YSSNLP 2012
Shenzhen, Aug. 17 2012

Learning to Match for Natural Language Processing and Information Retrieval

Hang Li
Huawei Technologies

* Work was done at Microsoft Research, with former colleagues and interns

Language Understanding is Difficult for Computer, If Not Impossible

- 真热！
- 枯藤老树昏鸦 小桥流水人家
- 韩寒 方舟子

Language Processing without Language Understanding

- Transformation
- Transform one string to another string
- Applications
  - Machine Translation

- Matching
- Match between two strings
- Applications
  - Search
  - Question Answering

Talk Outline

- Introduction
- Regularized Latent Semantic Indexing
- Matching in Latent Space
- String Rewriting Kernel
- Conclusion

Same Search Intent, Different Query Representations Example = “Distance between Sun and Earth”

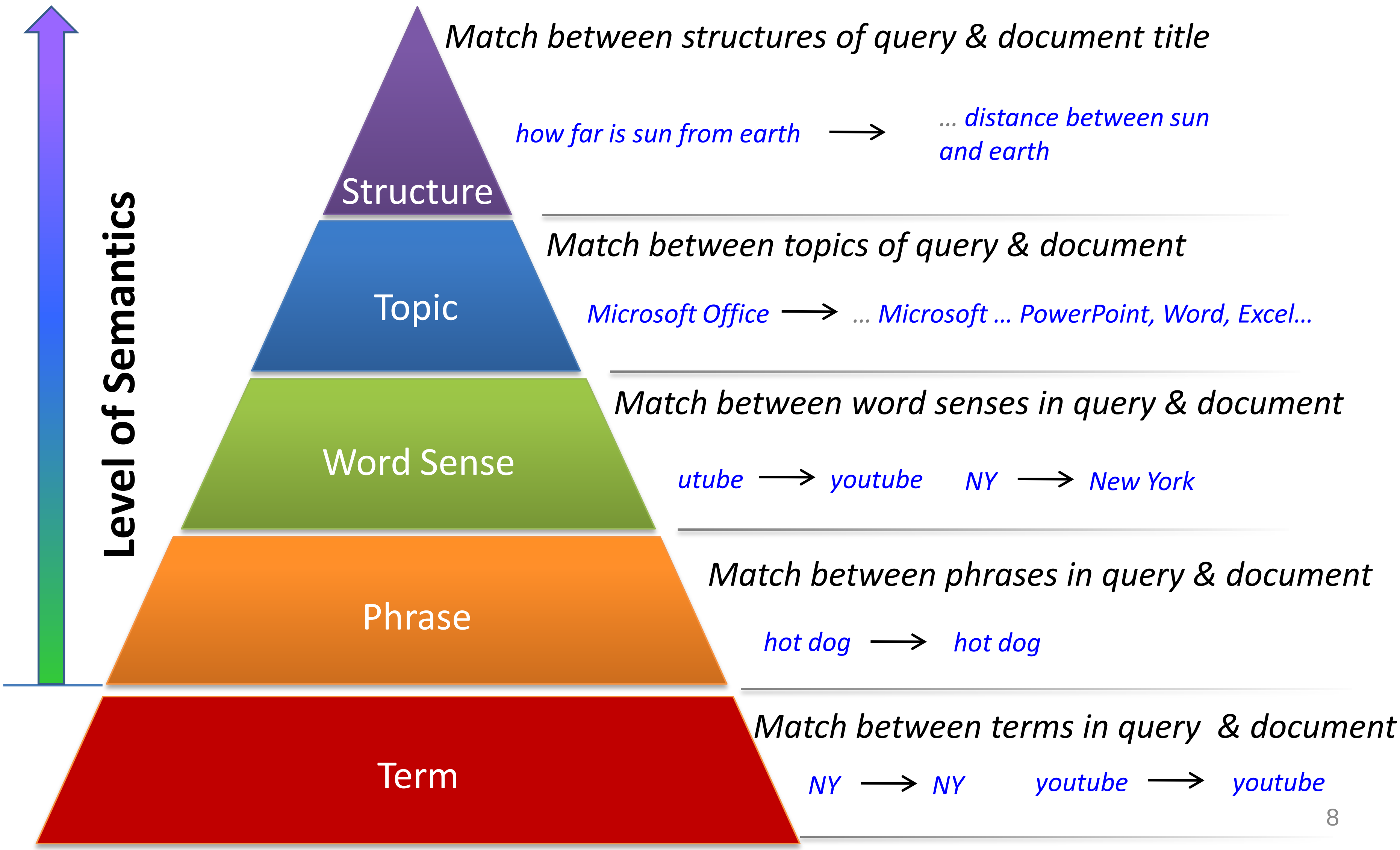
- "how far" earth sun
- "how far" sun
- "how far" sun earth
- average distance earth sun
- average distance from earth to sun
- average distance from the earth to the sun
- distance between earth & sun
- distance between earth and sun
- distance between earth and the sun
- distance from earth to the sun
- distance from sun to earth
- distance from sun to the earth
- distance from the earth to the sun
- distance from the sun to earth
- distance from the sun to the earth
- distance of earth from sun
- distance between earth sun
- how far away is the sun from earth
- how far away is the sun from the earth
- how far earth from sun
- how far earth is from the sun
- how far from earth is the sun
- how far from earth to sun
- how far from the earth to the sun
- distance between sun and earth

Same Search Intent, Different Query Representations
Example = “Youtube”

• yutube	yuotube	yuo tube
• ytube	youtubr	yu tube
• youtubo	youtuber	youtubecom
• youtube om	youtube music videos	youtube videos
• youtube	youtube com	youtube co
• youtub com	you tube music videos	yout tube
• youtub	you tube com yourtube	your tube
• you tube	you tub	you tube video clips
• you tube videos	www you tube com	wwww youtube com
• www youtube	www youtube com	www youtube co
• yotube	www you tube	www utube com
• ww youtube com	www utube	www u tube
• utube videos	utube com	utube
• u tube com	utub	u tube videos
• u tube	my tube	toutube
• outube	our tube	toutube

