The 8th Chinese Workshop on
Machine Learning and Applications

Microsoft Research
微软亚洲研究院

Statistical Learning Approach to Matching of Query and Document in Search

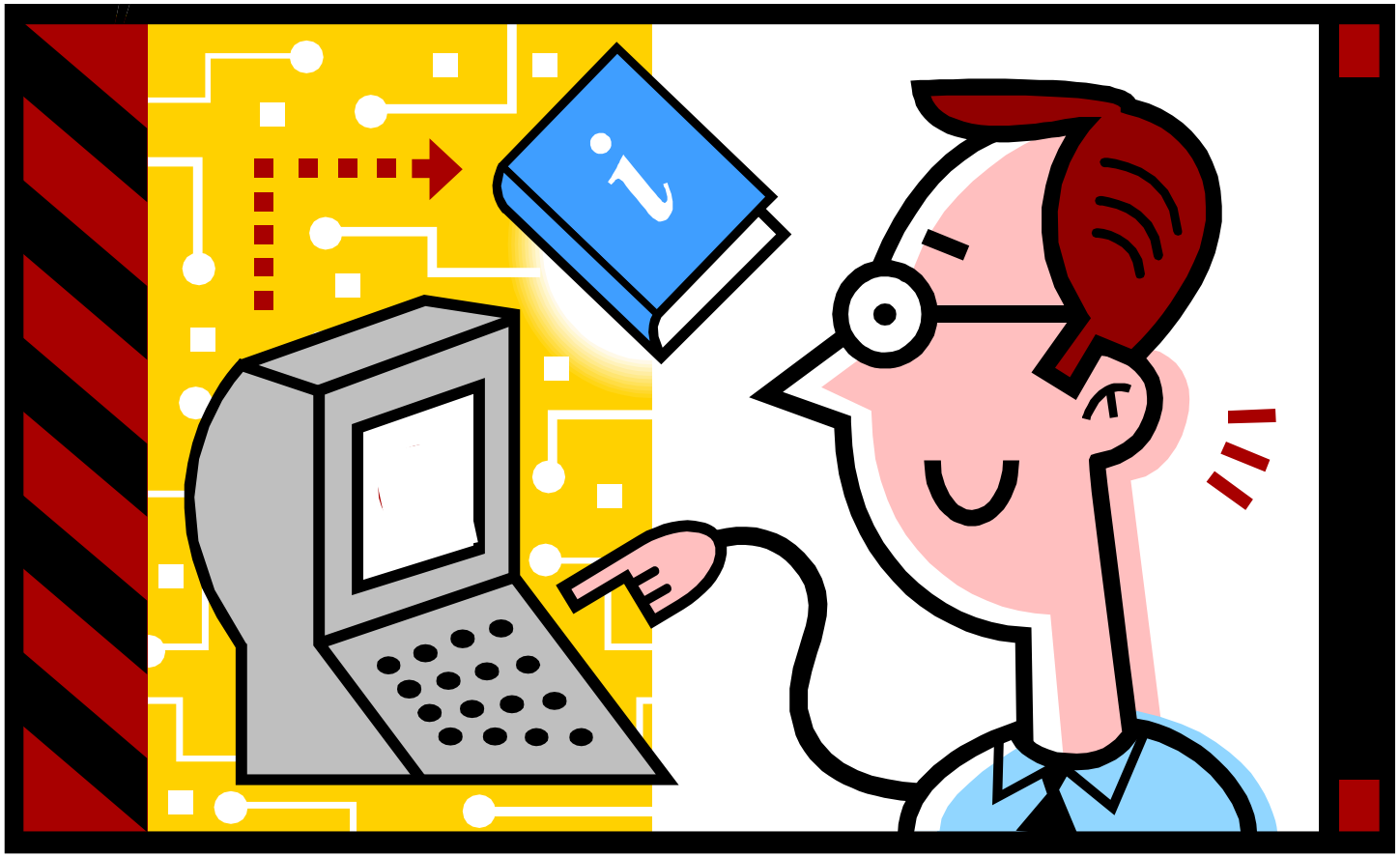
Hang Li

Microsoft Research Asia

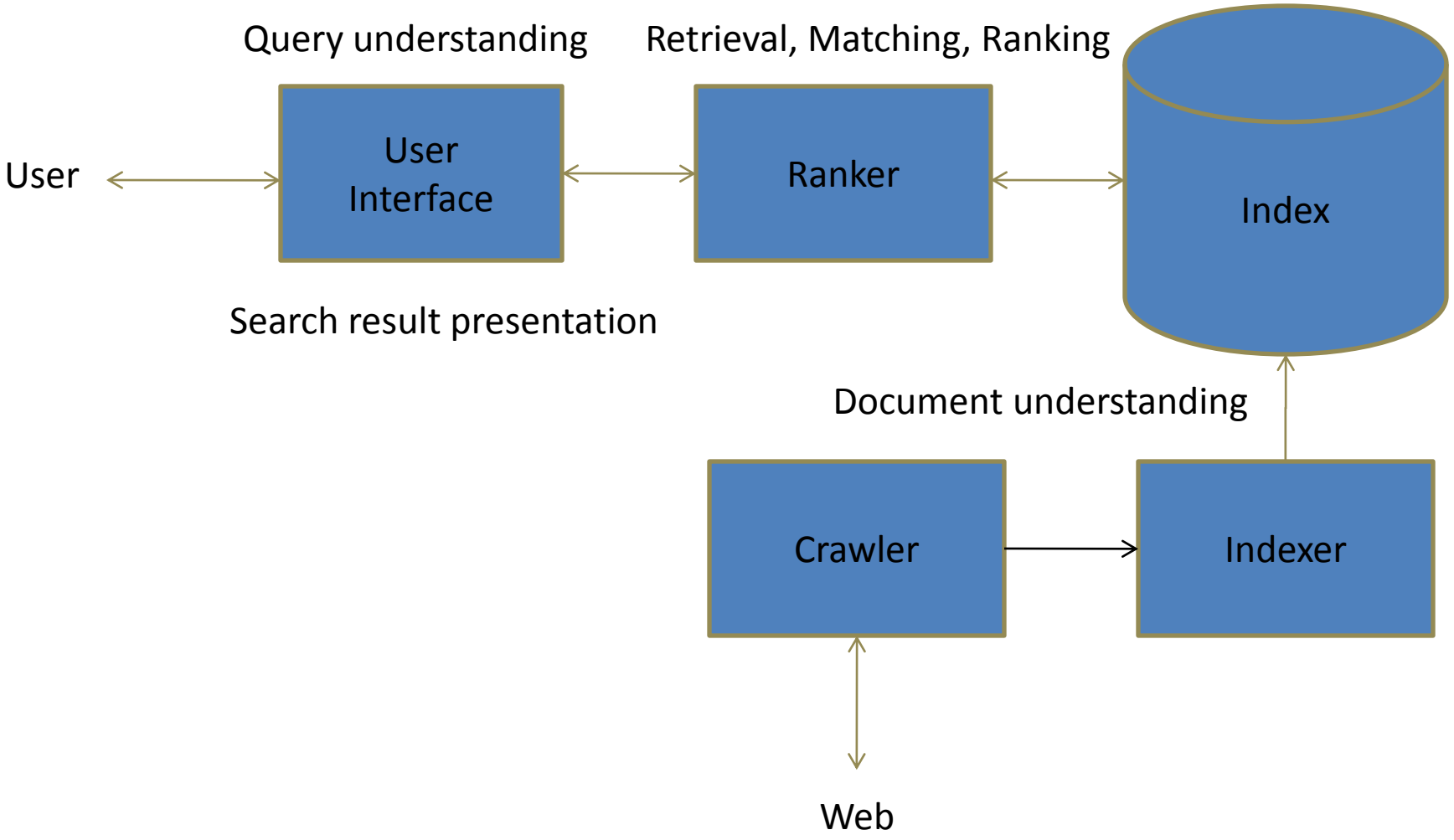
Talk Outline

- Background
  1. Key Problem in Search = Matching between Query and Document
  2. IR Models (BM25) = Matching Functions
  3. Mismatch Problem
- Problems and Our Solutions
  1. Generic IR Model as Asymmetric Kernel
