Improved Search for Socially Annotated Data

VLDB 2008

Nikos Sarkas, Gautam Das, Nick Koudas

Talk by Cheng Li

Outline

- Introduction to Social annotation
- Motivation
- Ranking method (RadING)
- Parameter Optimization
- Search method
- Evaluation result
- Conclusion

Outline

- Introduction to Social annotation
- Motivation
- Ranking method (RadING)
- Parameter Optimization
- Search method
- Evaluation result
- Conclusion

Background

- Social annotation:
  - Users add to their personal collection a number of resources (e.g pics, videos, URLs) and assign a sequence of keywords to each resource, in order to facilitate searching and navigation.

Alice

The Twilight Saga: New Moon → URL of a Movie

annotated

Background

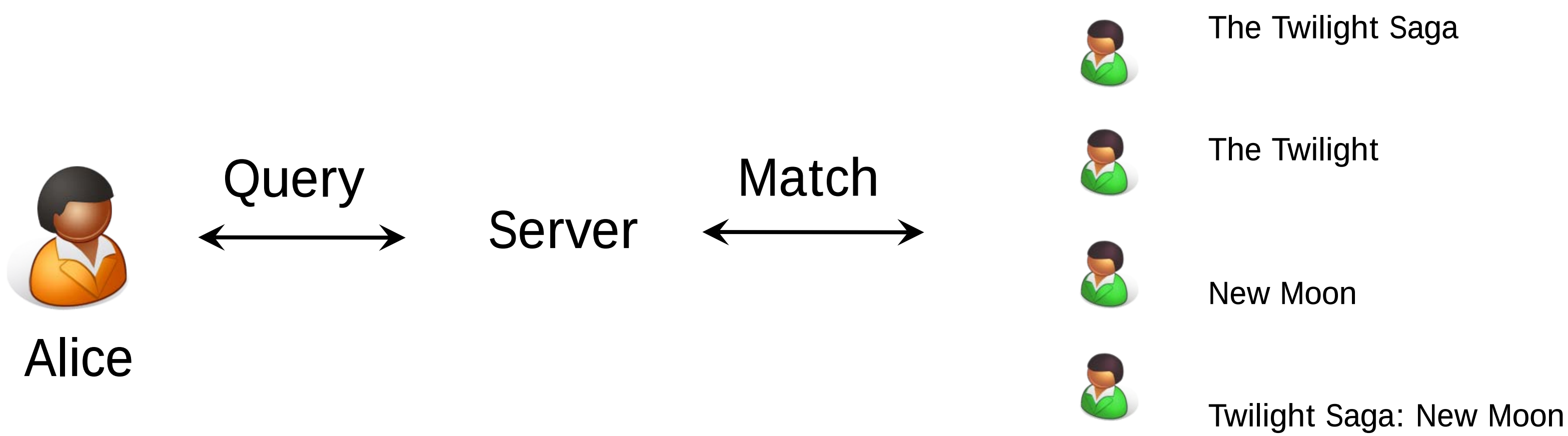
- Social annotation:
  - Users add to their personal collection a number of resources (e.g pics, videos, URLs) and assign a sequence of keywords to each resource, in order to facilitate searching and navigation.
- Characteristics of Social annotation:
  - Publicly available
  - Concise and accurate summary of resource content
  - Representative of non-textual resource
    - E.g. videos, pictures, music and etc.

Outline

- Introduction to Social annotation
- Motivation
- Ranking method (RadING)
- Parameter Optimization
- Search method
- Evaluation result
- Conclusion

Resource retrieval based on SA

- New searching paradigm
  - Compute the similarity of a query to a tag assignment of each resource in collection
  - Retrieve top-1 resource from the descending ranked list

