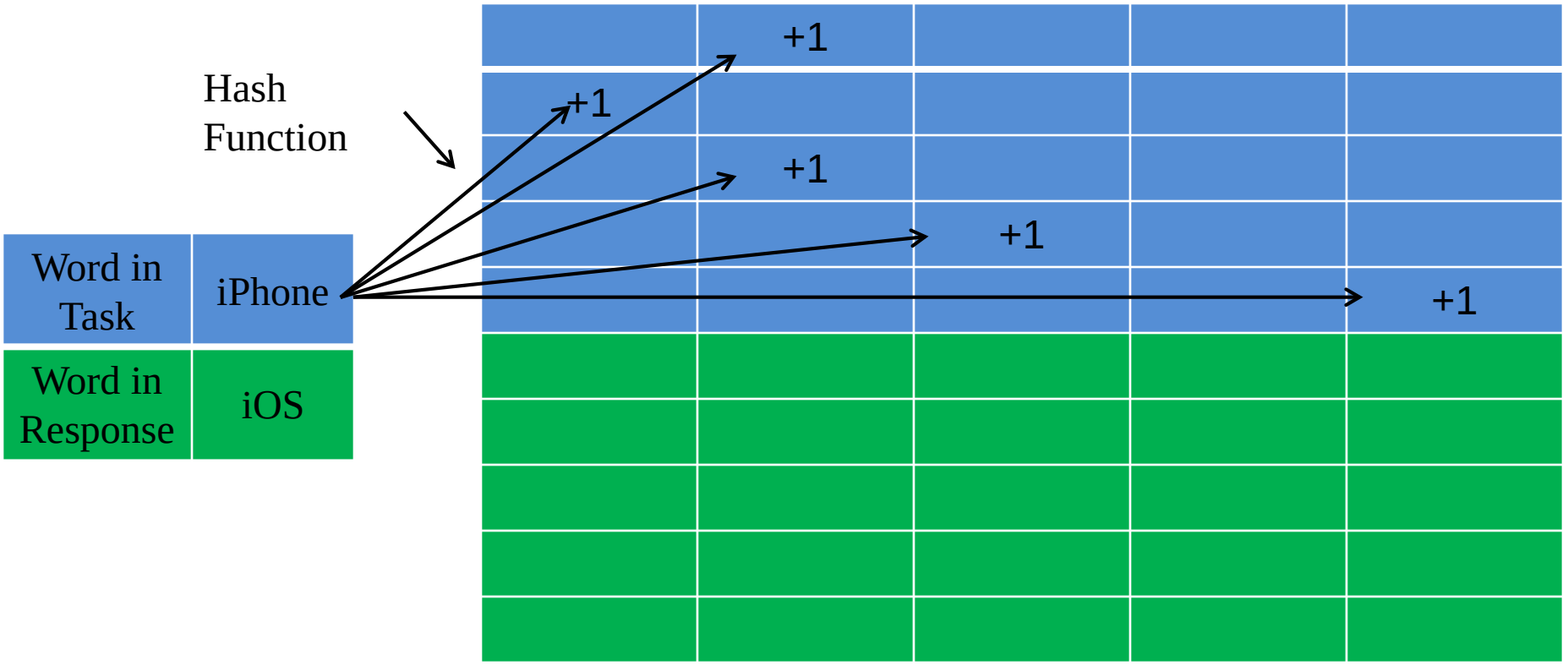
- Main ideas
  - A sketch-based method
  - Approximate the frequency of word-pairs in tasks and responses with bounded error within a probability.
- Only frequent word-pairs are significant for topic modeling
  - Extremely infrequent word-pairs in tasks and responses are removed.
- Effective Space Complexity
  - $O(\frac{1+\delta}{\varepsilon\delta}|T|)$
  - Refer to our paper for the proof if you are interested.