  2. Extension of IR Models as Asymmetric Kernels
  3. Learning of Robust IR Models as Asymmetric Kernels
  4. Learning of Matching Function

How Search Works



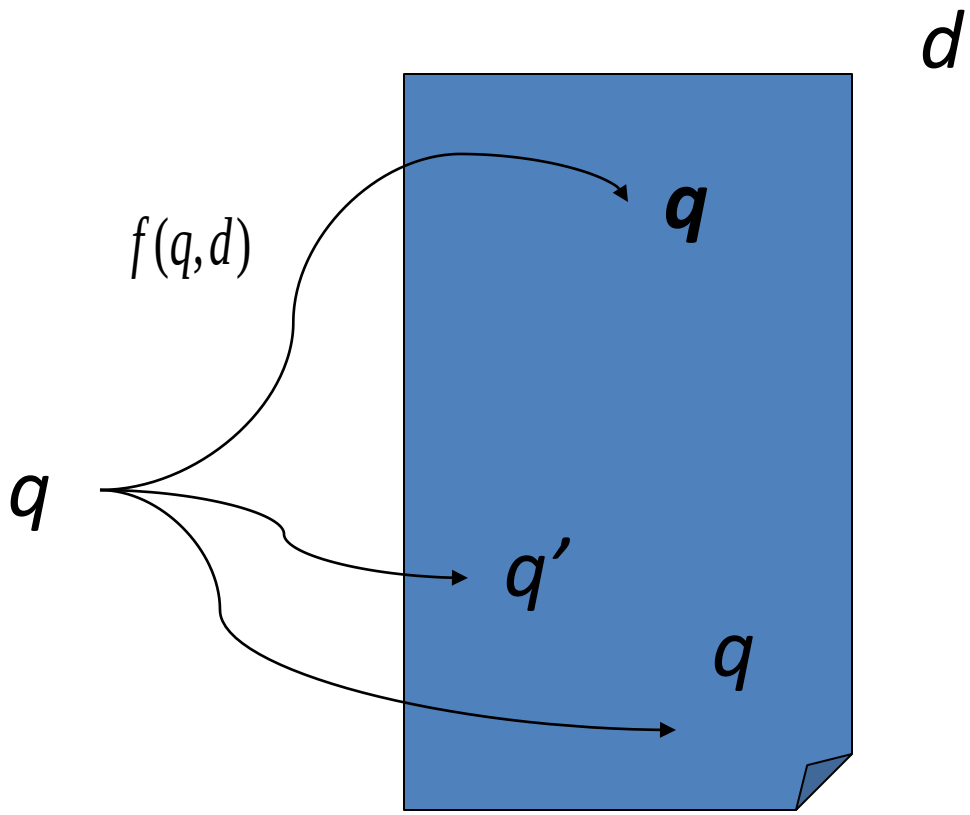
Search Architecture



Three Important Processes

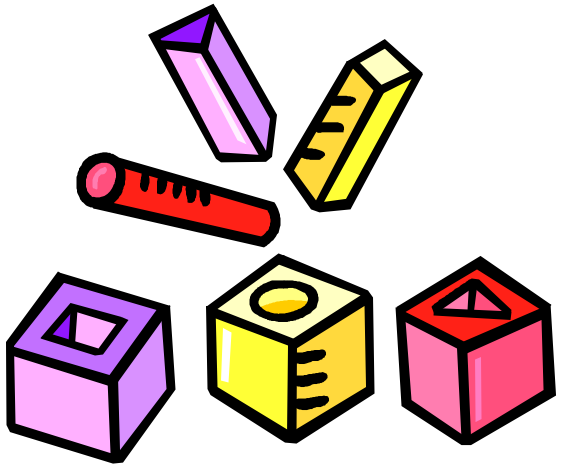
- Retrieval
  - Finding documents from inverted index
- Matching
  - Calculating relevance score between query and document pair
- Ranking
  - Ranking documents based on not only relevance scores but also importance scores, etc