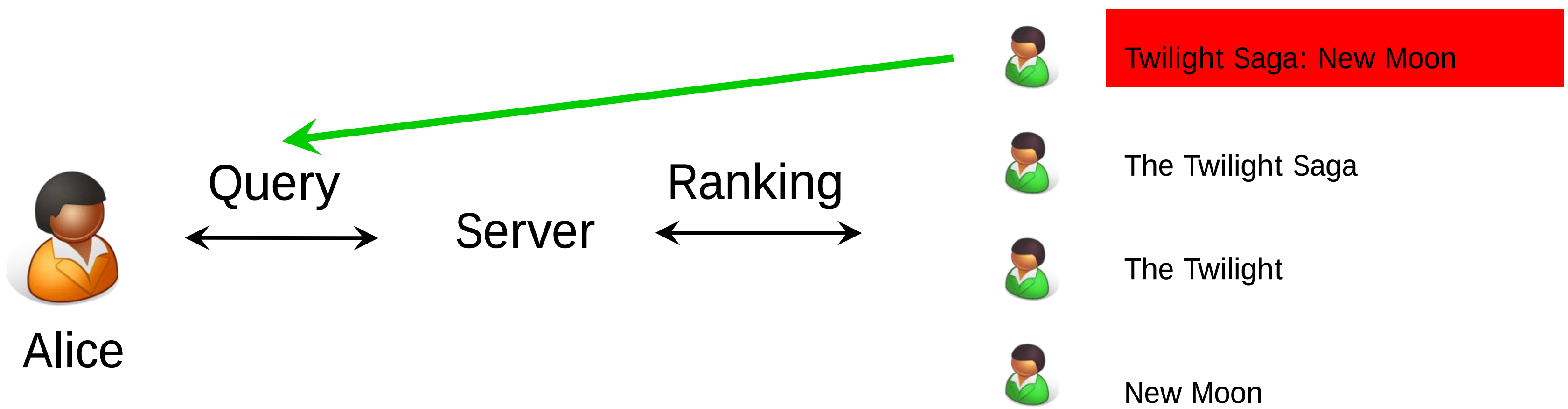


Query: The Twilight Saga: New Moon

annotated resources collection

Resource retrieval based on SA

- New searching paradigm
  - Compute the similarity of query to tags for a resource collections
  - Retrieval top-1 resource from the descending ranked list



Query: The Twilight Saga: New Moon

Descending list

Resource retrieval based on SA

- Given:
  - A keyword query: Q={t1,t2,...,tn}
  - Collection of tagged resources: R={R1,R2,...,Rn}
- Question?
  - how to rank R ?
- Solution:
  - Compute probability: p(R is relevant|Q)
  - Ranking items in descending order

Outline

- Introduction to Social annotation
- Motivation
- Ranking annotated data using Interpolated N-Grams (RadING)
- Parameter Optimization
- Search method
- Evaluation result
- Conclusion

Principled Ranking of annotated resources

- Applying Bayes' rule:

$$p(R \text{ is relevant}|\mathrm{Q}) = \frac{p(\mathrm{Q}|\mathrm{R} \text{ is relevant})p(\mathrm{R} \text{ is relevant})}{p(Q)}$$

  - p(R is relevant) is constant since it is indenpdent of query being posted
  - p(Q) is constant for all resources

$$p(R \text{ is relevant}|\mathrm{Q}) \propto \mathrm{p}(\mathrm{Q}|\mathrm{R} \text{ is relevant})$$

Properties of Social annotation

- Observation:
  - Distribution of tags converges to a heavy tailed distribution
    - Different users have a limited number of perspectives

p(Q|R is relevant) = p(Q is used to tag R)

The probability of a query containing the same keywords with R	The probability of a query being used to tag the resource R

  - The tag sequence in assignments are not orderless
  - Tags exhibiting strong tag co-occurrence patterns
    - i.e “mozilla browser“ identifies “firefox“

Probabilistic foundations

- Chain rule of probability

$$p(t_1,\ldots,t_l) = p(t_1)\,p(t_2 \mid t_1)\cdots p(t_l \mid t_1,\ldots,t_{l-1})$$

$$= \prod_{k=1}^{l} p(t_k \mid t_1,\ldots,t_{k-1})$$

  - The probability of a tag t_k appearing in the sequence depends on all of the preceding tags.

- Limitations of chain rule
  - Storage and computation overhead can not be addressed when the length of tag sequence increases.

Probabilistic foundations

- N-gram Models

$$p(t_k \mid t_1, \ldots, t_{k-1}) = p(t_k \mid t_{k-n+1}, \ldots, t_{k-1})$$

  - The probability of a tag t_k appearing in the sequence depends on the preceding subsequence with only the last n-1 tags.
  - 1- gram(unigram)

$$p(t_k \mid t_1, \ldots, t_{k-1}) = p(t_k)$$

  - 2-gram(bigram)

$$p(t_k \mid t_1, \ldots, t_{k-1}) = p(t_k \mid t_{k-1})$$

Question: How to estimate 2-gram probability $p(t_k \mid t_{k-1})$?

Estimation approach

- Maximum Likelihood Estimation
  - a popular statistical method used for providing estimates for the model's parameters.

- bigram with MLE:
  - The probability of a bigram t1, t2 (t2 follows t1):

$$p(t_2 \mid t_1) = \frac{c(t_1, t_2)}{\sum_t c(t_1, t)}$$

$c(t_1, t_2)$ The number of occurrences of the bigram in the history data

$\sum_t c(t_1, t)$ The sum of the occurrences of all different bigrams involving t1 as the first tag

Example of Estimation

- Assignments:

$t_1 t_2 t_3 \quad t_3 t_1 t_2 \quad t_2 t_3 \quad t_1 t_4$

- Bigram:

$t_1 t_2 \quad t_2 t_3 \quad t_3 t_1 \quad t_1 t_4$

1-gram	$P(t_i)$
t1	*3/4*
t2	*3/4*
t3	*3/4*
t4	*1/4*

Bigram	$c(t_1, t_2)$	$\sum_t c(t_1, t)$	$p(t_2 \mid t_1)$
$t_1 t_2$	*2*	*3*	*2/3*

Interpolation

- Limitation of bigram model with MLE
  - Trainning data is limited
  - If t1 and t2 fail to appear in adjacent positions
    - Then p(t1,t2)=0

- Example

Query(Q):

Saarbrucken(t1) snow(t2)

\+

Resource(R):

Saarbrucken

heavy snow

→

C(t1,t2) = 0

P(t2|t1)=0

Contradiction: R is not relevant to Q!