Semantic Matching Project: Solving Document Mismatch in Web Search

Matching at Different Levels



Query Understanding

Structure Identification

Topic Identification

Similar Query Finding

Phrase Identification

Spelling Error Correction

michael jordan berkele

michael jordan: **main phrase**
Structure

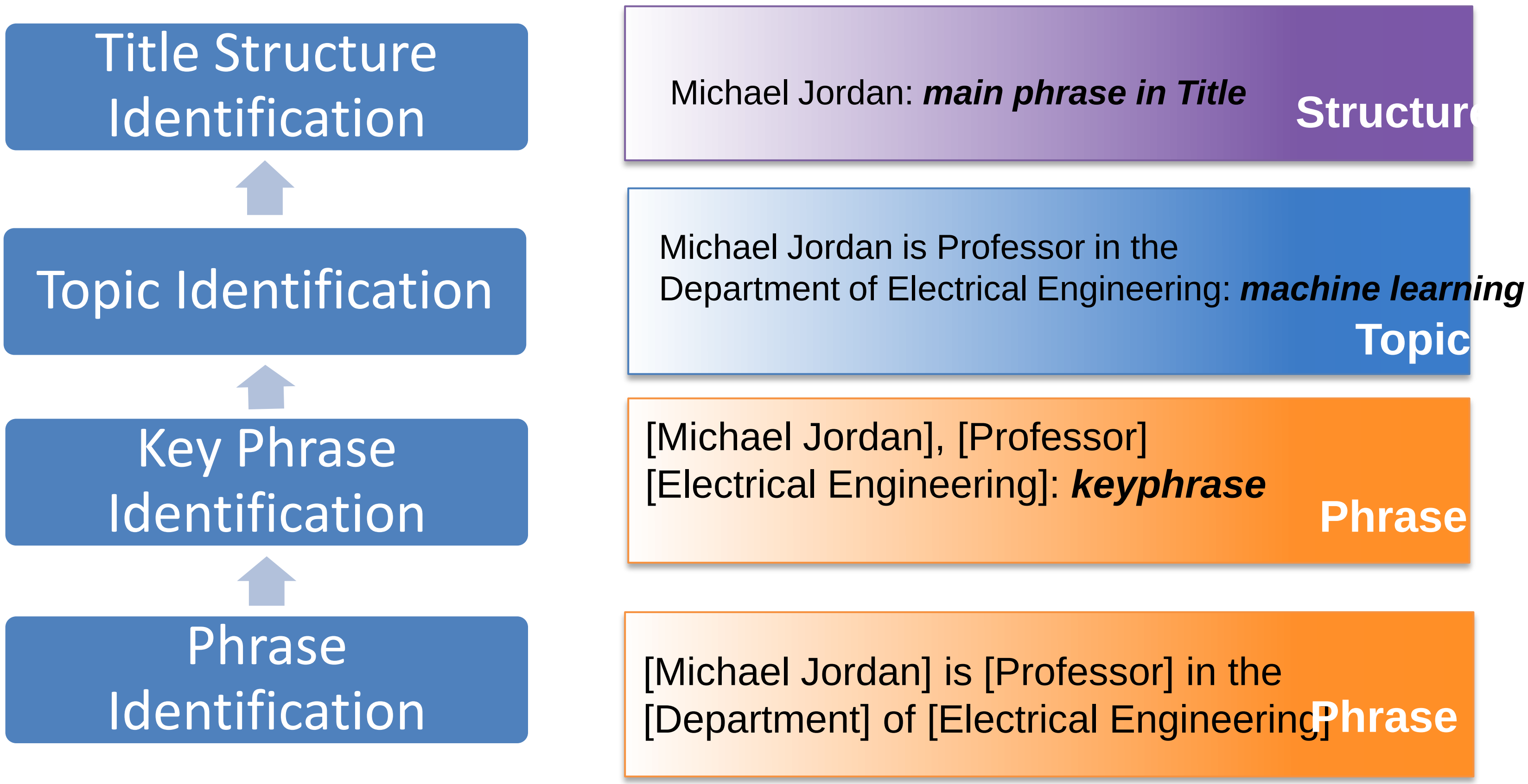
michael jordan berkely: **machine learning**
Topic