Key Factor for Search: Matching between Query and Document



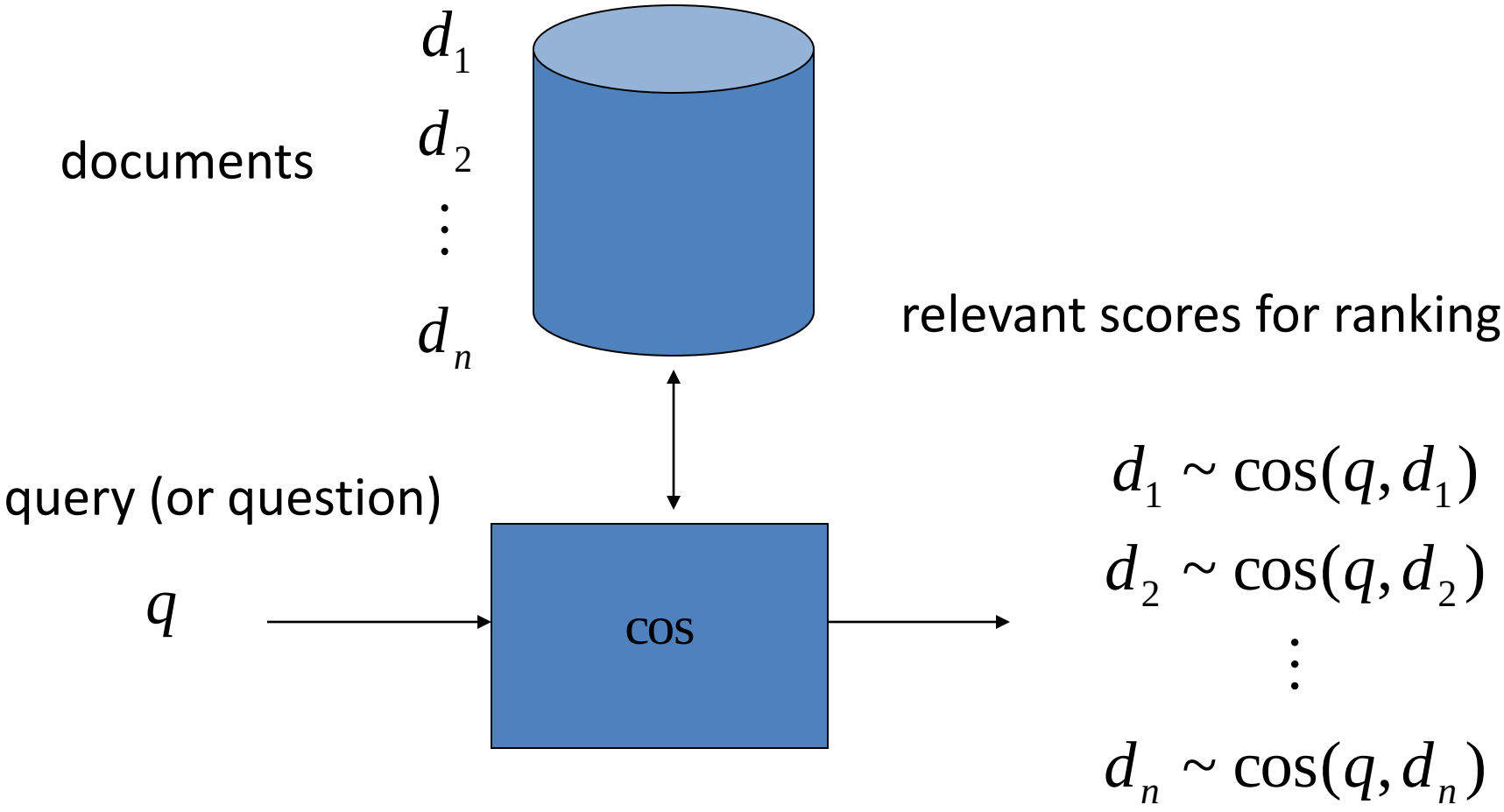
Matching between Heterogeneous Data is Everywhere

- Matching between languages (translation)
- Matching between text and image (image annotation)
- Matching between people (dating)
- Matching between user and item (collaborative filtering)

