Topic-level Random Walk Through Probabilistic Model

Zi Yang, Jie Tang, Jing Zhang, Juanzi Li, Bo Gao
KEG, DCST
Tsinghua University
4/4/2009

Search Engine

query

Keyword Matching

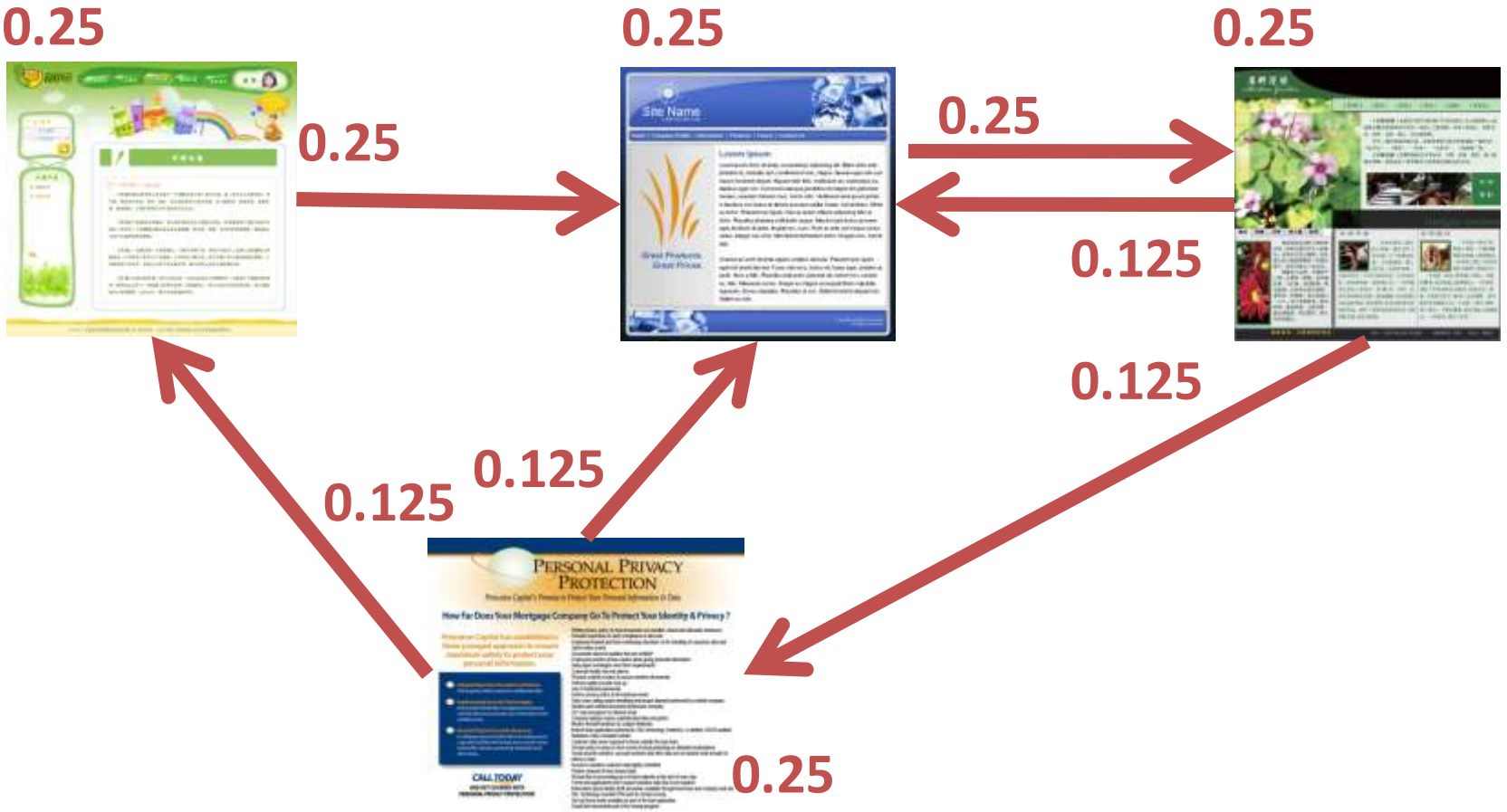
Document collection

Match!