michael I. jordan
michael jordan
Sense

[michael jordan] berkeley
Phrase

michael jordan berkeley
Term

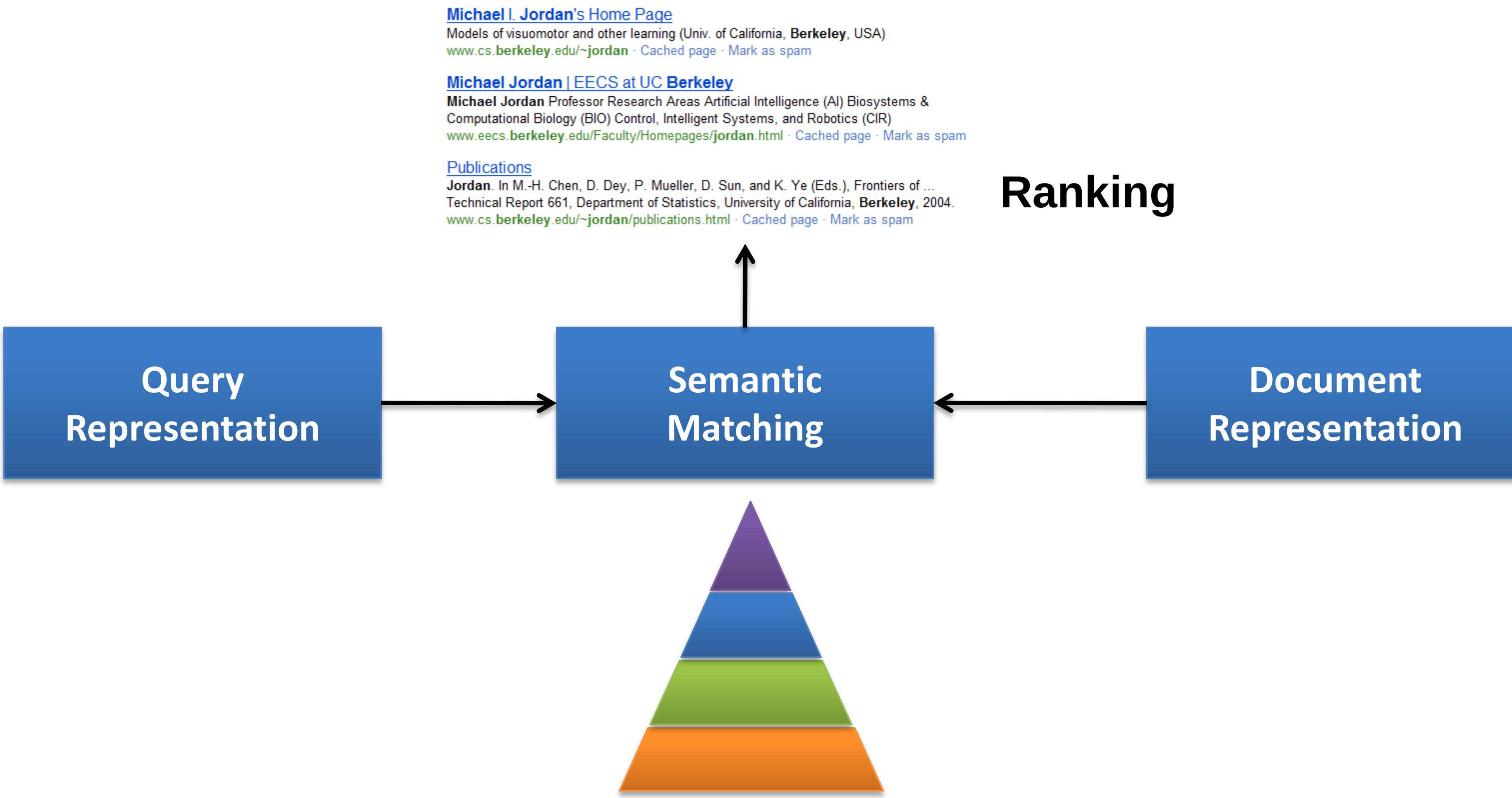
Document Understanding



Homepage of Michael Jordan

Michael Jordan is Professor in the
Department of Electrical Engineering

Online Matching



Matching is conducted at different levels

Related Work

- Studied in long history of IR
- Query expansion, pseudo relevance feedback
- Latent Semantic Indexing, Probabilistic Latent Semantic Indexing, Latent Dirichlet Allocation
- … …

- New problem setting
  - Large amount of data available
  - New machine learning techniques

Matching vs Ranking

In search, first matching and then ranking

	Matching	**Ranking**
Prediction	Matching degree between query and document	Ranking list of documents
Model	f(q, d)	f(q,d1), f(q,d2), … f(q,dn)
Challenge	Mismatch	Correct ranking on top

Matching between Heterogeneous Data is Everywhere

- Matching between user and product (collaborative filtering)
- Matching between text and image (image annotation)
- Matching between people (dating)
- Matching between languages (machine translation)
- Matching between receptor and ligand (drug design)

Regularized Latent Semantic Indexing

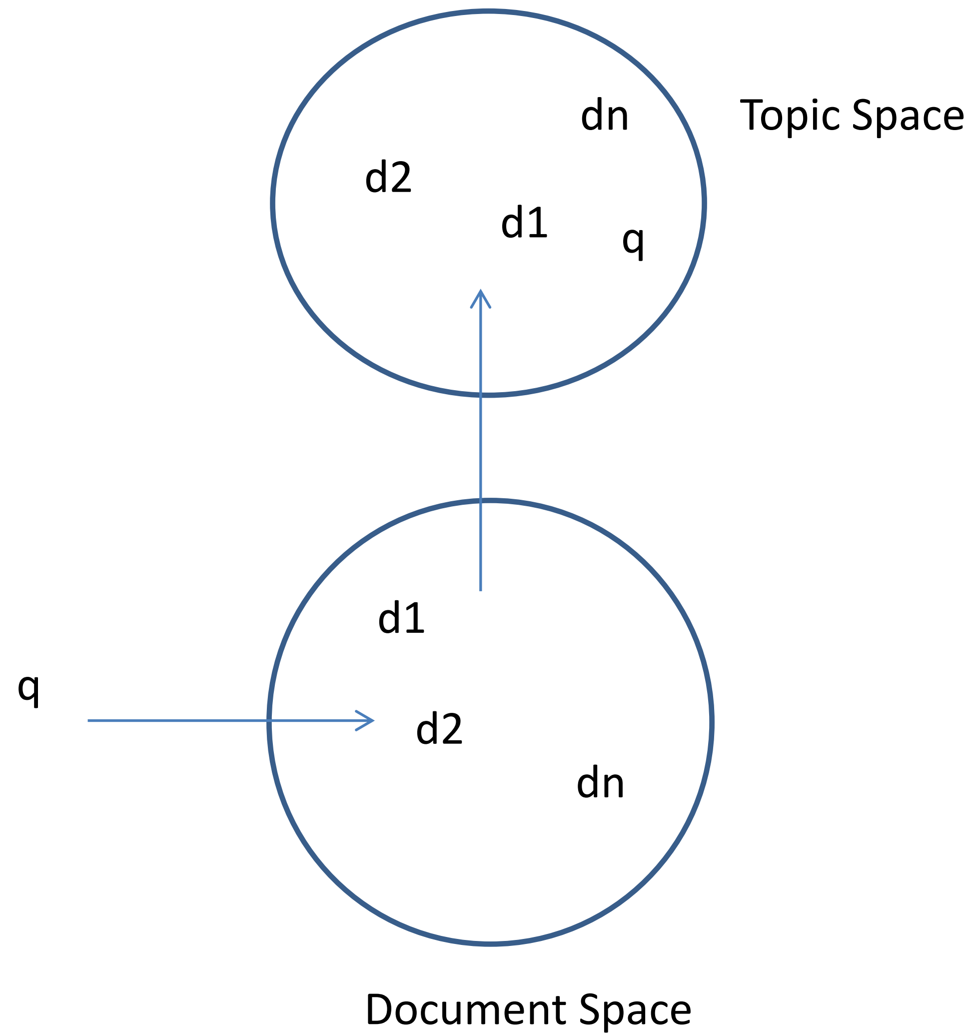
Joint Work with Quan Wang, Jun Xu, and Nick Craswell

SIGIR 2011

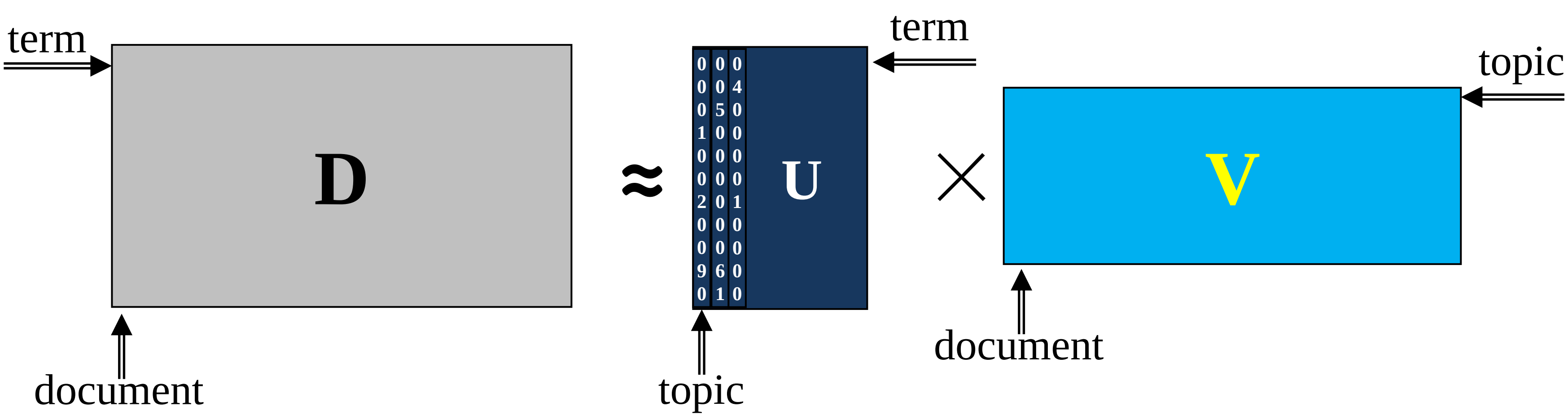
Regularized Latent Semantic Indexing

- Motivation
  - Matching between query and document at topic level
  - Scale up to large datasets (vs. existing methods)
- Approach
  - Matrix Factorization
  - Regularization on topics and documents (vs. Sparse Coding)
  - Learning problem can be easily decomposed
- Results
  - l_1 on topics leads to sparse topics and l_2 on documents leads to accurate matching
  - Comparable with existing methods in topic discovery and search relevance
  - But can easily scale up to large document sets