Pairwise Sketch

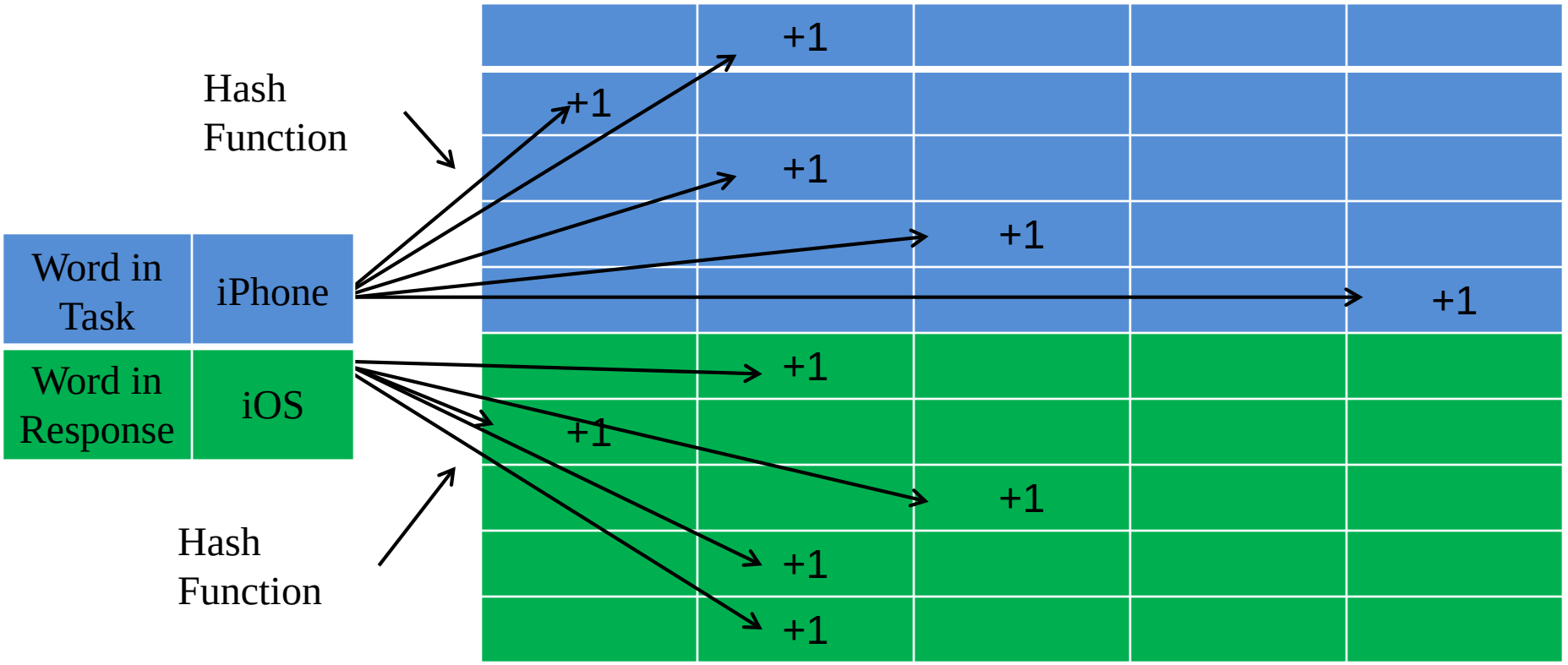
Sketch for counting frequency of words in tasks

Sketch for counting frequency of words in responses

Pairwise Sketch



Pairwise Sketch



Outline

- Motivations
- Problem Definitions
- Topic Crowd Service Model
- Pairwise Sketch
- Parameter Estimation
- Experimental Study
- Conclusion

Belief Residual of Sentences

- The belief that a sentence ***s*** of a specific document *d* is generated by topic ***k*** is denoted by $\mu^k_{d,s}$
  - $\mu^k_{d,s}$ is calculated as follows,

The estimation of the topic distribution inthe the document level.

The estimation of the word distribution in the topic level.

$$\mu^k_{d,s} \propto \frac{(1-\zeta)\mu^k_{d,-s} + \zeta\mu^k_{l_d} + \alpha_k}{\sum_{k'}\left((1-\zeta)\mu^{k'}_{d,-s} + \zeta\mu^{k'}_{l_d} + \alpha_{k'}\right)} \cdot \frac{\Gamma\left(\sum_{r'}(n_{\cdot,-s,w'}\mu^k_{\cdot,-s,w'} + \beta_{w'})\right)}{\Gamma\left(\sum_{w'}(n_{\cdot,-s,w'}\mu^k_{\cdot,-s,w'} + \beta_{w'} + n_{d,s,w'})\right)} \prod_{w\in s}\left(\frac{\Gamma(n_{\cdot,-s,w}\mu^k_{\cdot,-s,w} + \beta_w + n_{d,s,w})}{\Gamma(n_{\cdot,-s,w}\mu^k_{\cdot,-s,w} + \beta_w)}\right)$$