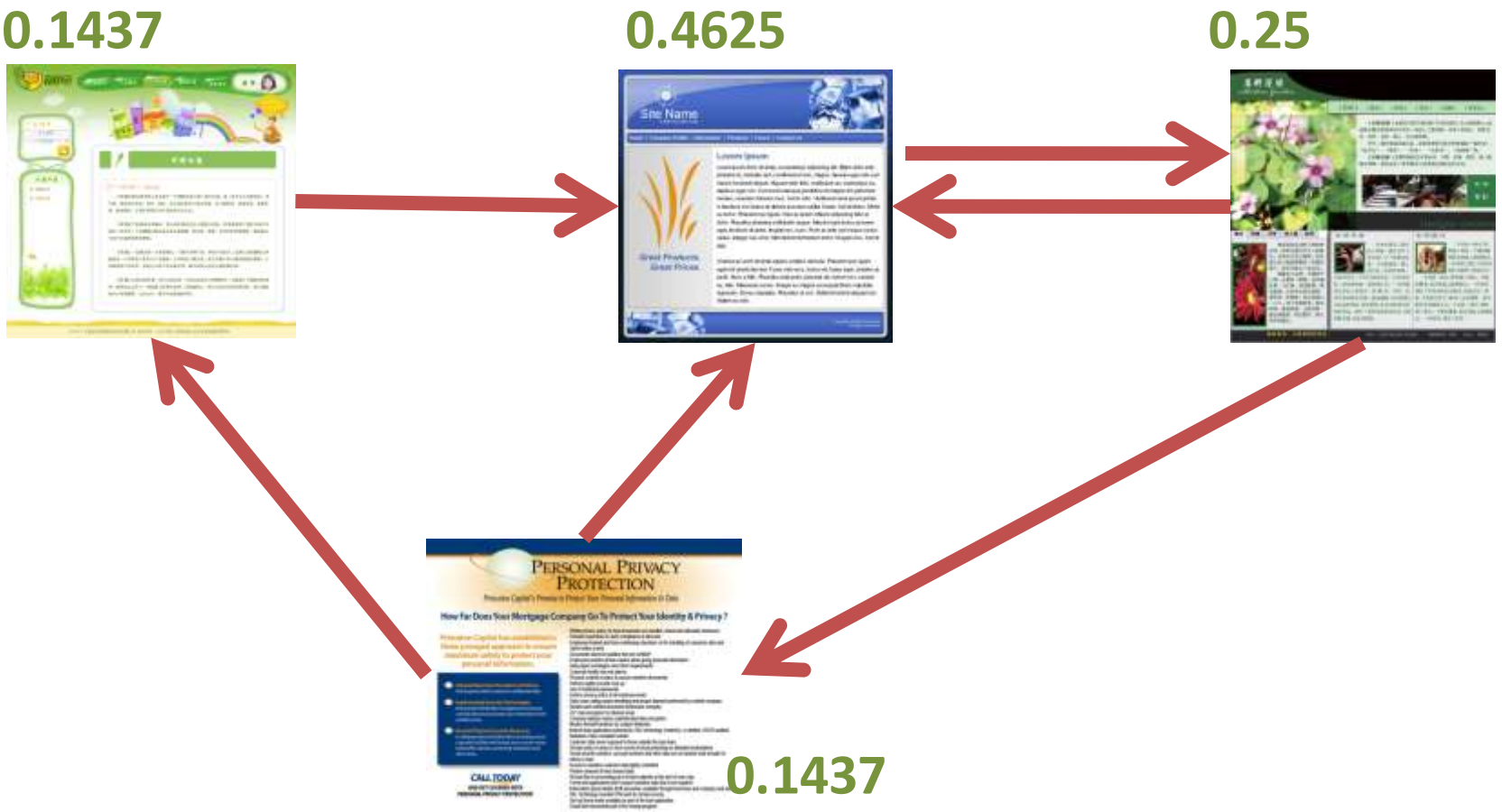
Language model
Vector space model

Sorted result

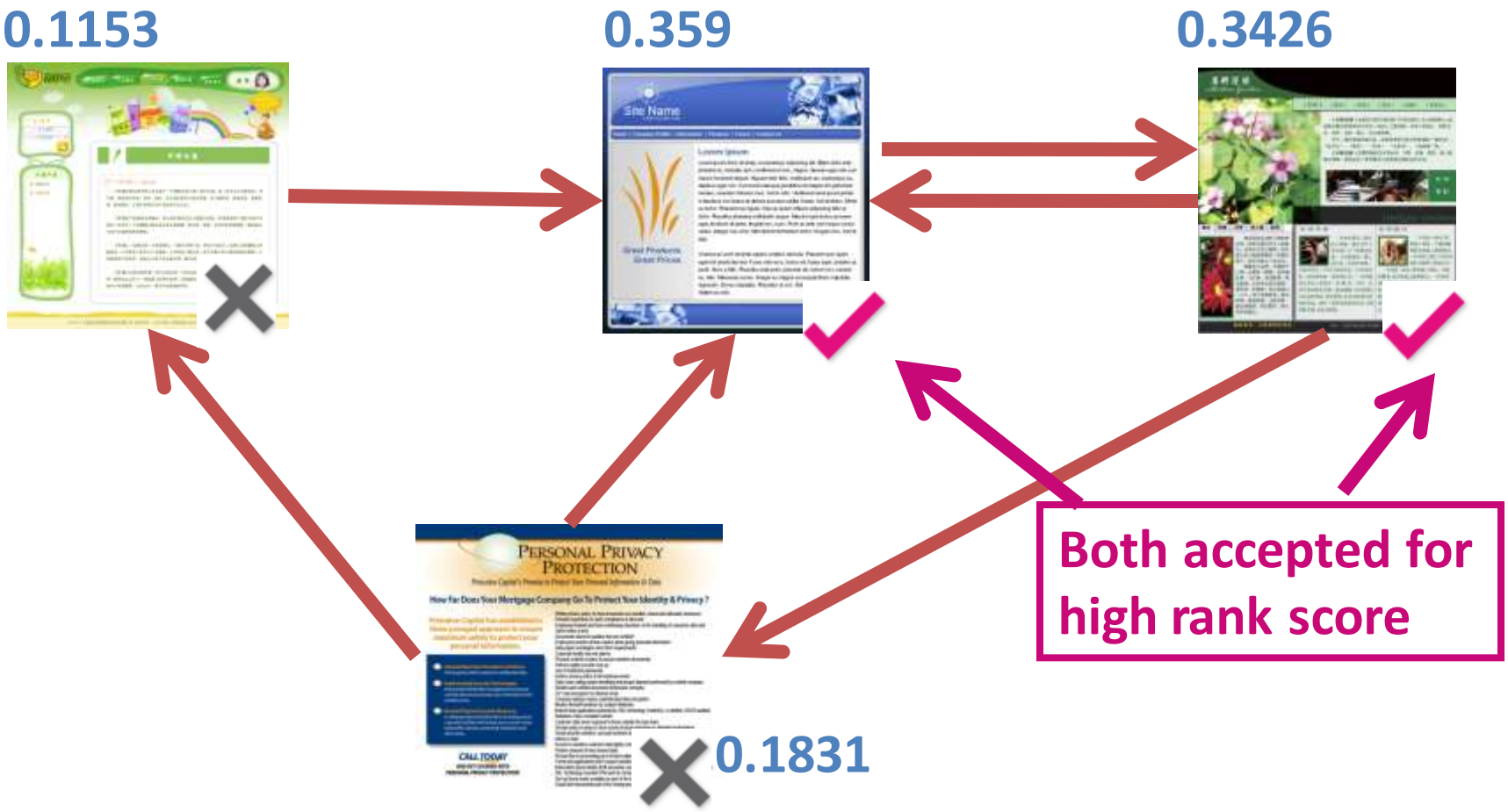
Random Walk


0.25
0.25
0.25
0.25
0.25
0.125
0.125
0.125
0.125
0.25
Site Name
PERSONAL PRIVACY PROTECTION