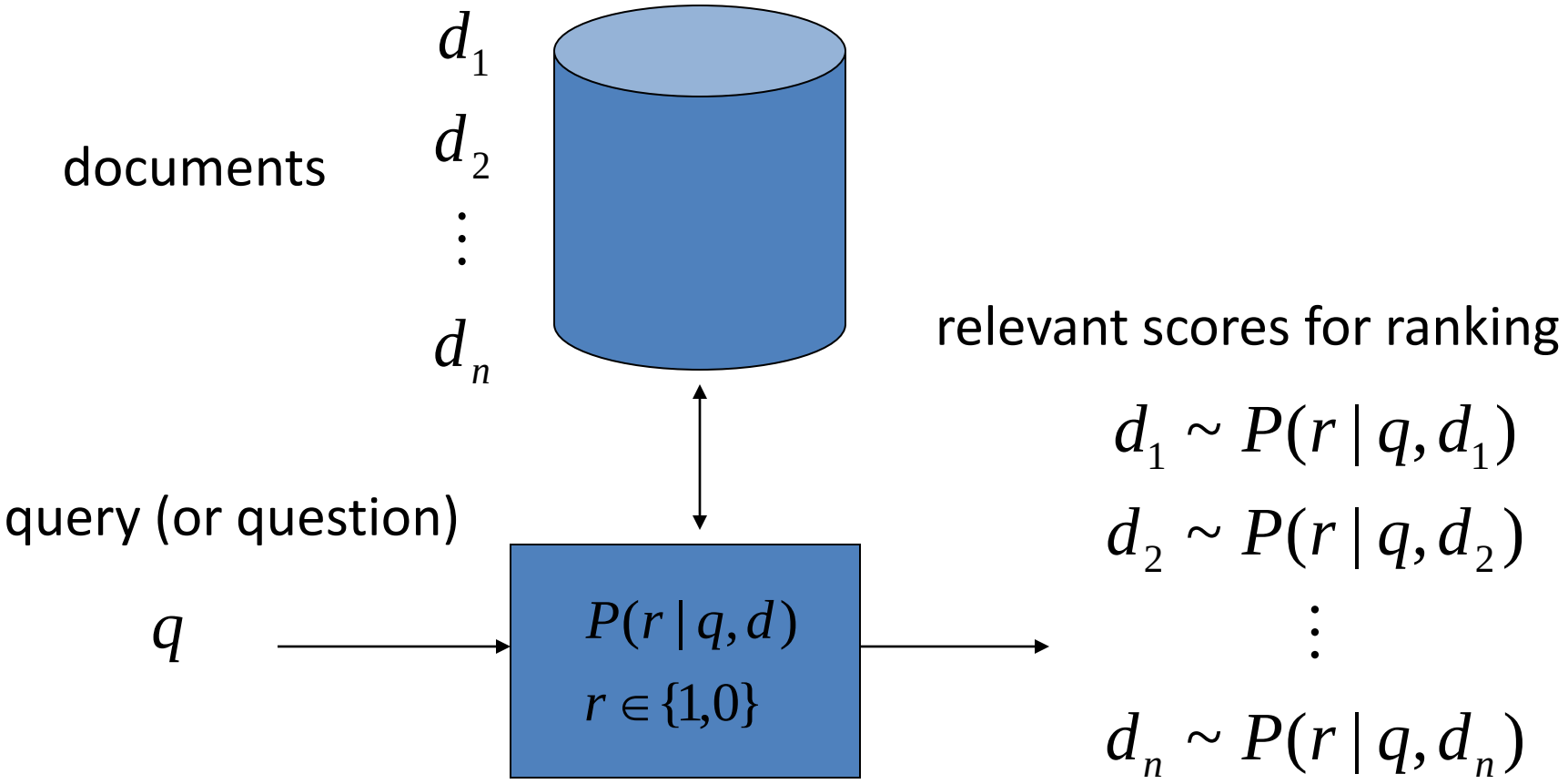


IR Models

Vector Space Model

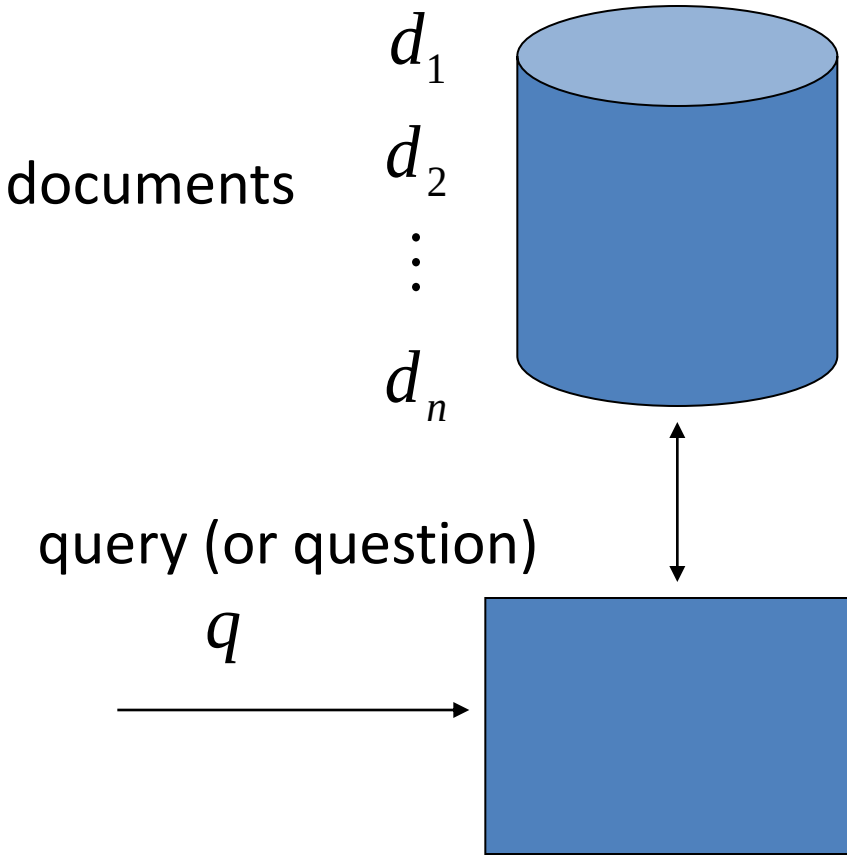


Probabilistic Model



Okapi or BM25

(Robertson and Walker 1994)



ranking function

$$\sum_{w \in d \cap q} \frac{(k+1)tf(w)}{(1-b)k + b\dfrac{dl}{avgdl} + tf(w)}$$

Language Mode

(Ponte and Croft 1998)