- The belief residual r^k_{ds} between two successive iterations t and (t-1) is calculated as follows,

$$r^k_{d,s} = \left|\mu^k_{d,s}(t) - \mu^k_{d,s}(t-1)\right|$$

Belief Residual of Documents

- The residual of a specific document *d* at topic *k* as follows:

$$r_d^k = \sum_s r_{d,s}^k$$

- The residual of the document *d* is calculated as follows:

$$r_d = \sum_k r_d^k .$$

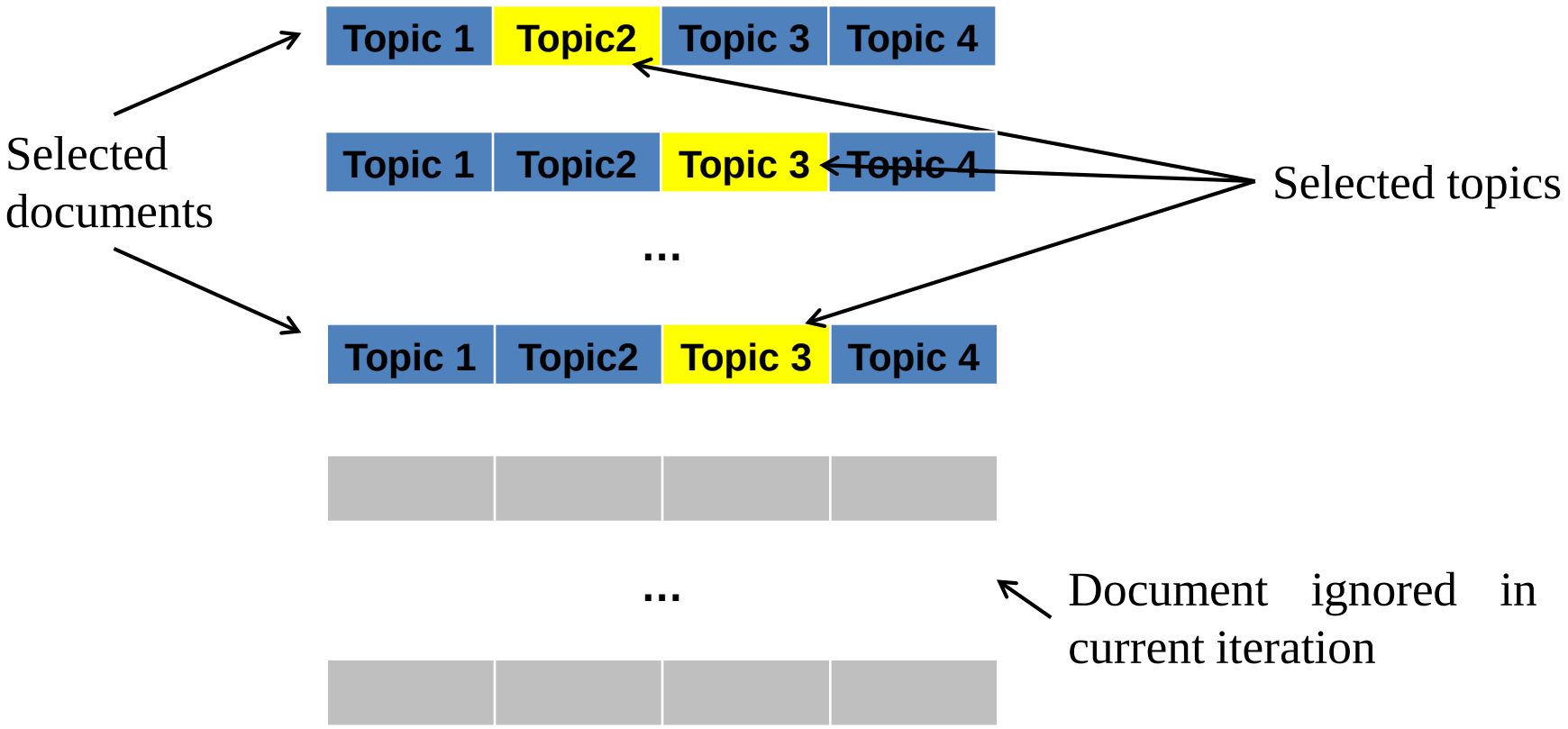
- Updating beliefs with large residuals accelerates convergence.