CALL TODAY

Random Walk

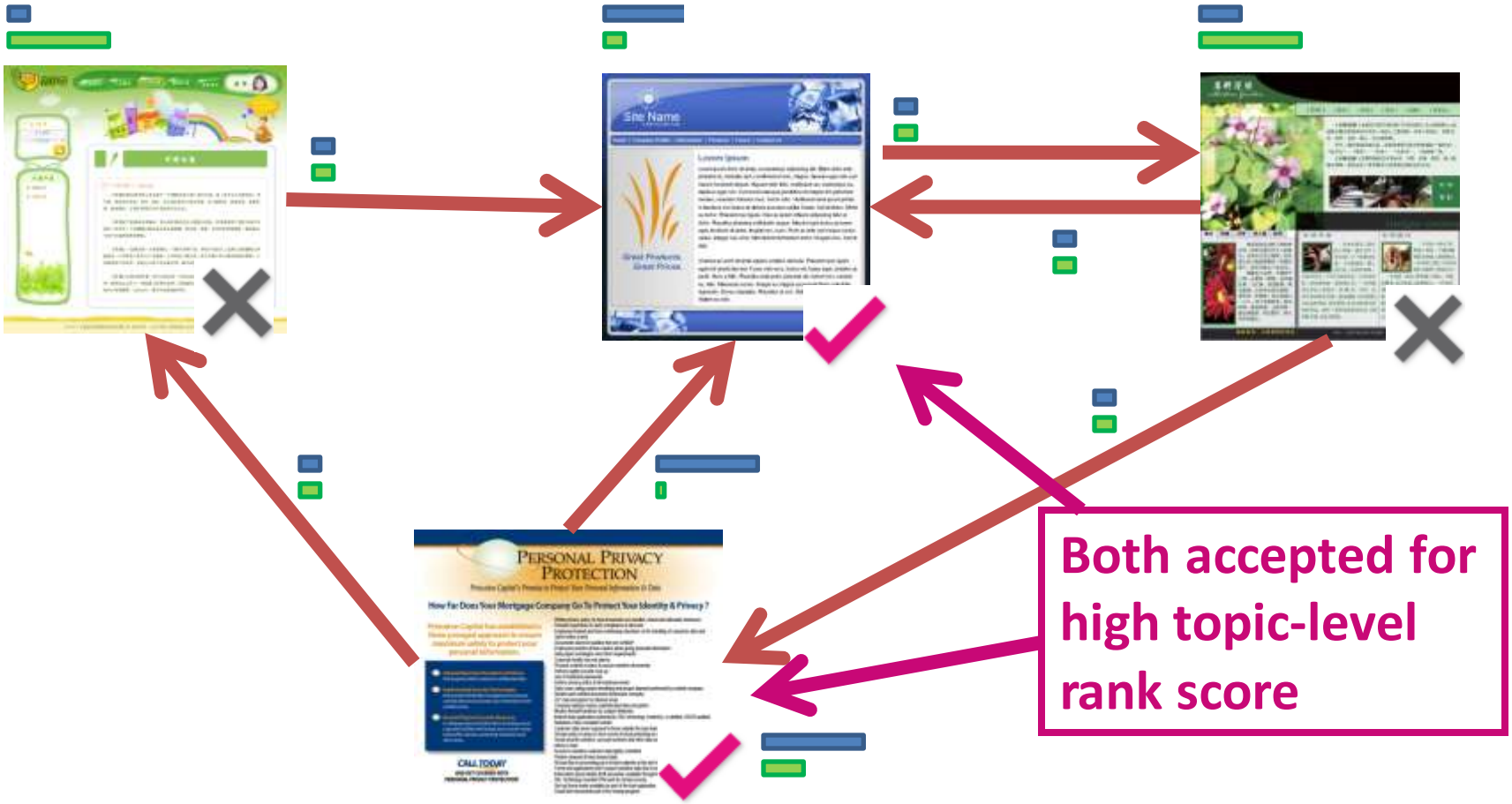

0.1437
0.4625
0.25
Site Name
PERSONAL PRIVACY PROTECTION
0.1437

Random Walk


0.1153
0.359
0.3426
Site Name
PERSONAL PRIVACY PROTECTION
0.1831
Both accepted for high rank score