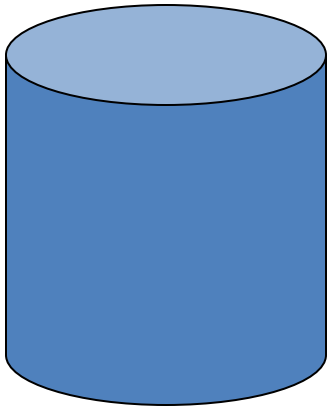
document = bag of words

$$d_1 = w_{11} w_{12} \cdots w_{1l_1}$$

$$d_2 = w_{21} w_{22} \cdots w_{2l_2}$$

$$\vdots$$

$$d_n = w_{n1} w_{n2} \cdots w_{nl_n}$$



$$q = w_{q1} w_{q2} \cdots w_{ql_q}$$

relevance scores for ranking

$$d_1 \sim P(q \mid d_1)$$

$$d_2 \sim P(q \mid d_2)$$

$$\vdots$$

$$d_n \sim P(q \mid d_n)$$

Term Mismatch = Main Challenge in Search

Examples of Term Mismatch

- Query ←→ document
- pool schedule ←→ swimming pool schedule
- seattle best hotel ← → seattle best hotels
- natural logarithm transformation ←→ logarithm tranformation
- china kong ← → china hong kong
- why are windows so expensive ←→ why are macs so expensive

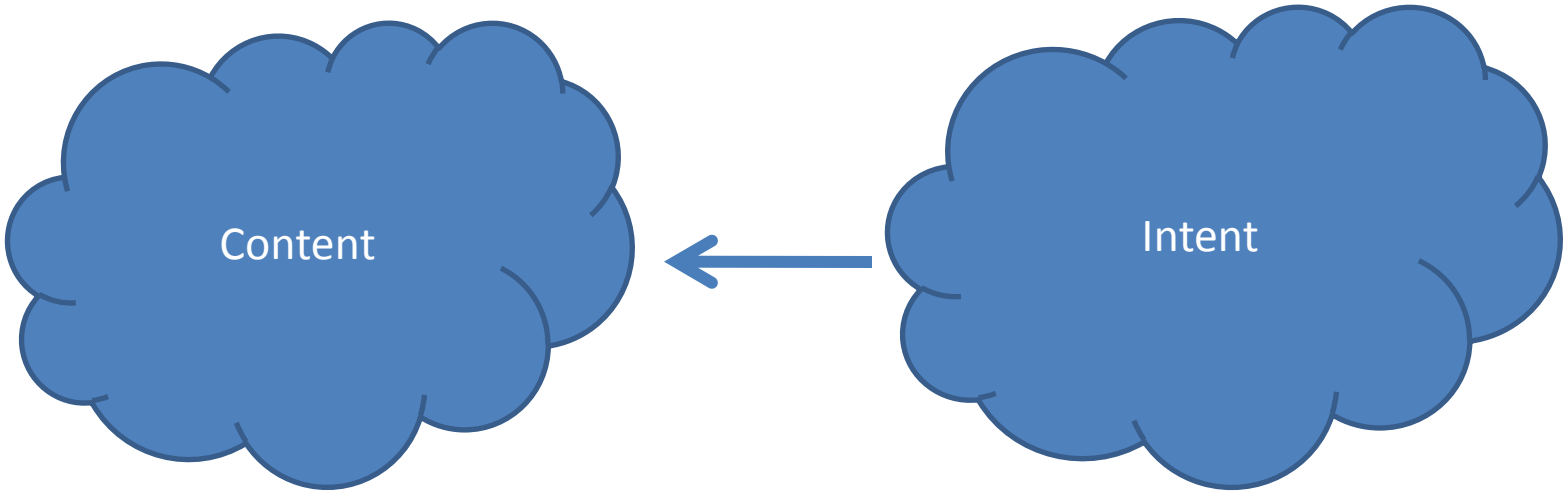
Different Queries Can Represent Same Intent "Distance between Sun and Earth"

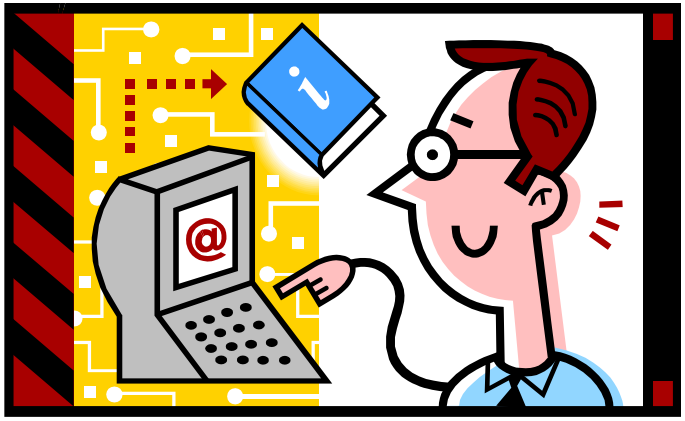
- "how far" earth sun
- "how far" sun
- "how far" sun earth
- average distance earth sun
- average distance from earth to sun
- average distance from the earth to the sun
- distance between earth & sun
- distance between earth and sun
- distance between earth and the sun
- distance between earth sun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

Different Queries Can Represent Same Intent “Youtube”