Query and Document Matching in Topic Space



Regularized Latent Semantic Indexing



term representation of doc n

topics

topic representation of doc n

documents are smooth

$$\min_{\mathbf{U},\{\boldsymbol{v}_n\}} \sum_{n=1}^{N} \|\boldsymbol{d}_n - \mathbf{U}\boldsymbol{v}_n\|_2^2 + \lambda_1 \sum_{k=1}^{K} \|\boldsymbol{u}_k\|_1 + \lambda_2 \sum_{n=1}^{N} \|\boldsymbol{v}_n\|_2^2$$

topics are sparse

Optimization Strategy

$$\min_{\mathbf{U},\{\boldsymbol{v}_n\}} \sum_{n=1}^{N} \|\boldsymbol{d}_n - \mathbf{U}\boldsymbol{v}_n\|_2^2 + \lambda_1 \sum_{k=1}^{K} \|\boldsymbol{u}_k\|_1 + \lambda_2 \sum_{n=1}^{N} \|\boldsymbol{v}_n\|_2^2$$

Coordinate Decent

Update $\mathbf{U}$

Update $\mathbf{V}$

$$\min_{\{\bar{\boldsymbol{u}}_m\}} \sum_{m=1}^{M} \left\|\bar{\boldsymbol{d}}_m - \mathbf{V}^T\bar{\boldsymbol{u}}_m\right\|_2^2 + \lambda_1 \sum_{m=1}^{M} \|\bar{\boldsymbol{u}}_m\|_1$$

$$\min_{\{\boldsymbol{v}_n\}} \sum_{n=1}^{N} \|\boldsymbol{d}_n - \mathbf{U}\boldsymbol{v}_n\|_2^2 + \lambda_2 \sum_{n=1}^{N} \|\boldsymbol{v}_n\|_2^2$$

$$\min_{\bar{\boldsymbol{u}}_m} \left\|\bar{\boldsymbol{d}}_m - \mathbf{V}^T\bar{\boldsymbol{u}}_m\right\|_2^2 + \lambda_1 \|\bar{\boldsymbol{u}}_m\|_1$$

for $m = 1, \cdots, M$

$$\min_{\boldsymbol{v}_n} \|\boldsymbol{d}_n - \mathbf{U}\boldsymbol{v}_n\|_2^2 + \lambda_2 \|\boldsymbol{v}_n\|_2^2$$

for $n = 1, \cdots, N$

$$u_{mk} = \begin{cases} \dfrac{(r_{mk} - \sum_{l\neq k} s_{kl}u_{ml}) - \frac{1}{2}\lambda_1}{s_{kk}}, & \text{if } u_{mk} > 0 \\[2ex] \dfrac{(r_{mk} - \sum_{l\neq k} s_{kl}u_{ml}) + \frac{1}{2}\lambda_1}{s_{kk}}, & \text{if } u_{mk} < 0 \end{cases}$$

$$\boldsymbol{v}_n^* = \left(\mathbf{U}^T\mathbf{U} + \lambda_2\mathbf{I}\right)^{-1} \mathbf{U}^T \boldsymbol{d}_n$$

Analytic Solution

RLSI Algorithm

- Single machine multi core version
- Multiple machine version (MapReduce and MPI)

Require: $\mathbf{D} \in \mathbb{R}^{M \times N}$
$\mathbf{V}^{(0)} \in \mathbb{R}^{K \times N} \leftarrow$ random matrix
for $t = 1 : T$ **do**
  $\mathbf{U}^{(t)} \leftarrow \text{Update}\mathbf{U}(\mathbf{D}, \mathbf{V}^{(t-1)})$
  $\mathbf{V}^{(t)} \leftarrow \text{Update}\mathbf{V}(\mathbf{D}, \mathbf{U}^{(t)})$
end for
return $\mathbf{U}^{(T)}, \mathbf{V}^{(T)}$

Algorithm 2 Update **U**

Require: $\mathbf{D} \in \mathbb{R}^{M \times N}$, $\mathbf{V} \in \mathbb{R}^{K \times N}$
$\mathbf{S} \leftarrow \mathbf{V}\mathbf{V}^T$
$\mathbf{R} \leftarrow \mathbf{D}\mathbf{V}^T$
for $m = 1 : M$ **do**
  $\bar{\boldsymbol{u}}_m \leftarrow \mathbf{0}$
  repeat
    for $k = 1 : K$ **do**
      $w_{mk} \leftarrow r_{mk} - \sum_{l \neq k} s_{kl} u_{ml}$
      $u_{mk} \leftarrow \frac{\left(|w_{mk}| - \frac{1}{2}\lambda N\right)_+ \text{sign}(w_{mk})}{s_{kk}}$
    end for
  until convergence
end for
return **U**

terms processed in parallel

Algorithm 3 Update **V**

Require: $\mathbf{D} \in \mathbb{R}^{M \times N}$, $\mathbf{U} \in \mathbb{R}^{M \times K}$
$\mathbf{\Sigma} \leftarrow \left(\mathbf{U}^T\mathbf{U} + \theta\mathbf{I}\right)^{-1}$
$\mathbf{\Phi} \leftarrow \mathbf{U}^T\mathbf{D}$
for $n = 1 : N$ **do**
  $\boldsymbol{v}_n \leftarrow \mathbf{\Sigma}\boldsymbol{\phi}_n$, where $\boldsymbol{\phi}_n$ is the n^{th} column
end for
return **V**

docs processed in parallel

Scalability Comparison

algorithm	max dataset applied (#docs; #words)	# topics	# processors used
PLDA and PLDA+ (by Google)	Wiki-200T(2,112,618; **200,000**)	1000	2, 048
AD-LDA	NY Times (300,000; **102,660**) PubMed (8,200,000; **141,043**)	200	16
RLSI	B01 (1,562,807; **7,014,881**) Bing News (940,702; **500,033**) Wiki-All (3,239,884; **6,043,069**) MSWeb Data (2,635,158; **2,371,146**)	500 ~ 1000	single machine, 24 cores