Question: How to compensate for this limitation?

Compensation by JM linear interpolation

- Jelinek-Mercer linear interpolation
  - Smooth technique
  - Assign a non-zero value

$$\underbrace{p(t_2 \mid t_1)}_{\substack{0 \\ >0}} = \lambda_2 \underbrace{\hat{p}(t_2 \mid t_1)}_{0} + \lambda_1 \underbrace{\hat{p}(t_2)}_{>0}$$

$$\lambda_1 + \lambda_2 = 1$$

$\hat{p}(t_2)$ The probability of t2 appearing in the training data

Question: if $\hat{p}(t_2) = 0$?

Compensation by JM linear interpolation

- Jelinek-Mercer linear interpolation
  - Smooth technique
  - Assign a non-zero value

$$p(t_2 \mid t_1) = \lambda_2 \ \hat{p}(t_2 \mid t_1) + \lambda_1 \ \hat{p}(t_2) + (1 - \lambda_1 - \lambda_2) \boxed{p_{bg}(t_2)}$$

$$0 \le \lambda_1, \lambda_2 \le 1, \qquad \lambda_1 + \lambda_2 \le 1$$

$p_{bg}(t_2)$ The background probability of t2 appearing in random text

$$p(t_1, \ldots, t_l) = \prod_{k=1}^{l} p(t_k \mid t_{k-1})$$

Outline

- Introduction to Social annotation
- Motivation
- Ranking method (RadING)
- Parameter Optimization
- Search method
- Evaluation result
- Conclusion

Parameter optimization

- Data set:
  - M assignments: $a_1,\ldots,a_i,\ldots,a_m$
  - Each assignment has k(i) tags: $t_{i1},\ldots,t_{ik(i)}$
  - All assignments comprised of $\boldsymbol{l}$ bigrams
- Likelihood function

$$L(\lambda_1,\lambda_2)=\sum_{i=1}^{l}\log(\underbrace{\lambda_2 p_{i2}+\lambda_1 p_{i1}+p_{i0}}_{p(t_{i2}\mid t_{i1})})$$

$$0\le\lambda_1,\lambda_2\le 1,\lambda_1+\lambda_2\le 1$$

Maximize likelihood function

- Likelihood function needs to be maximized:

$$L(\lambda_1, \lambda_2) = \sum_{i=1}^{l} \log(\lambda_2 p_{i2} + \lambda_1 p_{i1} + p_{i0})$$

$$0 \le \lambda_1, \lambda_2 \le 1, \lambda_1 + \lambda_2 \le 1$$

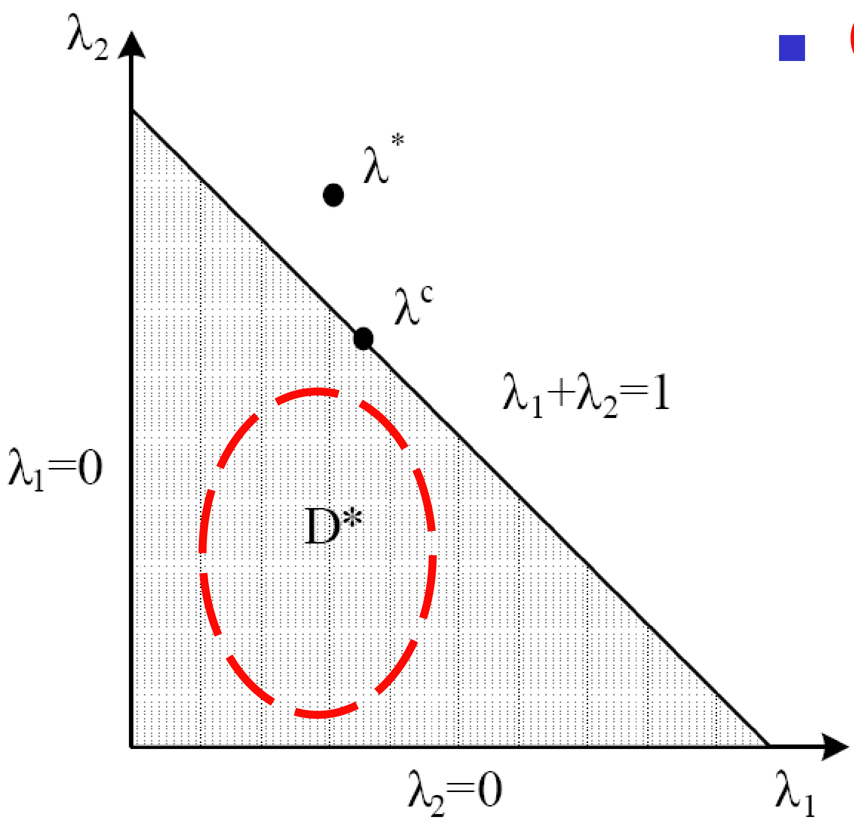
- Denote

$\lambda^* = (\lambda_1^*, \lambda_2^*)$ Global maximum of $L(\lambda_1, \lambda_2)$