Belief Update Algorithm

- After each iteration
  - Sort $\boldsymbol{r_d}$ in a descending order for all documents;
  - Select several documents with the largest residuals;
  - For each selected document
    - Sort $\boldsymbol{r^k_d}$ in descending order;
    - Select several topic with the largest residual;
  - Update the corresponding μ^k_{ds};
  - Normalize the corresponding μ^k_{ds};

Belief Update Algorithm

- A Running Example



Parameter Estimation in A Bucket

- Topic distribution in the document *d*,

$$\theta_d^k = \frac{\mu_{d,\cdot}^k + \alpha_k}{\sum_{k'} \left(\mu_{d,\cdot}^{k'} + \alpha_{k'} \right)}$$

The prior of the topic distribution of the document d.

- Word distribution in the topic *k*,

$$\phi_w^k = \frac{\mu_{\cdot,\cdot,w}^k + \beta_w}{\sum_{w'} \left(\mu_{\cdot,\cdot,w'}^k + \beta_{w'} \right)}$$

The prior of the word distribution of the topic k.

Belief Calculation over Multiple Buckets

- Statistic of the previous consecutive (*m-1*) buckets
  - Word information accumulated from previous bucket

The number that the word w in all sentences of all documents in previous (m-1) buckets .

The belief that the word w in all sentences of all documents is assigned to the topic k in previous (m-1) buckets.

$$\Omega^{k}_{w}[m\text{-}1]=n_{.,.,w}[m\text{-}1]\mu^{k}_{.,.,w}[m\text{-}1]$$

- For the current bucket, $n_{.,-s,w}[m]\mu^{k}_{.,-s,w}[m]+\Omega^{k}_{w}[m\text{-}1]$ work as the delegates of $n_{.,-s,w}[m]\mu^{k}_{.,-s,w}[m]$

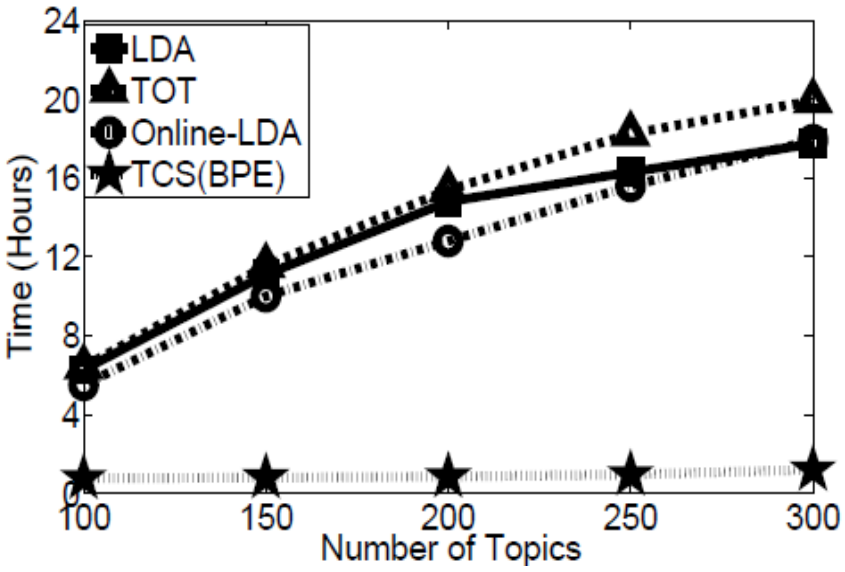
Bucket (m-1)		Bucket (m)
$\Omega^{k}_{w}[m\text{-}1]$	→	$n_{.,-s,w}[m]\mu^{k}_{.,-s,w}[m]+\Omega^{k}_{w}[m\text{-}1]$

Outline

- Motivations
- Problem Definitions
- Topic Crowd Service Model
- Pairwise Sketch
- Parameter Estimation
- Experimental Study
- Conclusion