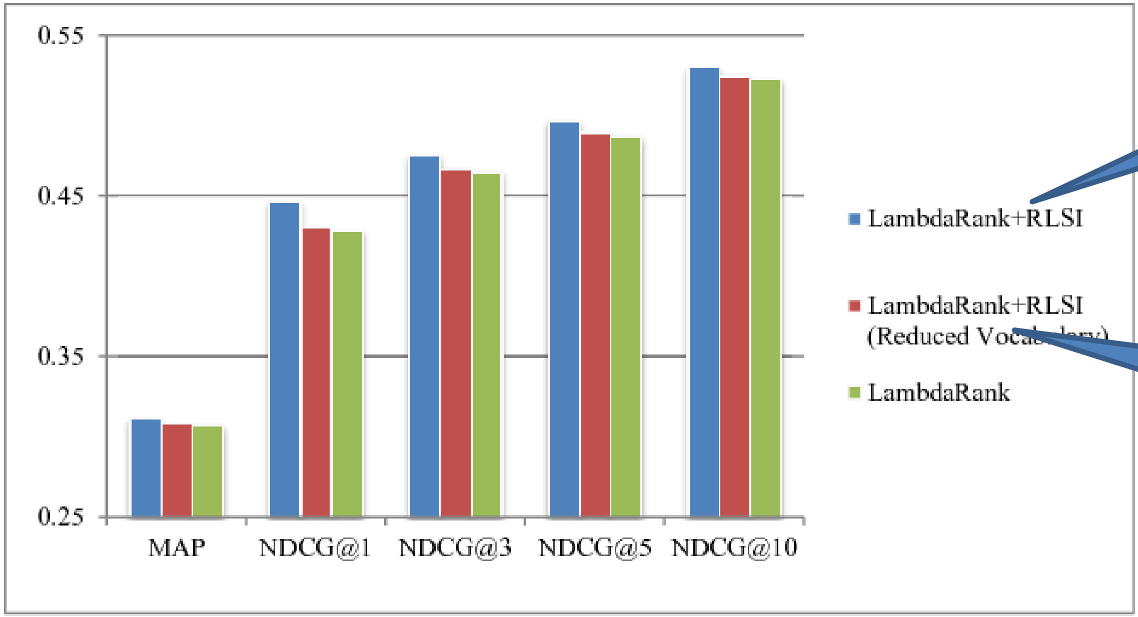
Experimental Results on Topic Discovery

Topics discovered by RLSI are equally readable compared with LDA, PLSI, LSI

Table 8: Topics discovered by RLSI, LDA, PLSI, and LSI from AP dataset.

	Topic 1	**Topic 2**	**Topic 3**	**Topic 4**	**Topic 5**	**Topic 6**	**Topic 7**	**Topic 8**	**Topic 9**	**Topic 10**
RLSI	opec	africa	aid	school	noriega	percent	plane	israeli	nuclear	bush
	oil	south	virus	student	panama	billion	crash	palestinian	soviet	dukakis
AvgComp = 0.0075	cent	african	infect	teacher	panamanian	rate	flight	israel	treaty	campaign
	barrel	angola	test	educate	delval	0	air	arab	missile	quayle
	price	apartheid	patient	college	canal	trade	airline	plo	weapon	bentsen
LDA	soviet	school	dukakis	party	year	water	price	court	air	iran
	nuclear	student	democrat	govern	new	year	year	charge	plane	iranian
AvgComp = 1	union	year	campaign	minister	time	fish	market	case	flight	ship
	state	educate	bush	elect	television	animal	trade	judge	crash	iraq
	treaty	university	jackson	nation	film	0	percent	attorney	airline	navy
PLSI	company	israeli	year	year	bush	court	soviet	year	plane	year
	million	iran	state	state	dukakis	charge	treaty	state	flight	state
AvgComp = 0.9534	share	israel	new	new	democrat	attorney	missile	new	airline	new
	billion	palestinian	nation	nation	campaign	judge	nuclear	nation	crash	people
	stock	arab	govern	0	republican	trial	gorbachev	govern	air	nation
LSI	soviet	567	0	earthquake	drug	0	israel	yen	urgent	student
	percent	234	yen	quake	school	dukakis	israeli	dukakis	oil	school
AvgComp = 1	police	0	dollar	richter	test	bush	student	bush	opec	noriega
	govern	percent	percent	scale	court	jackson	palestinian	dollar	dukakis	panama
	state	12	tokyo	damage	dukakis	dem	africa	jackson	cent	teacher