$D^* : 0 \le \lambda_1, \lambda_2 \le 1, \lambda_1 + \lambda_2 \le 1$ Constrained domain

Maximize $L(\lambda_1, \lambda_2)$ $\Longleftrightarrow$ Find $\lambda^*(\lambda_1^*, \lambda_2^*)$

Maximize likelihood function



$$D^*: 0 \le \lambda_1, \lambda_2 \le 1, \lambda_1 + \lambda_2 \le 1$$

- Questions
  - Unbounded: λ^* does not exist

$$\lim_{\lambda_2 \to \infty} L(\lambda_1, \lambda_2) = \infty$$

$$\lim_{\lambda_1 \to \infty} L(\lambda_1, \lambda_2) = \infty$$

  - Bounded: λ^* exists but outside D*

$$\lambda^* \notin D^*$$

Maximize likelihood function

- Observation
  - $L(\lambda_1, \lambda_2)$ is a concave function.
- Property of concave function
  - If f is concave, any point that is a local maximum is also a global maximum.

$\boxed{\lambda^c = (\lambda_1^c, \lambda_2^c) \quad \text{Local maximum in} \quad D^*}$

Maximize $L(\lambda_1, \lambda_2)$ $\Longleftrightarrow$ Find $\lambda^*(\lambda_1^*, \lambda_2^*)$

$\Longleftrightarrow$ Locate $\lambda^c(\lambda_1^c, \lambda_2^c)$ in D^*

EM algorithm

- Expectation-Maximization
  - *A good choice for optimizing the likelihood function and setting parameters*
  - *Iterative computation to increase the value of likelihood function in constrained domain D**
  - *Finally, converges to a local maximum in D**
- However
  - Hundreds of millions of resources
  - Large number of assignments
  - Its convergence is very slow.

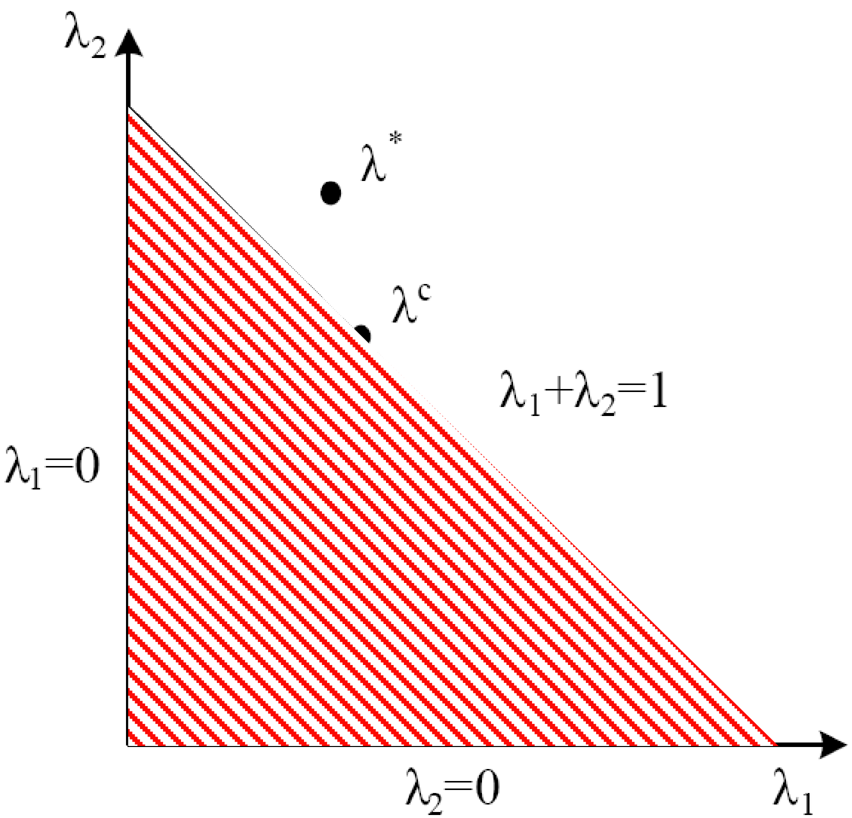
Optimization framework

How to locate the local maximum?