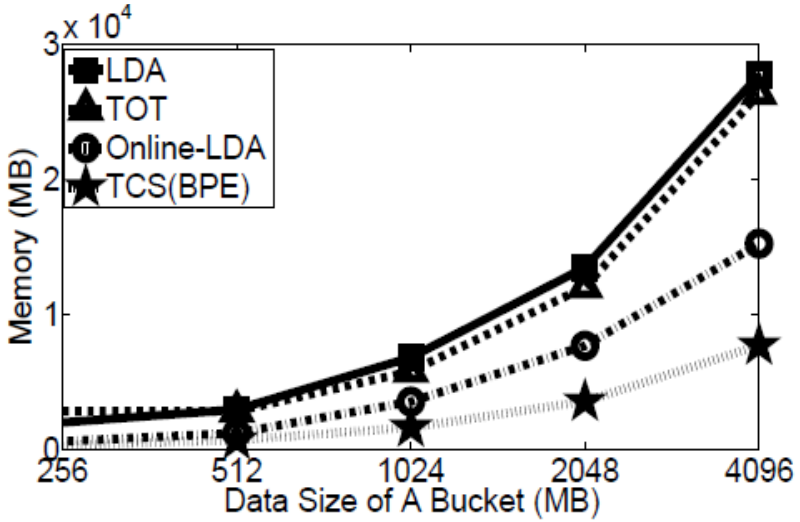
Efficiency

- The efficiency of TCS(BPE) is better than the other topic models:
  - TCS is a light-weight topic model.
  - TCS reduces the amount of data that need to be processed.
  - BPE reduces the document scope and topic scope that need to be scanned in each iteration.

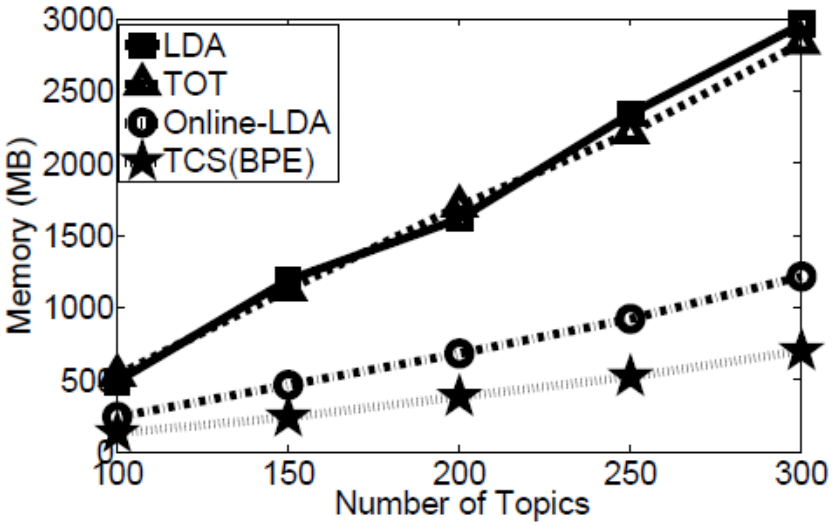


Memory Consumption

- The memory consumption of each topic model increases with the size of a bucket and the number of topics, respectively.