Topic-level Random Walk

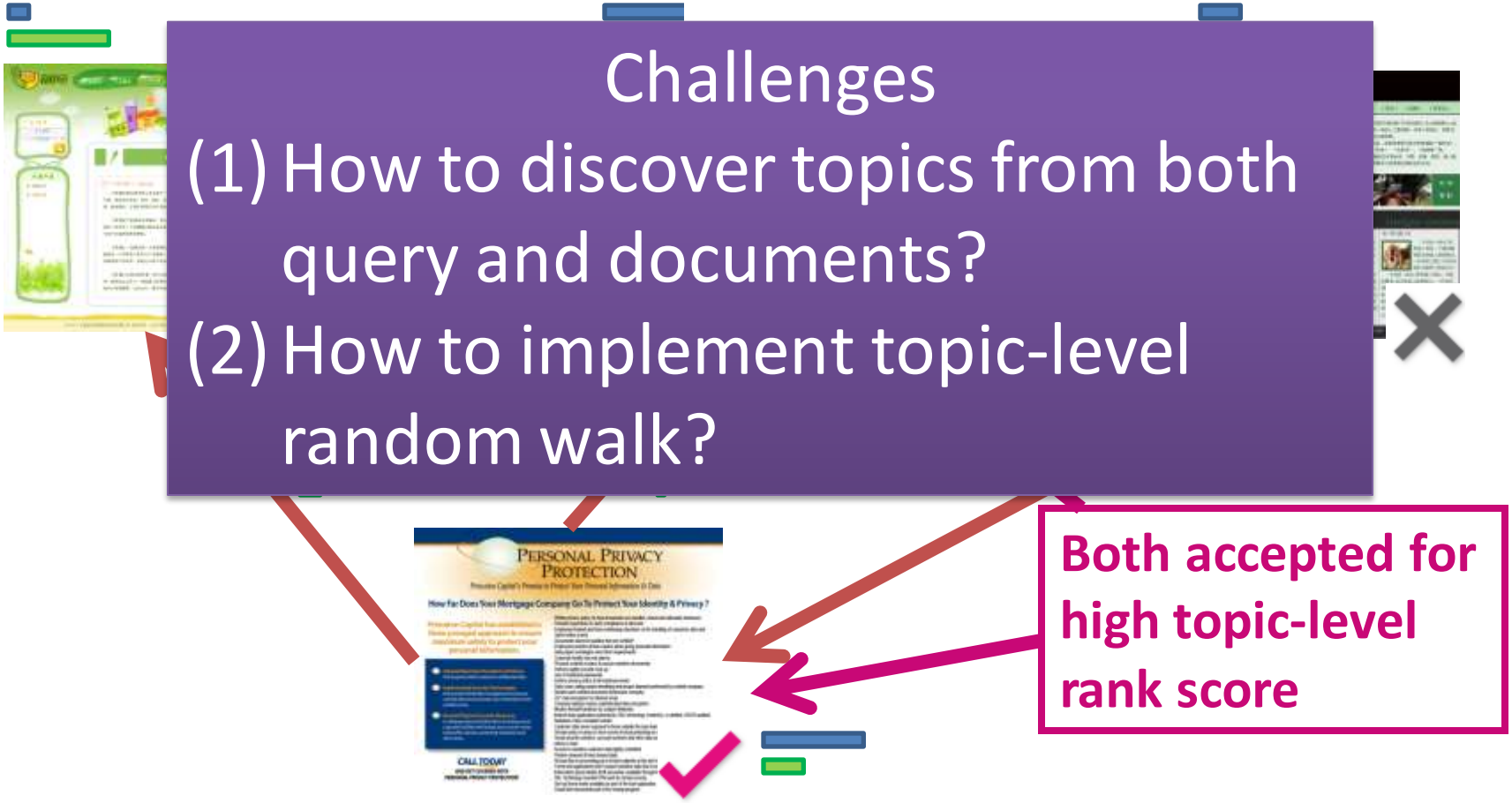
Importance score of the page for topics
e.g. data mining and machine learning



Both accepted for high topic-level rank score

Topic-level Random Walk

Importance score of the page for topics
e.g. data mining and machine learning



Outline

- **Related work**
- Our approach
  - Topic modeling
  - Topic-level random walk
  - Search with topics
- Experimental results
- Conclusion

Related Work

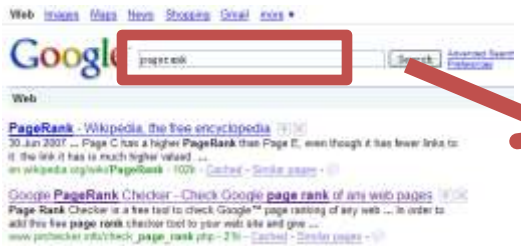
- Search with keywords
  - Language Model [Zhai, 01], VSM, etc.
- Random walk
  - PageRank [Page, 99], HITS [Kleinberg, 99],
  - Topic-sensitive PageRank [Haveliwala, 02], Topical PageRank [Nie, 2006], etc.
- Search with semantic topics
  - LSI [Berry,95], pLSI [Hofmann, 99], LDA [Blei,03] [Wei, 06], etc.

Outline

- Related work
- **Our approach**
  - **Topic modeling**
  - **Topic-level random walk**
  - **Search with topics**
- Experimental results
- Conclusion