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

Matching between Two Worlds





Problems to be Addressed

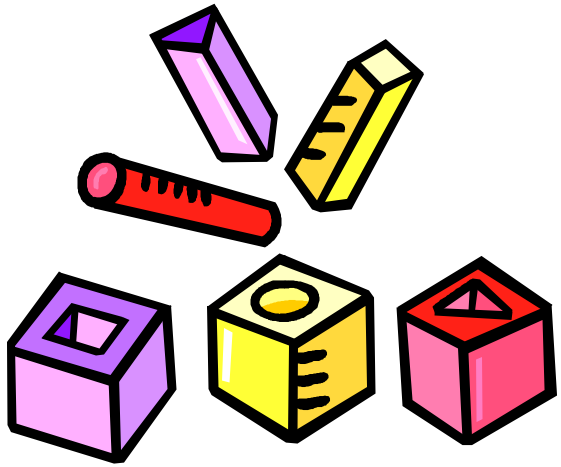
Problems to be Addressed

1. Is there unified and general framework for IR models (matching models)?
2. How to make extensions of IR models
3. How to make the IR models robust (deal with mismatch) by learning?
4. Is it possible to directly learn a matching function given training data?

Similarity Learning for Information Retrieval

Joint work with Wei Wu and Jun Xu

1. Generic IR Model as Asymmetric Kernel



Asymmetric Kernel

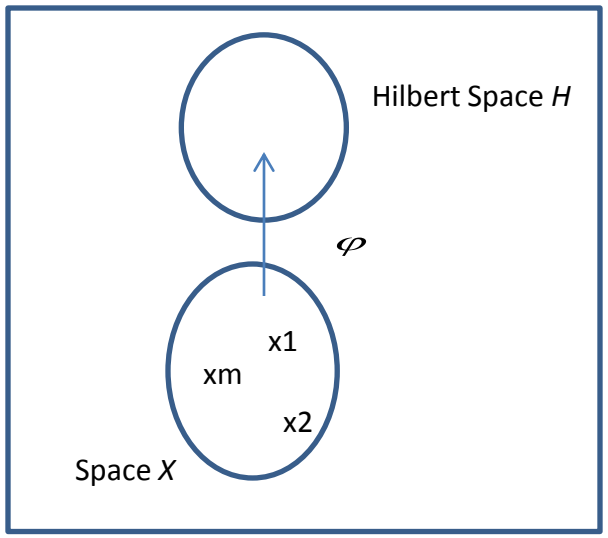
- Kernel $k: \mathcal{X} \times \mathcal{X} \rightarrow \Re$
  - Definition: $k(x, x') = \langle \phi(x), \phi(x') \rangle, where\ \phi: \mathcal{X} \rightarrow \mathcal{H}$
  - Given k_1 and k_2 are kernels, create new kernels:
    $\alpha k, where\ \alpha \geq 0\ ;\ k_1 + k_2;\ k_1 \cdot k_2$

- Asymmetric kernel: $k: \mathcal{X} \times \mathcal{Y} \rightarrow \Re$
  - Definition: $k(x, y) = \langle \phi(x), \phi'(y) \rangle, where\ \phi: \mathcal{X} \rightarrow \mathcal{H}$ and $\phi': \mathcal{Y} \rightarrow \mathcal{H}$
  - Given k_1 and k_2 are asymmetric kernels, create new asymmetric kernels:

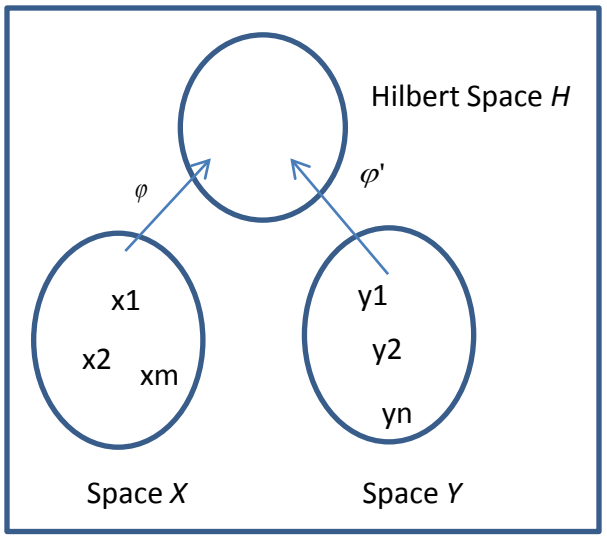
    $\alpha k, where\ \alpha \in \Re; k_1 + k_2;\ k_1 \cdot k_2$

Kernel vs Asymmetric Kernel

$$k(x, x') = \langle \phi(x), \phi(x') \rangle$$



$$k(x, y) = \langle \phi(x), \phi'(y) \rangle$$



IR Models are Asymmetric Kernels

- VSM
  - $\text{BM25}(q,d) = \left\langle \phi_Q^{VSM}(q), \phi_D^{VSM}(d) \right\rangle$, for all $w \in V$

    $\phi_Q^{VSM}(q)_w = tfidf(w,q)$ and $\phi_D^{VSM}(d)_w = tfidf(w,d)$

- BM25
  - $\text{BM25}(q,d) = \left\langle \phi_Q^{BM25}(q), \phi_D^{BM25}(d) \right\rangle$, for all $w \in V$