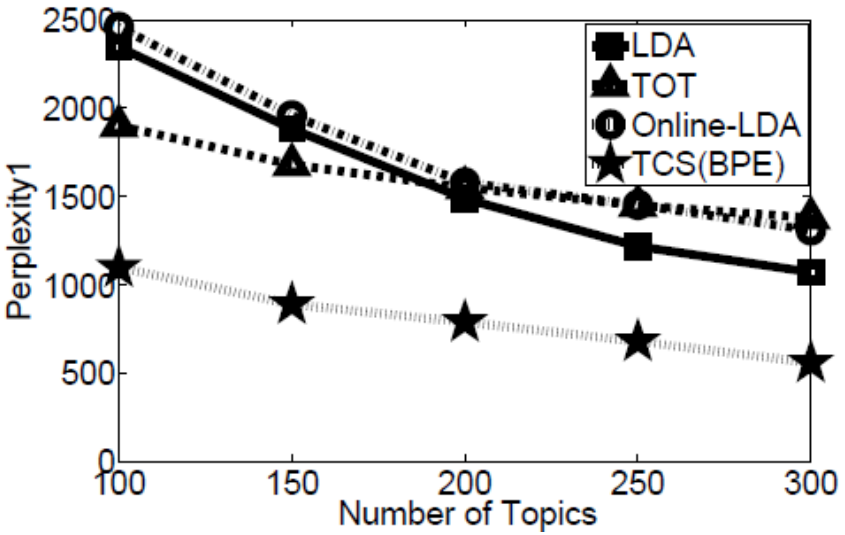


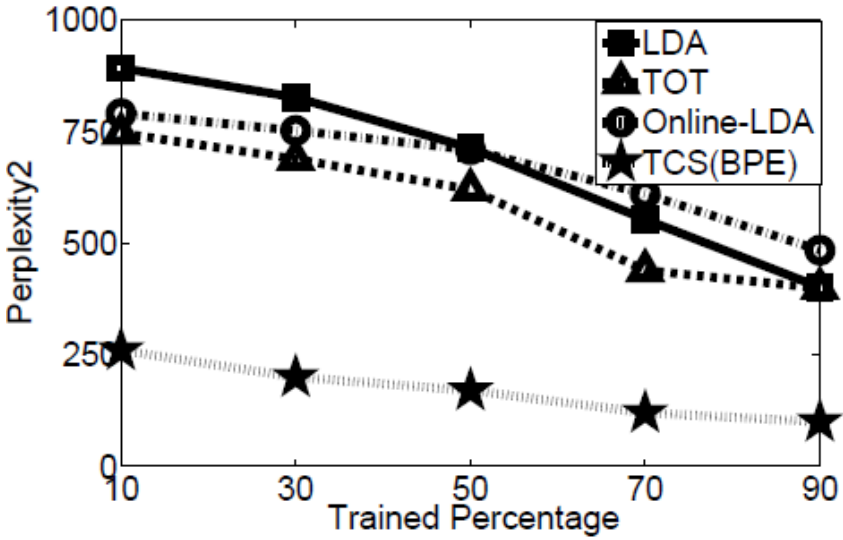
- pSketch consumes least memory
  - It reduces the amount of data that need to be processed



Effectiveness

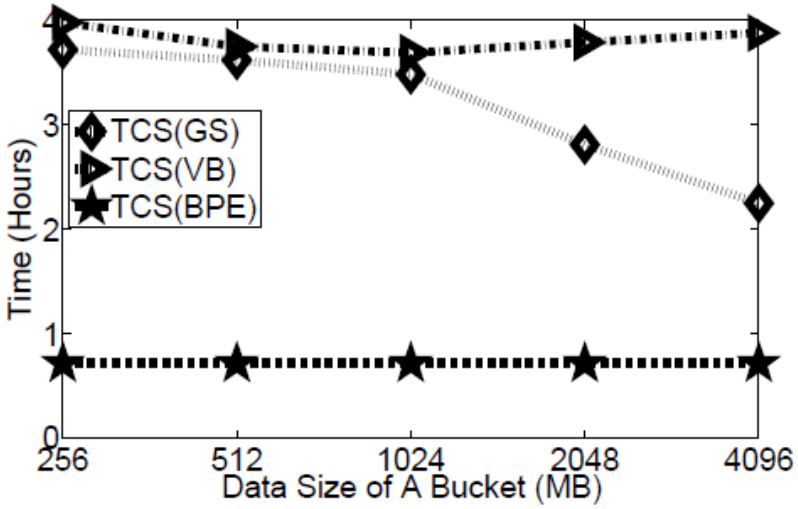
- TCS demonstrates good performance in terms of perplexity.
- Perplexity1 describes the held-out perplexity on the learned model .
- Perplexity2 is used to evaluate the effectiveness of prediction of the model.