Approach Overview

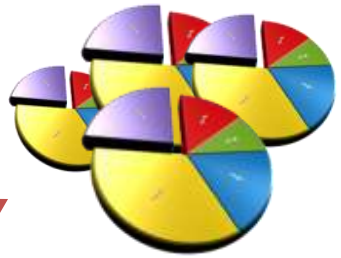
query



Topic distribution analysis

Topic-level importance analysis

Document collection

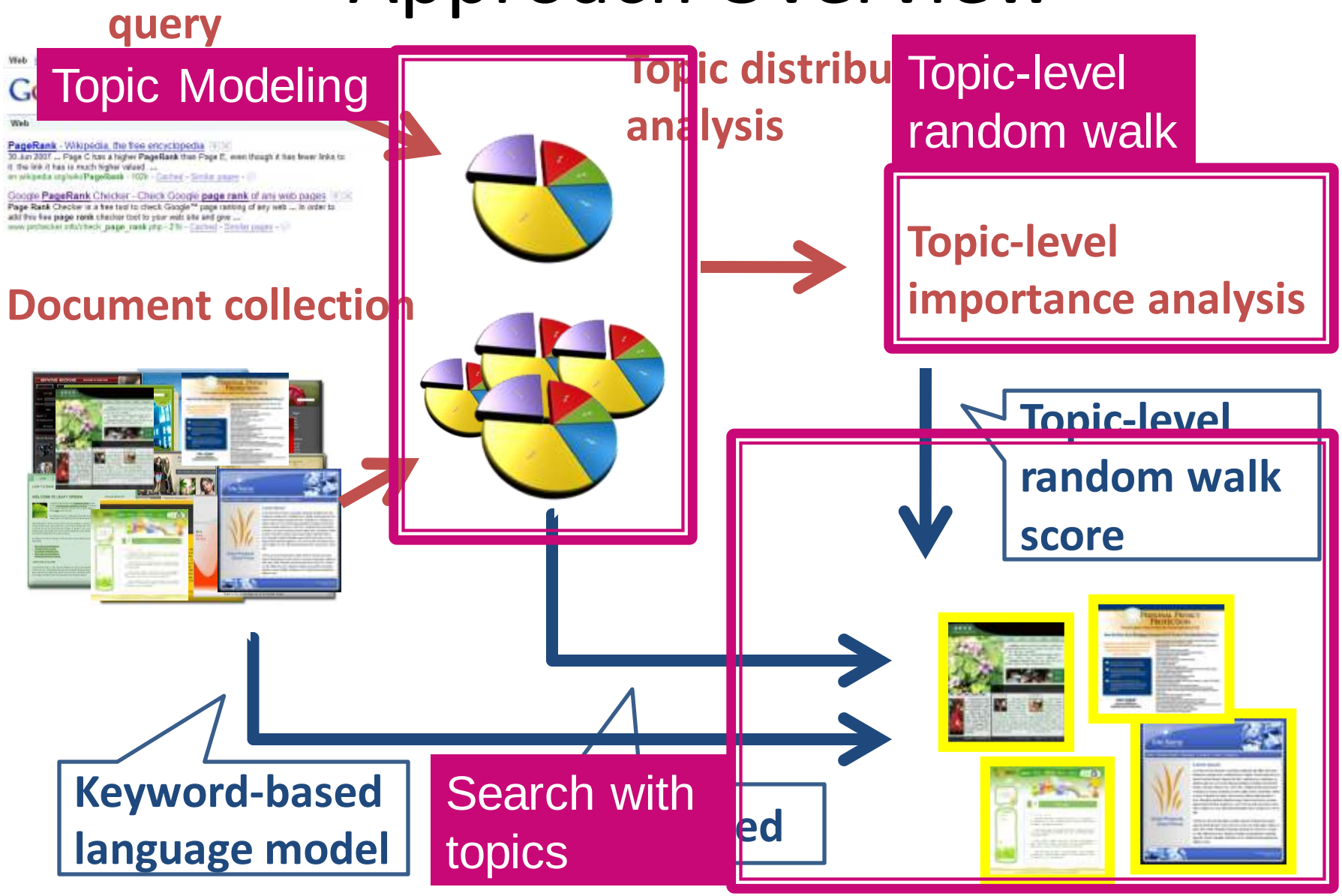


Topic-level random walk score

Keyword-based language model

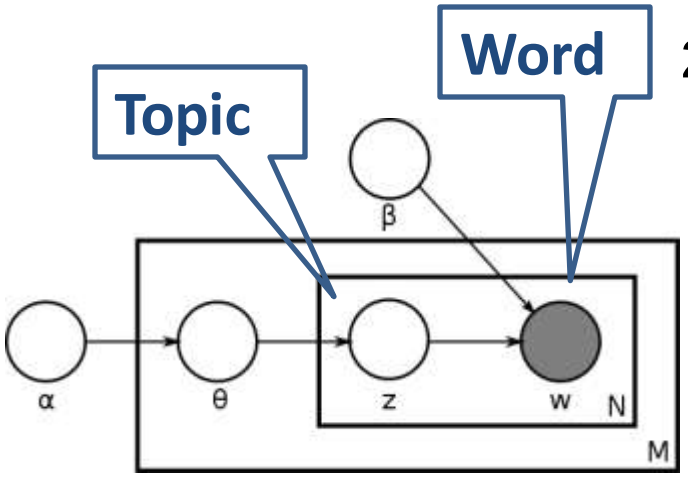
Topic-related

Approach Overview



Topic Modeling

- Automatically find topics in documents
  - LDA



1. For each topic z, draw φ_z respectively from Dirichlet prior β;
2. For each document d:
   Draw θ_d from Dirichlet prior α;
   For each word w_{di} in document d:
   draw a topic z_{wd} from multinomial distribution θ_d;
   draw a word w_{di} from multinomial distribution $\varphi_{z_{di}}$;

- Automatically assign topics for the query
  - Inference

Topic-level Random Walk

- Transition probability

$$P(d_l \mid d_k, z_i) = \frac{1}{\left|O(d_k)\right|}$$

$$P(d_l, z_j \mid d_k, z_i) = P(z_j \mid d_l) P(z_i \mid d_k)$$

- Ranking score

$$r[d, z_i] = \lambda \frac{1}{\left|D\right|} P(z_i \mid d) + (1-\lambda) \sum_{d':d' \to d} \left[\gamma P(d \mid d', z_i) + (1-\gamma) \frac{1}{\left|T\right|} \sum_{j \neq i} P(d, z_i \mid d', z_j) \right]$$

Search

- Score combined with language model
  - Proposed 1: TPR+

$$S_{\mathrm{TPR}+}(d,q) = \left[(1-t)P_{\mathrm{LM}}(q \mid d) + t\prod_{w\in Q}\sum_{z\in T} P(w \mid z)\cdot P(z \mid d)\right]\cdot\prod_{w\in Q}\sum_{z\in T} r[d,z]\cdot P(w \mid z)$$

  - Proposed 2: TPR*

$$S_{\mathrm{TPR}*}(d,q) = P_{\mathrm{LM}}(q \mid d)\cdot\prod_{w\in Q}\sum_{z\in T} P(w \mid z)\cdot P(z \mid d)\cdot\prod_{w\in Q}\sum_{z\in T} r[d,z]\cdot P(w \mid z)$$

Text-based language model

Topic-related information

Topic-level random walk score

- Search with query modeling
  - Proposed 3: TPRq

$$S_{\mathrm{TPRq}}(d,q) = P_{\mathrm{LM}}(q \mid d)\cdot\sum_{z\in T} r[d,z]\cdot P(z \mid q)$$

Outline

- Related work
- Our approach
  - Topic modeling
  - Topic-level random walk
  - Search with topics
- **Experimental results**
- Conclusion