    $\phi_Q^{BM25}(q)_w = \frac{(k_3+1)\times tf(w,q)}{k_3+tf(w,q)}$

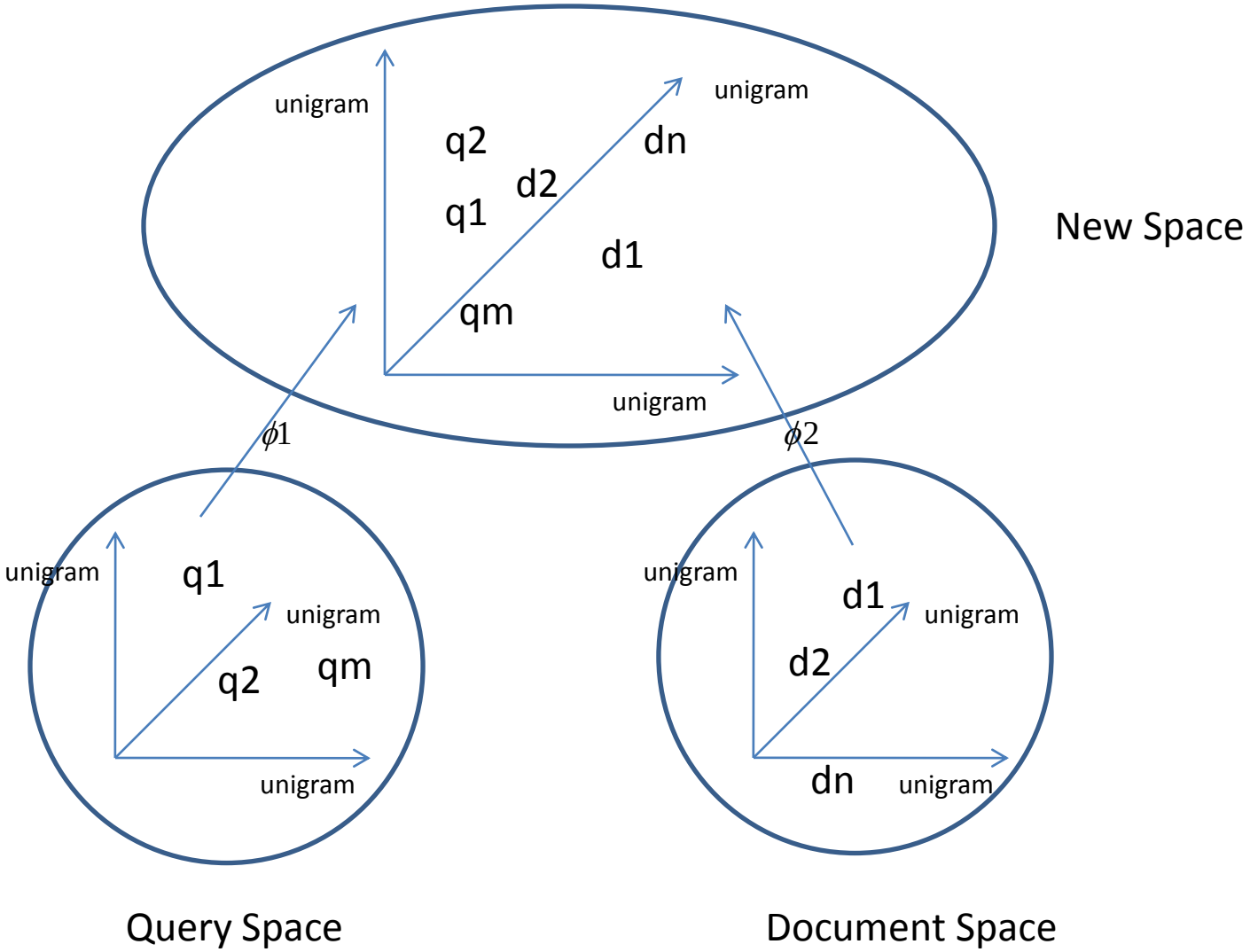
    $\phi_D^{BM25}(d)_w = \text{IDF}(w) \cdot \frac{(k_1+1)\times tf(w,d)}{k_1\left(1-b+b\cdot\frac{len(d)}{avgDocLen}\right)+tf(w,d)}$

- LMIR
  - $\text{LMIR}(q,d) = \left\langle \phi_Q^{LMIR}(q), \phi_D^{LMIR}(d) \right\rangle + len(q) \cdot \log\frac{\mu}{len(d)+\mu}$, for all $w \in V$

    $\phi_Q^{LMIR}(q)_w = tf(w,q)$

    $\phi_D^{LMIR}(d)_w = \log\left(1+\frac{tf(w,d)}{\mu \cdot P(w)}\right)$, where $P(w)$ plays similar role as IDF in BM25