Experimental Results on Web Search



RLSI can help improve search relevance

Reducing vocabulary hurts ranking accuracy

Matching in Latent Space

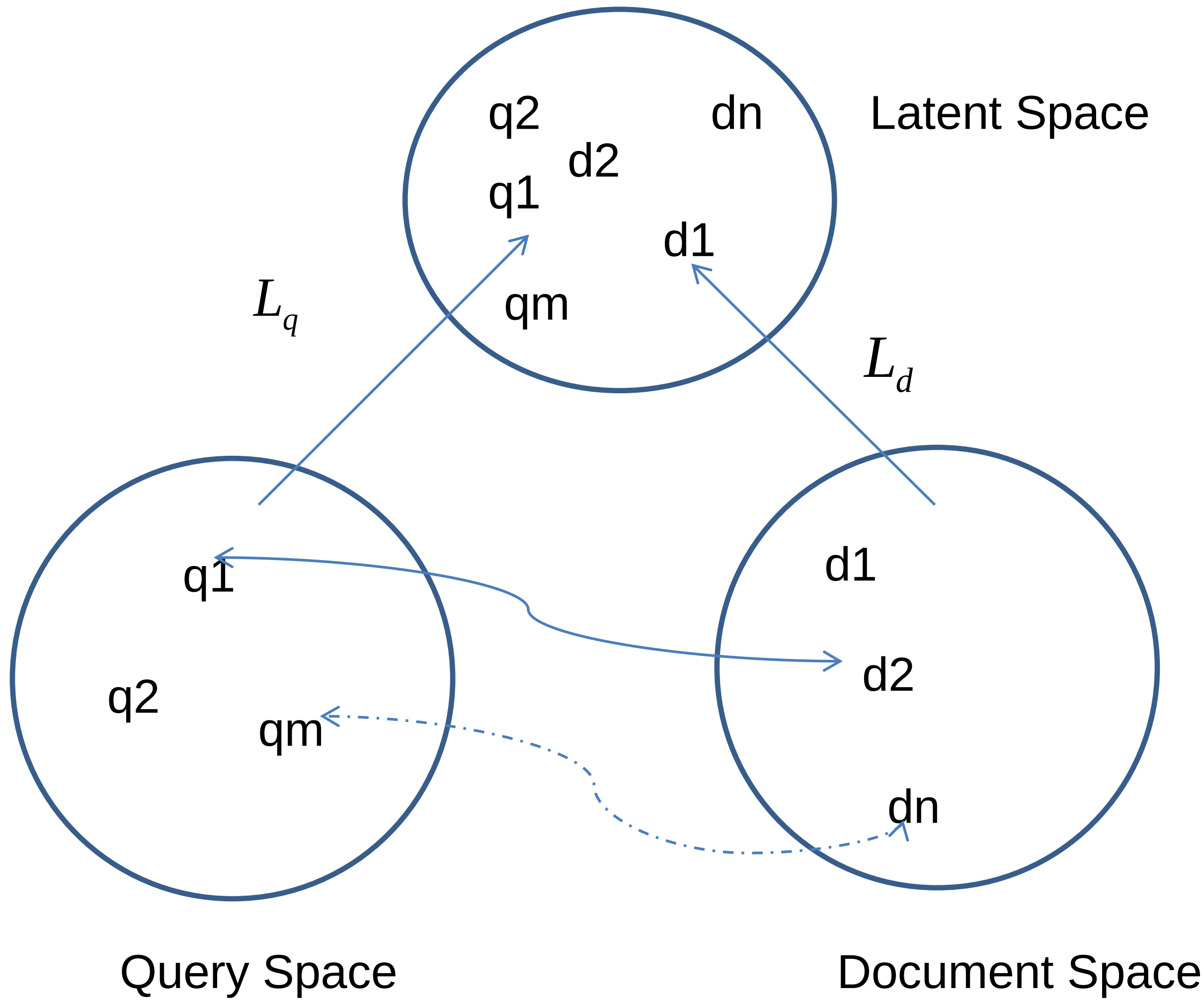
Joint Work with Wei Wu, Zhengdong Lv

Under review

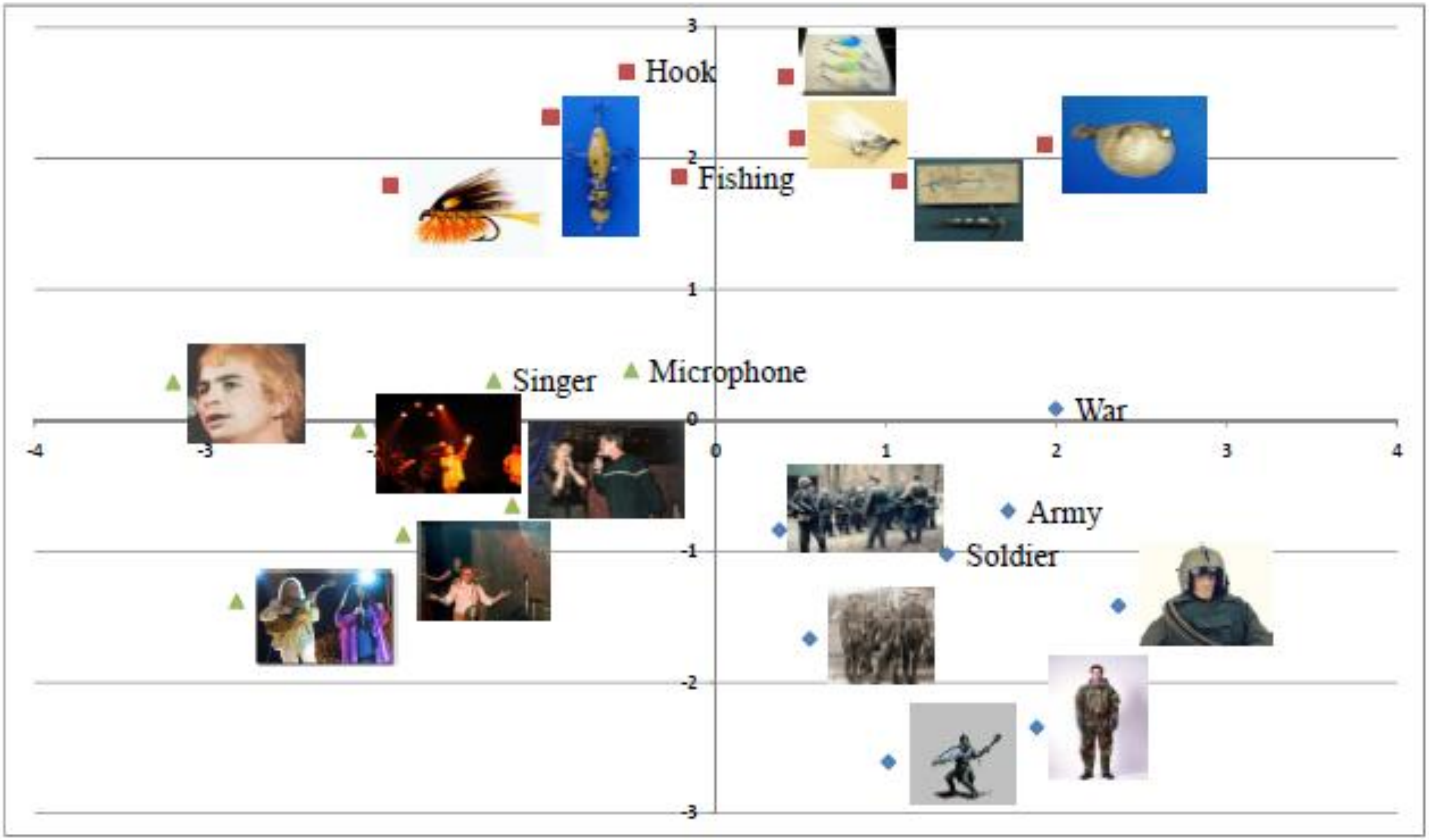
Matching in Latent Space

- Motivation
  - Matching between query and document in latent space
- Assumption
  - Queries have similarity
  - Document have similarity
  - Click-through data represent “similarity” relations between queries and documents
- Approach
  - Projection to latent space
  - Regularization or constraints
- Results
  - Significantly enhance accuracy of query document matching

Matching in Latent Space



Example: Projecting Keywords and Images into Latent Space



Partial Least Square (PLS)

- Setting
  - Two spaces: $\mathcal{X} \subset \mathbb{R}^m$ and $\mathcal{Y} \subset \mathbb{R}^n$.
- Input
  - Training data: $\{(x_i, y_i, r_i)\}_{1\le i\le N}$, $r_i \in \{+1, -1\}$ (or $r_i \in R$)
- Output
  - Similarity function $f(x, y)$
- Assumption
  - Two linear (and orthonormal) transformations $L_\mathcal{X}$ and $L_\mathcal{Y}$
  - Dot product as similarity function $\langle {L_\mathcal{X}}^T x, \ {L_\mathcal{Y}}^T y\rangle = x^T L_\mathcal{X} {L_\mathcal{Y}}^T y$
- Optimization

$$argmax_{L_\mathcal{X}, L_\mathcal{Y}} \sum_{r_i=+1} {x_i}^T L_\mathcal{X} {L_\mathcal{Y}}^T y_i - \sum_{r_i=-1} {x_i}^T L_\mathcal{X} {L_\mathcal{Y}}^T y_i$$

$$subject\ to\ {L_\mathcal{X}}^T L_\mathcal{X} = I_{k\times k}, {L_\mathcal{Y}}^T L_\mathcal{Y} = I_{k\times k}$$