Solution

Unconstrained Optimization Methods for Constrained Optimization

Bounded likelihood function



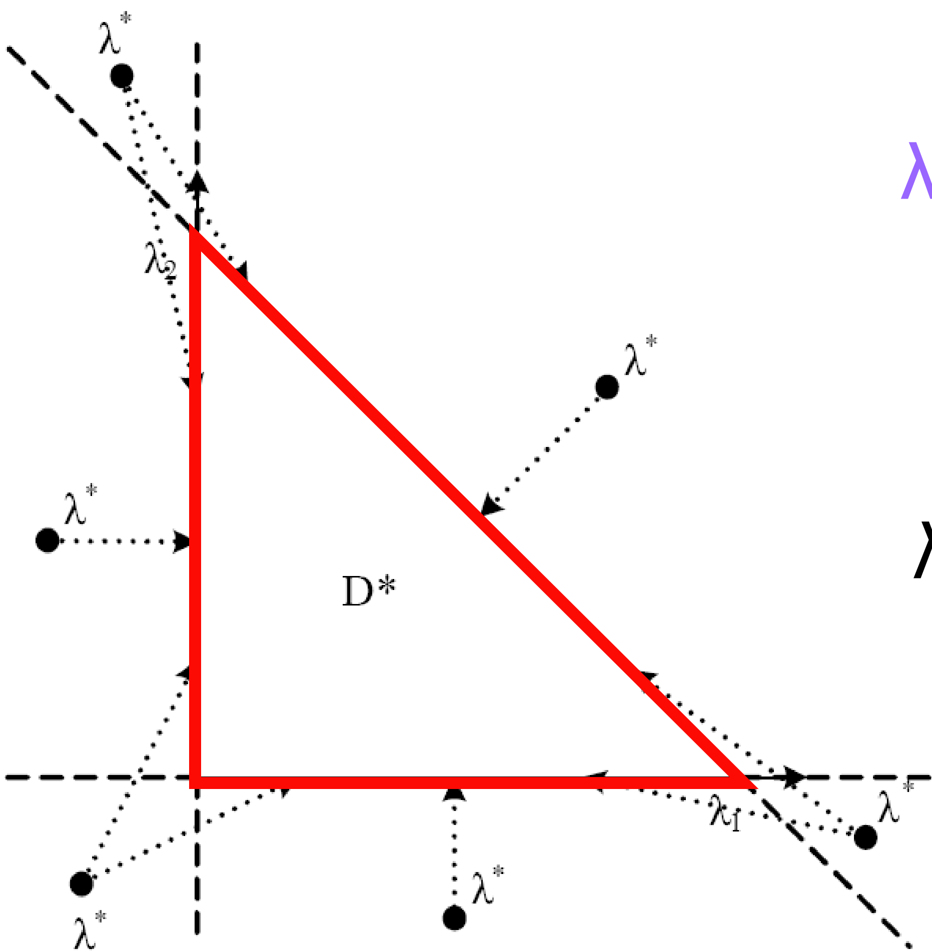
λ^* lies inside D^*

- $\lambda^* = \lambda^c$
- Two dimensional numerical optimization algorithm(2D)
- Search inside D^*

λ^* lies outside D^*

- One dimensional numerical optimization algorithm
- Search along the boundary

Bounded likelihood function (2)



λ* lies inside D*

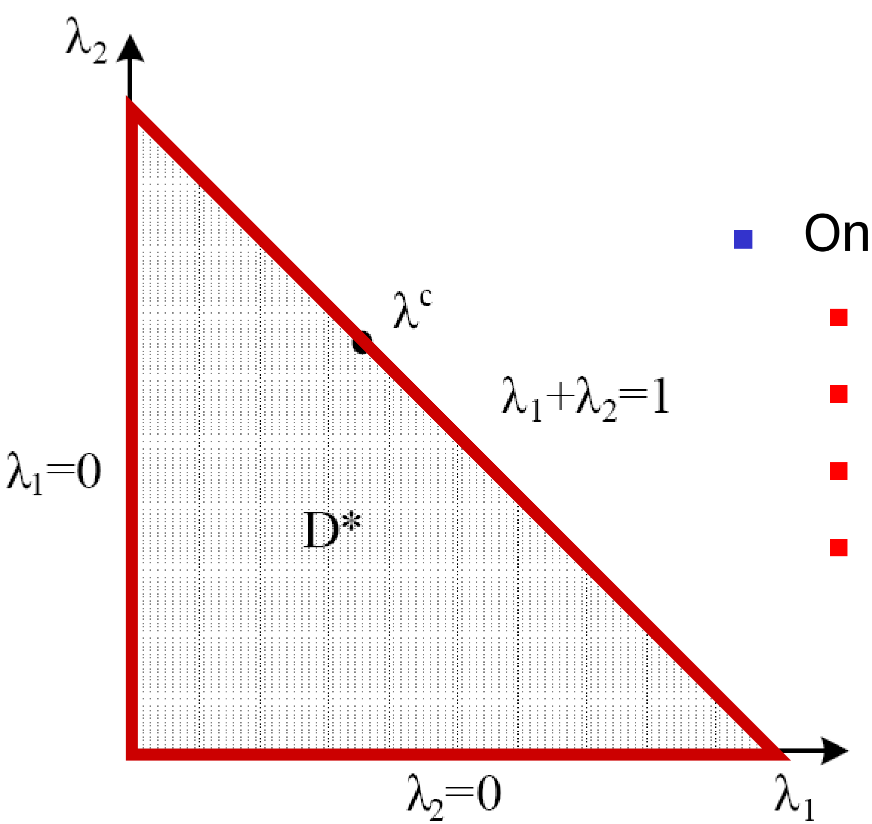
- Two dimensional numerical optimization algorithm(2D)
- Search inside D*

λ* lies outside D*

- $\lambda^* <> \lambda^c$
- One dimensional numerical optimization algorithm(1D)
- Search along the boundary

Unbounded likelihood function

$$L(\lambda_1,\lambda_2)=\sum_{i=1}^{l}\log(\lambda_2 p_{i2}+\lambda_1 p_{i1}+p_{i0})$$



- One dimensional numerical optimization
  - If any $p_{i2}<0, p_{i1}>0$ $(\lambda_2, \lambda_1)=(0,1)$
  - If any $p_{i2}>0, p_{i1}<0$ $(\lambda_2, \lambda_1)=(1,0)$
  - If any $p_{i2}>0, p_{i1}>0$ $\lambda_2+\lambda_1=1$
  - ....