Experimental Results

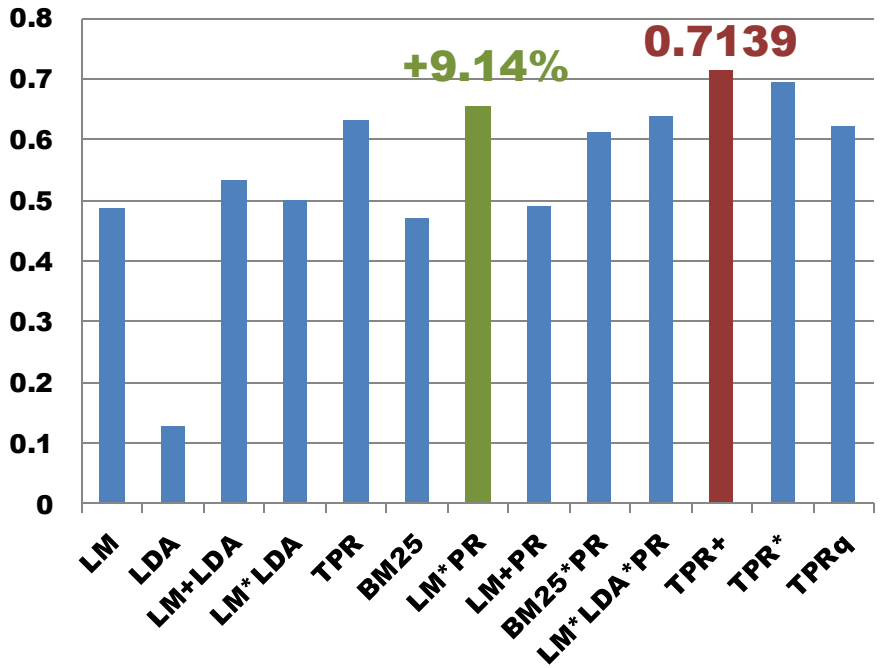
- Data sets
  - Arnetminer (http://www.arnetminer.org)
  - 14,134 authors, 10,716 papers
  - 7 most frequently searched queries
- Evaluation measures
  - P@5, P@10, P@20, R-pre, MAP

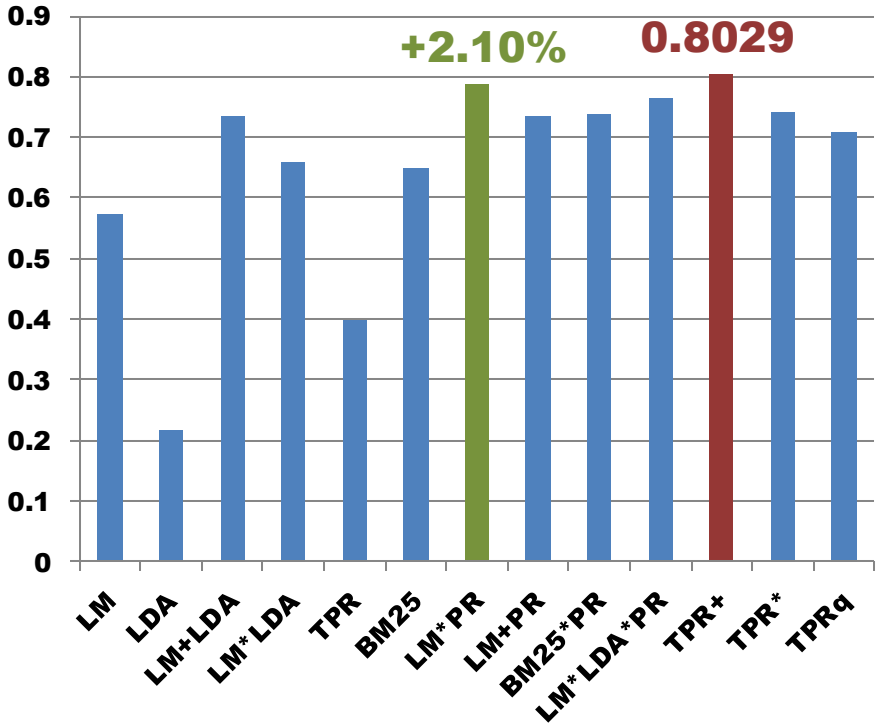
Experimental Results

- Baseline methods
  - Language model, BM25, LDA, PageRank
  - Several forms of combinations
    - LM+LDA, LM*LDA,LM*PR, LM+PR, LM*LDA*PR, BM25*PR
- Parameter settings
  - $\alpha = 0.1$, $\beta = 0.1$, $\lambda = 0.15$, $|T| = 5, 15, 80$
  - $\gamma = 0$ to 1.0 (interval 0.1)
  - $t = 0, 0.1, 0.2, 0.4, 0.6, 0.8, 1.0$