Solution of Partial Least Square

- Non-convex optimization
- Can prove that global optimal solution exists
- Global optimal can be found by solving SVD (Singular Value Decomposition)
- SVD of Matrix $M_s - M_D = U\Sigma V^T$

Regularized Mapping to Latent Space (RMLS)

- Setting
  - Two spaces: $\mathcal{X} \subset \mathbb{R}^m$ and $\mathcal{Y} \subset \mathbb{R}^n$.
- Input
  - Training data: $\{(x_i, y_i, r_i)\}_{1 \le i \le N}$, $r_i \in \{+1, -1\}$ (or $r_i \in R$)
- Output
  - Similarity function $f(x, y)$
- Assumption
  - L1 and L2 regularization on $L_\mathcal{X}$ and $L_\mathcal{Y}$ (sparse transfromations)
  - Dot product as similarity function $\langle L_\mathcal{X}{}^T x,\ L_\mathcal{Y}{}^T y \rangle = x^T L_\mathcal{X} L_\mathcal{Y}{}^T y$
- Optimization

$$argmax_{L_\mathcal{X}, L_\mathcal{Y}} \sum_{r_i=+1} x_i{}^T L_\mathcal{X} L_\mathcal{Y}{}^T y_i - \sum_{r_i=-1} x_i{}^T L_\mathcal{X} L_\mathcal{Y}{}^T y_i$$

$$subject\ to\ |lx| \le \vartheta x,\ |ly| \le \vartheta y,\ \| lx \| \le \lambda x,\ \| ly \| \le \lambda y,$$

Solution of Regularized Mapping to Latent Space

- Coordinate Descent
- Repeat
  - Fix Lx, updateLy
  - Fix Ly, updateLx
- Update can be parallelized by rows

Comparison

	PLS	RMLS
Assumption	Orthogonal	L1 and L2 Regularization
Optimization Method	Singular Value Decomposition	Coordinate Descent
Optimality	Global optimum	Local optimum
Efficiency	Low	High
Scalability	Low	High

Experimental Results

Enterprise Search

	NDCG@1	NDCG@3	NDCG@5
$MPLS_{Com}$	**0.715**	**0.733**	**0.747**
$MPLS_{Conca}$	0.700	0.728	0.742
$MPLS_{Word}$	0.688	0.718	0.739
$MPLS_{Bipar}$	0.659	0.684	0.705
BM25	0.653	0.657	0.663
RW	0.654	0.683	0.700
RW+BM25	0.664	0.688	0.705
LSI	0.656	0.676	0.695
LSI+BM25	0.692	0.701	0.712

Web Search

	NDCG@1	NDCG@3	NDCG@5
$MPLS_{Com}$	**0.681**	**0.731**	**0.739**
$MPLS_{Conca}$	0.676	0.728	0.736
$MPLS_{Word}$	0.674	0.726	0.732
$MPLS_{Bipar}$	0.612	0.680	0.693
BM25	0.637	0.690	0.690
RW	0.655	0.704	0.704
RW+BM25	0.671	0.718	0.716
LSI	0.588	0.665	0.676
LSI+BM25	0.649	0.705	0.706

- RMLS and PLS work better than BM25, Random Walk, Latent Semantic Indexing
- RMLS works equally well as PLS, with higher learning efficiency and scalability

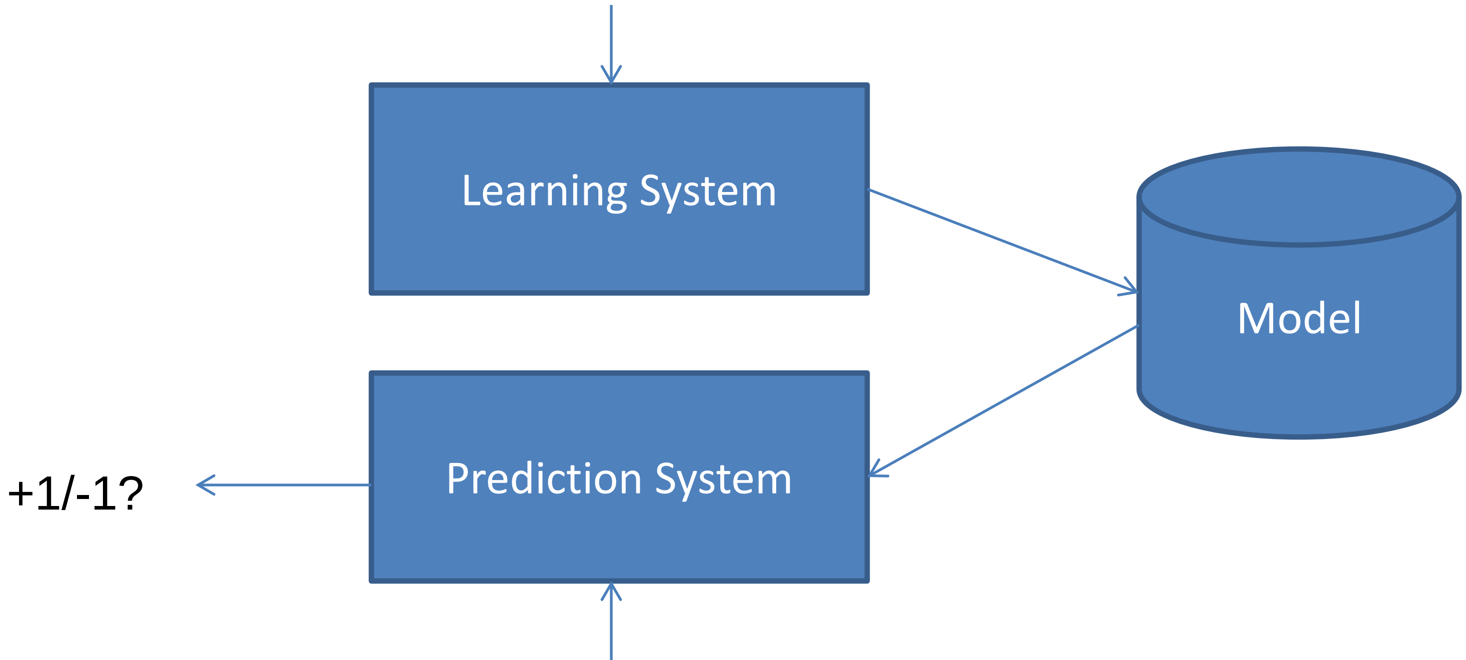
String Re-writing Kernel

Joint work with Fan Bu and Xiaoyan Zhu

ACL 2012

Learning with String Re-wring Kernel

((distance between sun and earth), (how far sun earth), +1)
((distance between beijing and shanghai), (how far is beijing from shanghai), +1)
… ….
((distance between moon and earth), (how far sun earth), -1)