RadING optimization framework

- Protocol
  - 1. If $L(\lambda_1,\lambda_2)$ is unbounded, use 1D optimization to locate λ^c along the boundary of D^*
  - 2. If bounded, apply a 2D algorithm to identify the global maximum inside D^*
  - 3. If λ^* *not inside* D^* , search λ^c along the boundary of D^*
- Incremental Maintenance
  - Update when new assignments exceeds a threshold
  - It is the same procedure as optimization

Outline

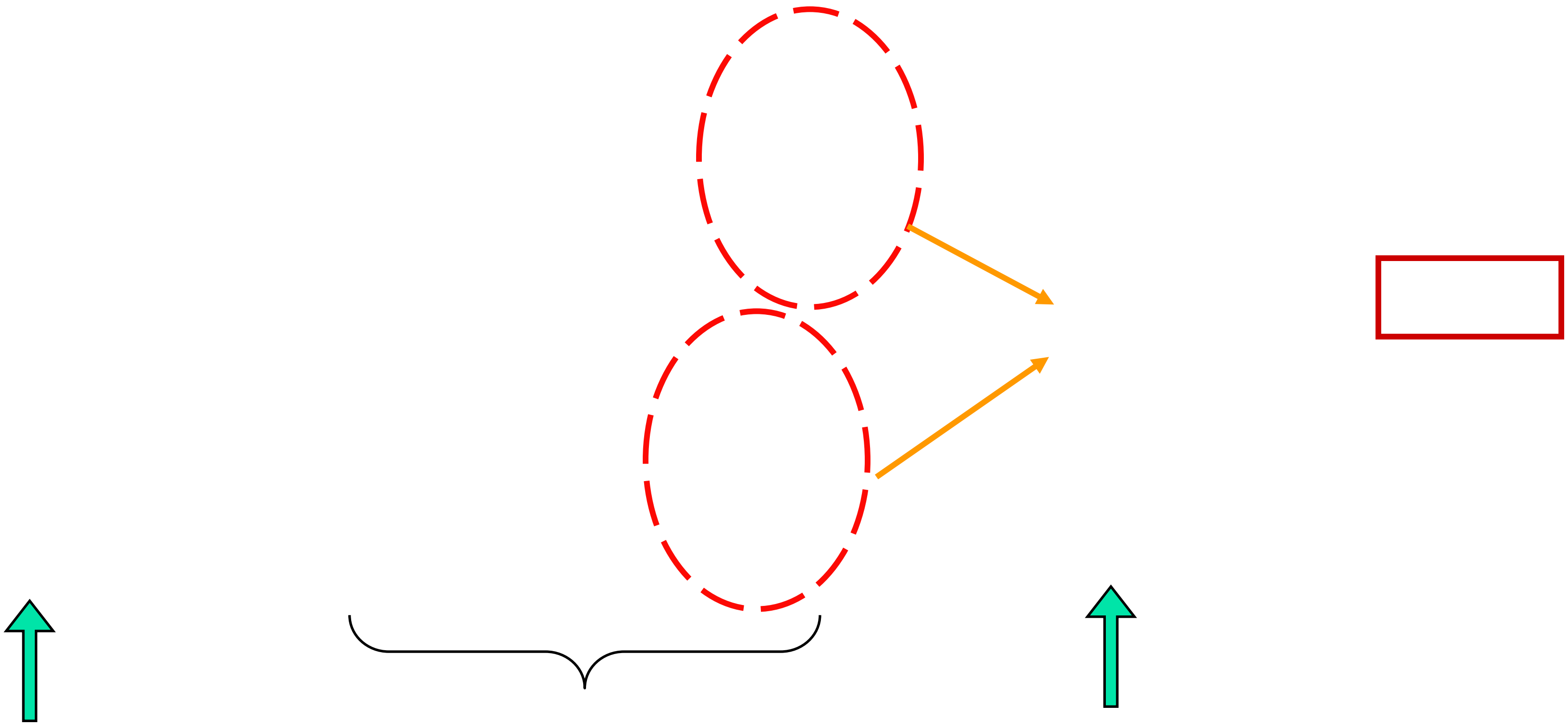
- Introduction to Social annotation
- Motivation
- Ranking method (RadING)
- Parameter Optimization
- Search method
- Evaluation result
- Conclusion

Searching(1)