Experimental Results

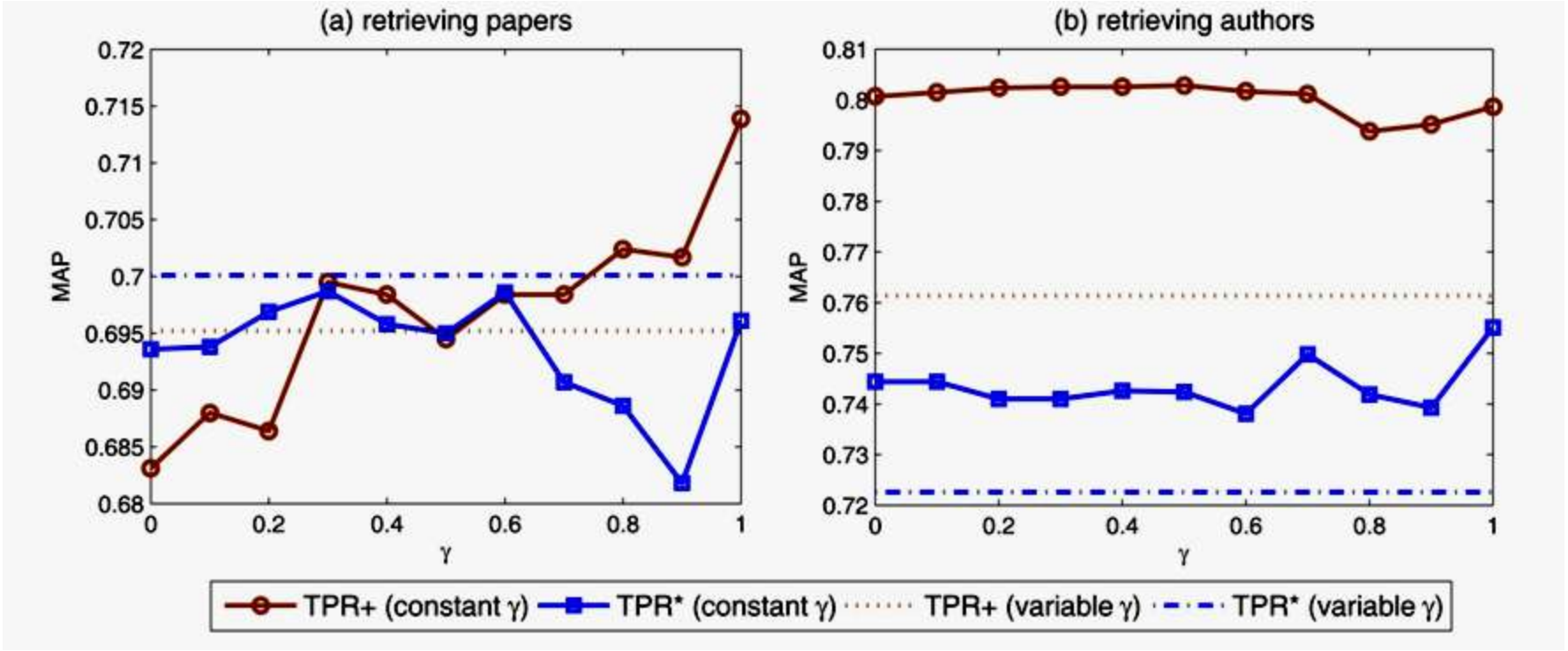
- Performance of retrieving papers/authors