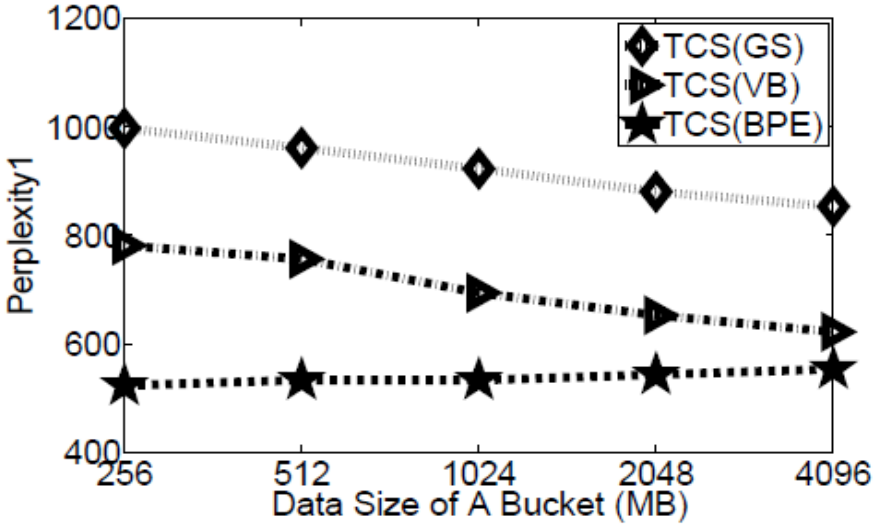




BPE v.s. GS and VB

- BPE is always the fastest one among the three methods.
  - GS and VB uses the variational Bayes and collapsed Gibbs sampling parameter estimation methods.
- BPE shows the best effectiveness when the data size is small.
  - The performance gain of BPE decreases when the data size increases.
  - The effectiveness of BPE is not worse than that of GS and VB .





Topic Evolution (*Sport Topic*)

- We compare the topics that are discovered from consecutive different buckets about the topic *sport*.

Bucket 1	Bucket 2	Bucket 3	Bucket 4	Bucket 5
Running	Running	Running	Running	Running
Basketball	Basketball	Swimming	Swimming	Swimming
Injury	Gym	Basketball	Gym	Gym
Shoe	Swimming	Gym	Football	Football
Gym	Injury	Football	Basketball	Basketball
Football	Football	Injury	Shoe	Shoe
Swimming	Shoe	Shoe	Injury	Injury

Outline

- Motivations
- Problem Definitions
- Topic Crowd Service Model
- Pairwise Sketch
- Parameter Estimation
- Experimental Study
- Conclusion

Conclusion

- Propose a new problem of discovering latent topics over massive crowd-oriented service data .
- Design the New probabilistic topic model to crowd-oriented service data.
  - Topic Crowd Service Model
- Propose an efficient solution to discover latent topics from crowd-oriented service data.
  - Pairwise Sketch (pSketch)
  - Bucket Parameter Estimation (BPE) Algorithm
- Show the effectiveness and efficiency of the proposed methods in extensive experimental results.

dank u
ju faleminderit
Tack
Asante
谢谢
Tak
mulţumesc
kiitos
Gracias
Salamat!
Terima kasih
Aliquam
Merci
Dankie
Obrigado
ありがとう
köszönöm
grazie
Aliquam
Go raibh maith agat
děkuji
Thank you
gam

Active Learning vs. Crowdsourcing

- There are main three differences

- Range of Applications
  - Active Learning: improving the accuracy of learning algorithms only
  - Crowdsourcing: enhancing the quality of any questions

- Quality of Feedback
  - Active Learning: assuming that workers is correct
  - Crowdsourcing: considering confidences of different crowd workers

- Task Assignment Strategy
  - Active Learning: no consideration of task assignment
  - Crowdsourcing: optimal task assignment result

Time Complexity of Algorithm 3

- The time complexity of Algorithm 3 is $O(|D| \times |I| \times |ASW|)$
  - |D| is the number of all documents including tasks and responses.
  - |I| is the number of iterations, which is set as a input parameter.
  - |ASW| is the average number of words among significant word-pairs in each document.

Effects of ρ_D and ρ_K

- ρ_D is the elected proportions of documents for message passing in each iteration.
- ρ_K is the elected proportions of topics for message passing in each iteration.
- Lower ρ_D and ρ_K make the faster BPE Algorithm since the number of updated topics and documents decreases.
- Default value of ρ_D and ρ_K are set as 0.5 since the perplexity is worse if ρ_D and ρ_K are lower than 0.5.

Effects of λ_T and λ_R

- λ_T is the significant (frequent) threshold for words among tasks in pSketch.
- λ_R is the significant (frequent) threshold for words among responses in pSketch.
- Higher λ_T and λ_R spend the fewer memory cost of pSketch and lead to the faster training process of BPE algorithm.
- Default value of λ_T and λ_R are set as 0.001 (1‰).

BPE vs. GS vs. VB

- Three different parameter estimation approach have the same inputs from pSketch, so we only compare their effectiveness in parameter estimation process.
- GS is stable but slow.
- VB is the faster one, but its accuracy is lower.
- BPE only updates elected proportions of documents and topics in each iteration, so is the fastest. Especially, the effectiveness of BPE is not lower than that of GS and VB.