- Resources: {R1,R2,R3,R4}
- Tags:{<s>,t1,t2}
- S: a start assignment
- Query: Q=t1,t2
- Bigrams:
  - (t1|<s>), (t2|<s>), (t1|t2), (t2|t1)
  - The probability of a query t1, t2 used to tag R:

$$p(Q) = p(t_1, t_2 \mid <s>) = p(t_1 \mid <s>)\, p(t_2 \mid t_1)$$

Searching(2)



Social

annotation

Bigram Ranking by RadING

Compute top-k relevant resources list

Outline

- Introduction to Social annotation
- Motivation
- Ranking method (RadING)
- Parameter Optimization
- Search method
- Evaluation result
- Conclusion

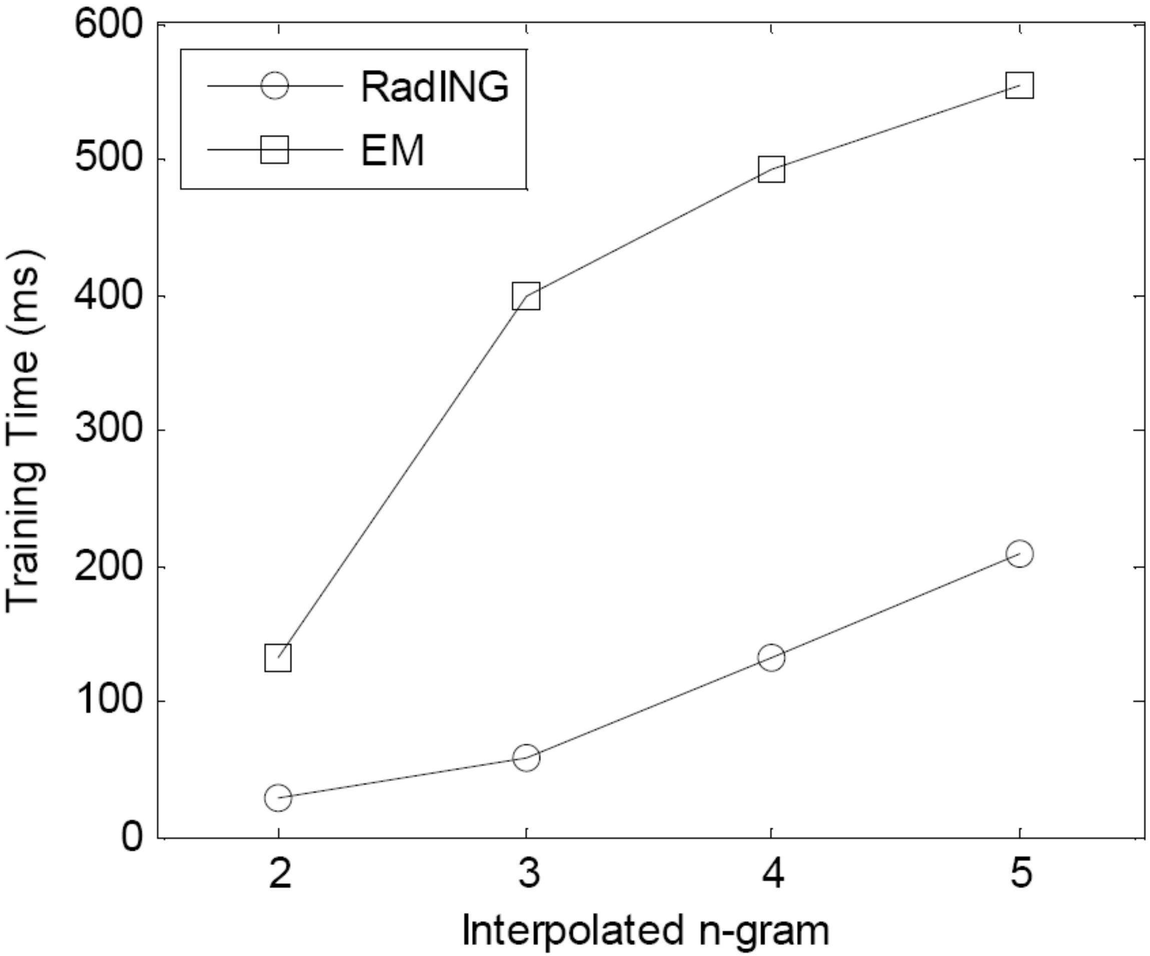
Experimental Evaluation(1)

- Data set from del.icio.us

User	Resource (URL)	Assignment
567,539	24,245,248	70,658,851

- Efficiency
  - Consider the training time
  - EM algorithm vs RadING
- Effectiveness Ranking quality
  - Interpolated grams vs plain gram
  - RadING vs Tf/idf Ranking

Optimization efficiency



- EM algorithm
  - A standard method for optimizing and setting parameters
  - An alternative of RadING

Experimental Evaluation(2)

- Data set from del.icio.us

User	Resource (URL)	Assignment
567,539	24,245,248	70,658,851

- Efficiency
  - Consider the training time
  - Comparison between EM algorithm and RadING
- Effectiveness Ranking quality
  - Interpolated grams vs non-interpolated (plain) gram
  - RadING vs Tf/idf Ranking

Ranking Effectiveness(1)