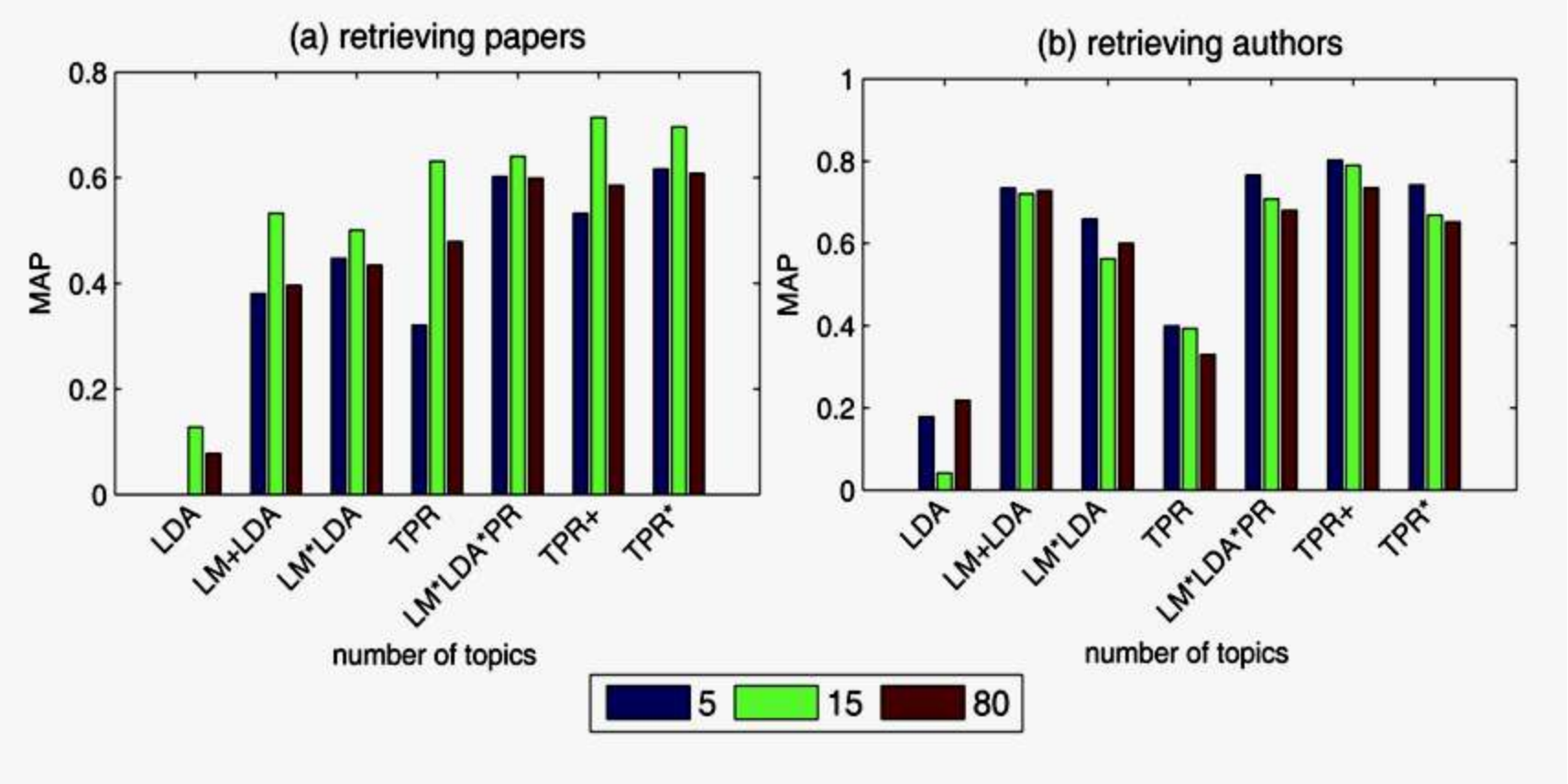
Experimental Results

- Tuning parameter γ



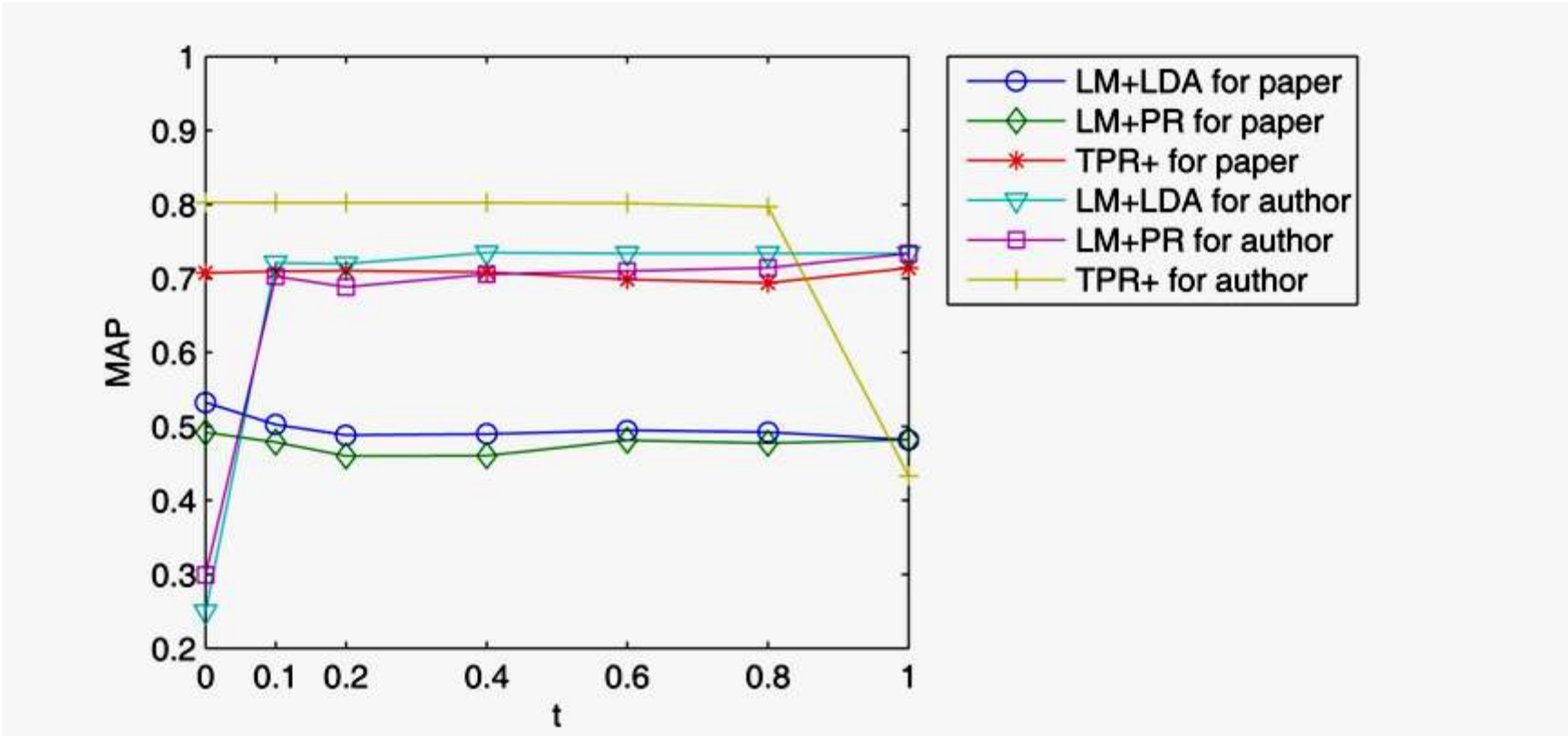
Experimental Results

- Tuning parameter $|T|$



Experimental Results

- Tuning parameter *t*



Experimental Results

- Example analysis
  - Topic differe

Intelligent agents

Natural language processing

Query Word	Topic #4	Topic #7	Topic #10	Topic #13
natural	0.000018	0.000018	**0.018966**	0.000022
language	0.000018	0.002946	**0.043322**	0.000022
processing	0.000018	0.000018	**0.012652**	0.000022
intelligent	**0.002363**	**0.022158**	0.000023	0.000022
agents	**0.037541**	**0.034784**	0.000023	0.000022

Experimental Results

– Importance scores of documents by TPR+ and PR

Paper	TPR+				PageRank
	Topic #4	Topic #7	Topic #10	Topic #13	
A	0.000113	0.000026	0.000007	0.000005	**0.000612**
B	0.000002	0.000002	**0.000055**	0.000014	0.000306
C	0.000062	0.000052	0.000050	0.000037	**0.003042**
D	**0.000236**	**0.000179**	0.000027	0.000029	0.002279

A: ***Verifiable Semantics for Agent Communication Language***
B: ***Probabilistic Parsing Using Left Conner Language Models***
C: ***The GRAIL Concept Modeling Language for Medical Terminology***
D: ***Agent-based Business Process Management***

Outline

- Related work
- Our approach
  - Topic modeling
  - Topic-level random walk
  - Search with topics
- Experimental results
- **Conclusion**

Conclusion

- Propose a 4-step framework for search through topic-level random walk.
  - Employ a probabilistic topic model to automatically extract topics from documents and further model queries
  - Perform random walk at topic level
  - Propose combination methods

Conclusion

- Experimental results show improvements (+9.14% and 2.10%).
- Future work
  - Distributed calculation?
  - Semantic link?

Thanks

Q & A

Demo : http://www.arnetminer.org