((distance between moon and earth), (how far is earth from moon))

Problem Formulation

- Training data

$$((s_1, t_1), y_1) \cdots ((s_n, t_n), y_n)$$

- Model

$$y = \mathrm{sign}(\sum_{i=1}^{n} \alpha_i y_i K((s_i, t_i), (s, t)))$$

- String Re-writing Kernel

$$K((s_i, t_i), (s, t))$$

Re-Writing Rule

- Measure similarity between two pairs of strings using re-writing rule

Shakespeare wrote Hamlet

Hamlet was written by Shakespeare

Cao Xueqin wrote Dream of the Red Chamber

Dream of the Red Camber was written by Cao Xueqin

* wrote *

* was written by *

Re-writing Rule

Cao Xueqin wrote Dream of the Red Chamber

Hamlet was written by Shakespeare

Formulation of String Re-writing Kernel (SRK)

$$K((s_1,t_1),(s_2,t_2)) = \left\langle \Phi(s_1,t_1), \Phi(s_2,t_2) \right\rangle$$

$$\Phi(s,t) = (\phi_r(s,t))_{r \in R}$$

$$\phi_r(s,t) = n\lambda^i \qquad \lambda \in (0,1]$$

String Re-writing Kernel

- Advantage: Matching between informally written sentences such as long queries in search can be effectively performed
- Challenge
  - Number of re-writing rules is infinite
  - Number of matched rules increase exponentially when length of sentence increases
- Our Approach
  - Sub-class: kb-SRK

Definition of kb-SRK

- Special class of SRK
- Re-writing rules in kb-SRK
  - String patterns in rule are of length k
  - Wildcard ? only substitutes a single character
  - Alignment between string patterns is bijective

? wrote ?

? described ?

Formulation of kb-SRK

$$K_k((s_1,t_1),(s_2,t_2)) = \sum_{\substack{\alpha_{s_1} \in k-gram(s_1) \\ \alpha_{s_2} \in k-gram(s_2)}} \sum_{\substack{\alpha_{t_1} \in k-gram(t_1) \\ \alpha_{t_2} \in k-gram(t_2)}} \overline{K}_k((\alpha_{s_1},\alpha_{t_1}),(\alpha_{s_2},\alpha_{t_2}))$$

$$\overline{K}_k = \sum_{r \in R} \overline{\phi}_r(\alpha_{s_1},\alpha_{t_1})\overline{\phi}_r(\alpha_{s_2},\alpha_{t_2})$$

Experiment: Paraphrase Identification

- Comparison with state-of-the-arts methods.

lexical-based

Method	Acc.
Zhang and Patrick (2005)	71.9
Lintean and Rus (2011)	73.6
Heilman and Smith (2010)	73.2
Qiu et al. (2006)	72.0
Wan et al. (2006)	75.6
Das and Smith (2009)	73.9
Das and Smith (2009)(PoE)	76.1
Our baseline (PR)	73.6
Our method (ps-SRK)	75.6
Our method (pw-SRK)	75.0
Our method (kb-SRK)	**76.3**

Experiment: Recognizing Textual Entailment

- Comparison with state-of-the-arts methods.

Lexical-based

Method	Acc.
Harmeling (2007)	59.5
de Marneffe et al. (2006)	60.5
M&M, (2007) (NL)	59.4
M&M, (2007) (Hybrid)	64.3
Zanzotto et al. (2007)	**65.75**
Heilman and Smith (2010)	62.8
Our baseline (PR)	62.0
Our method (ps-SRK)	64.6
Our method (pw-SRK)	63.8
Our method (kb-SRK)	65.1

Conclusion

Conclusion

- Transformation and matching are two fundamental problems in natural language processing
- Query document mismatch is greatest challenge in search
- Learning to match can deal with mismatch
  - Topic Modeling
  - Latent Space
  - String Re-writing Kernel

Publications of the Project

- **Quan Wang, Zheng Cao, Jun Xu, Hang Li, Group Matrix Factorization for Scalable Topic Modeling, In Proceedings of the 35th Annual International ACM SIGIR Conference (SIGIR'12), to appear, 2012.**
- Xiaobing Xue, Yu Tao, Daxin Jiang and Hang Li, Automatically Mining Question Reformulation Patterns from Search Log Data, In Proceedings of the 50th Annual Meeting of Association for Computational Linguistics (ACL'12), to appear, 2012.
- **Fan Bu, Hang Li, Xiaoyan Zhu, String Re-Writing Kernel, In Proceedings of the 50th Annual Meeting of Association for Computational Linguistics (ACL'12), to appear, 2012.**
- Chen Wang, Keping Bi, Yunhua Hu, Hang Li, and Guihong Cao. Extracting Search-Focused Key N-Grams for Relevance Ranking in Web Search. In Proceedings of the 3rd ACM International Conference on Web Search and Data Mining (WSDM'12), 343-352, 2012.
- Wei Wu, Jun Xu, Hang Li, and Satoshi Oyama, Learning A Robust Relevance Model for Search Using Kernel Methods, Journal of Machine Learning Research, 12, 1429-1458. 2011.
- **Quan Wang, Jun Xu, Hang Li, Nick Craswell, Regularized Latent Semantic Indexing, In Proceedings of the 34th Annual International ACM SIGIR Conference (SIGIR'11), 685-694, 2011.**
- Ziqi Wang, Gu Xu, Hang Li and Ming Zhang, A Fast and Accurate Method for Approximate String Search, In Proceedings of the 49th Annual Meeting of Association for Computational Linguistics: Human Language Technologies (ACL-HLT'11), 52-61, 2011.
- Jun Xu, Hang Li, Chaoliang Zhong, Relevance Ranking Using Kernels, In Proceedings of the 6th Asian Information Retrieval Societies Symposium (AIRS'10), Best Paper Award, 1-12, 2010.
- Jiafeng Guo, Gu Xu, Hang Li, Xueqi Cheng. A Unified and Discriminative Model for Query Refinement. In Proceedings of the 31st Annual International ACM SIGIR Conference (SIGIR'08), 379-386, 2008.
- **Wei Wu, Zhengdong Lv, Hang Li, Regularized Mapping to Latent Structures and Its Application to Web Search, under review.**

Thank You!

Contact: hangli.hl@huawei.com