- Precision@10 performance
  - *How many items of the top-10 retrieved results are relevant?*

Better results

Query	$I2g$	$I3g$	$2g$	$3g$
birthday gift ideas	7.4	7.4	3.6 (4)	1.0 (1)
college blog	7.6	7.6	6.6 (10)	7.6 (10)
trigonometric formulas	7.9	7.9	0.9 (1)	0.9 (1)
stock market bubble	8.4	8.4	1.0 (1)	0.0 (1)
sea pictures	6.7	6.3	2.1 (3)	0.9 (1)

- *I2g: interpolated bigram*
- *I3g: interpolated trigram*
- *2g: non-interpolated bigram*
- *3g: non-interpolated trigram*

Ranking Effectiveness(2)

Better results

Query	$I2g$ (RadING)	Tf/Idf	Tf/Idf+
birthday gift ideas	7.4	2.8	5.8
college blog	7.6	5.9	4.9
trigonometric formulas	7.9	6.3	5.2
stock market bubble	8.4	5.1	6.1

- *Tf/Idf and Tf/Idf+: widely used ranking methods*

Outline

- Introduction to Social annotation
- Motivation
- Ranking method (RadING)
- Parameter Optimization
- Search method
- Evaluation result
- Conclusion

Summary

- Ranking annotated data using Interpolated N-Grams
  - RadING
  - A search and resource ranking methodology
- Optimization Framework
  - Parameters setting
  - Incremental maintenance
- Evaluation results
  - Efficiency
  - Effectiveness

Weakness

- Scalability
  - RadING works well in bigram and trigram
  - Bad performance for high order n-gram
- Accuracy of Linear Interpolation
  - Result may get worse by Interpolation
  - It may not reflect the reality
- User perspective diversity
  - RadING finds the similar term to the query but fails to get relevant terms with different assignments
- Potential threat
  - Malicious annotation have a good opportunity to harm the search quality

Questions?

Parameter optimization

- Held-out data:
  - m assignments: a1,...,ai,...,am
  - Each assignment has k(i) tags: $t_{i1},\ldots,t_{ik(i)}$
- Log likelihood of an assignment:

$$\log p(a_i) = \log \prod_{j=1}^{k(i)} p(t_{ij} \mid t_{i(j-1)}) = \sum_{j=1}^{k(i)} \log p(t_{ij} \mid t_{i(j-1)})$$

- Log likelihood function of all assignments:

$$\log \prod_{i=1}^{m} p(a_i) = \sum_{i=1}^{m} \log p(a_i) = \sum_{i=1}^{m} \sum_{j=1}^{k(i)} \log p(t_{ij} \mid t_{i(j-1)})$$

Assignments are generated independently by different users.

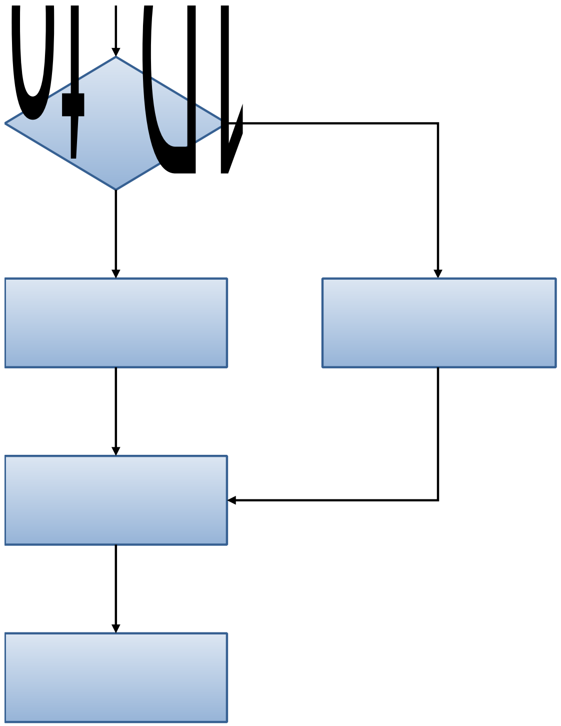
Parameter optimization

- Ease annotation using bigram model
  - Assignments are comprised by *l* bigrams t_{i1}, t_{i2}

$$\log \prod_{i=1}^{m} p(a_i) = \sum_{i=1}^{l} \log p(t_{i2} \mid t_{i1})$$

$$\left.\begin{aligned} p_{i2} &= \hat{p}(t_{i2} \mid t_{i1}) - p_{bg}(t_{i2}) \\ p_{i1} &= \hat{p}(t_{i2}) - p_{bg}(t_{i2}) \\ p_{i0} &= p_{bg}(t_{i2}) \end{aligned}\right\} \quad p(t_{i2} \mid t_{i1}) = \lambda_2 p_{i2} + \lambda_1 p_{i1} + p_{i0}$$

RadING optimization framework



Related work on SA

- PageRanking algorithm
  - Not scalable
- Machine learning approach
  - Limited to web pages
  - Scalability and updates?
- Ranking in neighborhoods
- Analysis and modeling SA
  - The distribution of tags converges rapidly
  - Co-occurrence patterns