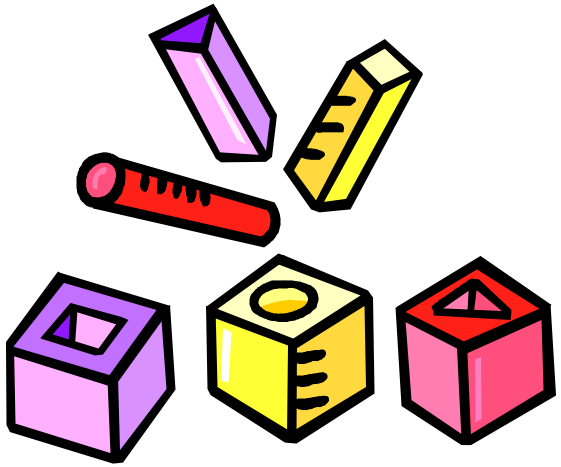
IR Models as Asymmetric Kernels



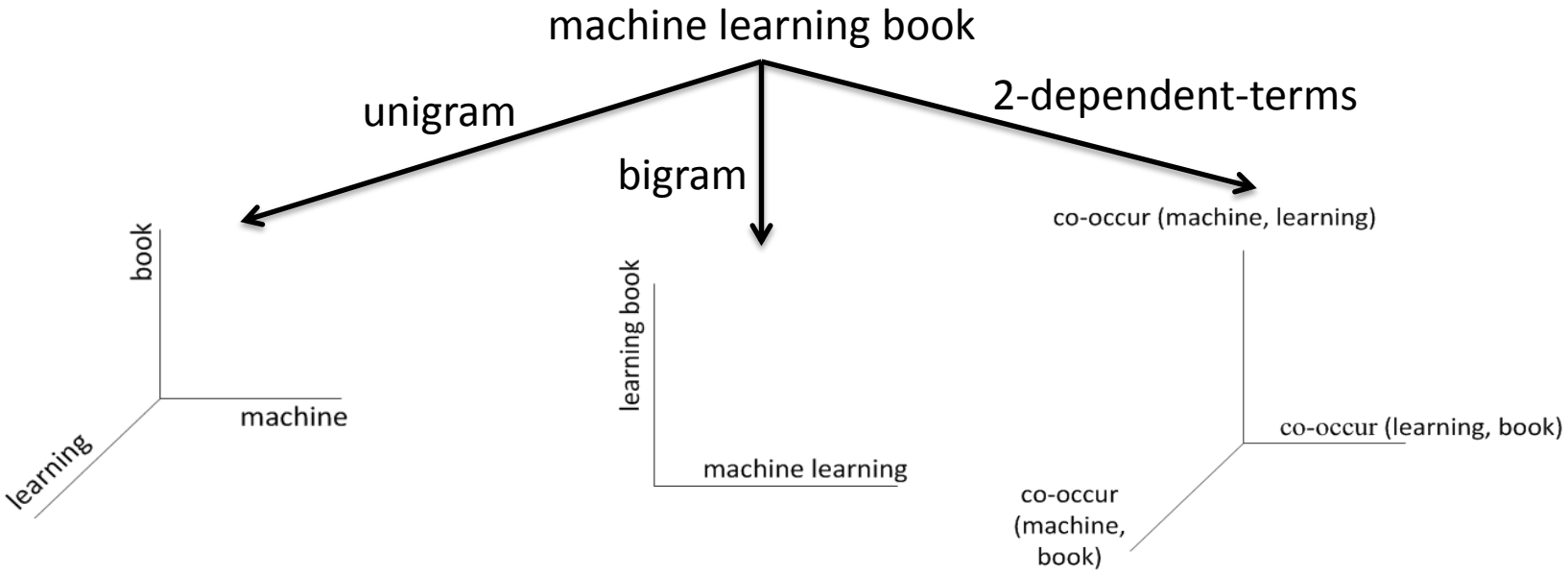
Open Questions on Asymmetric Kernels

- Sufficient and necessary condition
- Counterpart of Mercer's theorem
- What function class should work for matching in search

2. Extension of IR Models as Asymmetric Kernels



Relevance beyond Unigram



Extension of IR models

- BM25
  - $\mathrm{BM25}(q,d) = \left\langle \phi_Q^{BM25}(q), \phi_D^{BM25}(d) \right\rangle$, and for all $w \in V$

$$\phi_Q^{BM25}(q)_w = \frac{(k_3+1)\times tf(w,q)}{k_3+tf(w,q)}$$

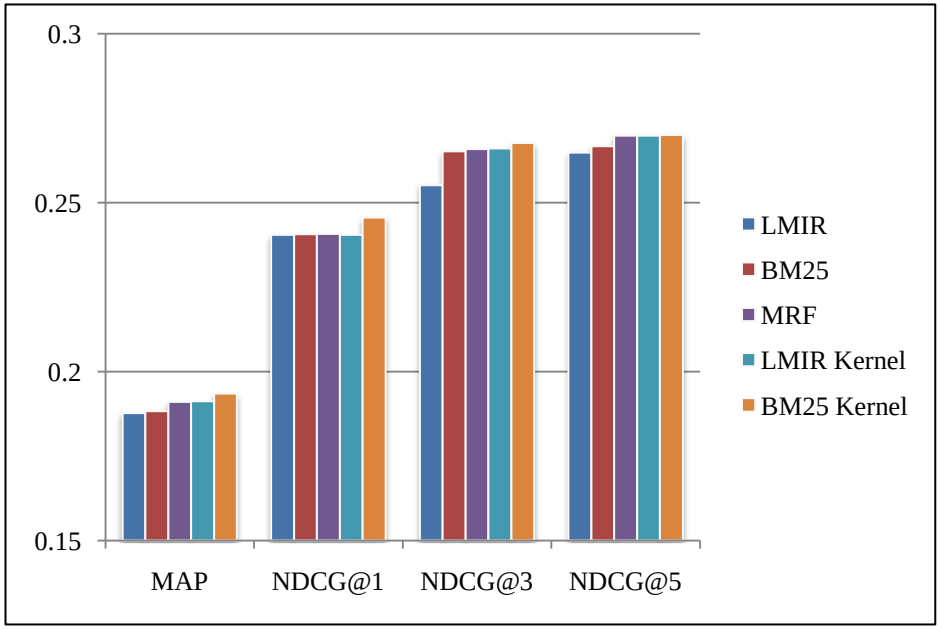
$$\phi_D^{BM25}(d)_w = \mathrm{IDF}(t) \cdot \frac{(k_1+1)\times tf(w,d)}{k_1\left(1-b+b\cdot\frac{len(d)}{avgDocLen}\right)+tf(w,d)}$$

- BM25_Kernel
  - $\mathrm{BM25}\ \_Kernel(q,d) = \sum_t BM25\ \_Kernel_t(q,d)$ where t is dependence type
  - $\mathrm{BM25}\ \_Kernel_t(q,d) = \left\langle \phi_{Q,t}^{BM25}(q), \phi_{D,t}^{BM25}(d) \right\rangle$, and for all $x \in V_t$

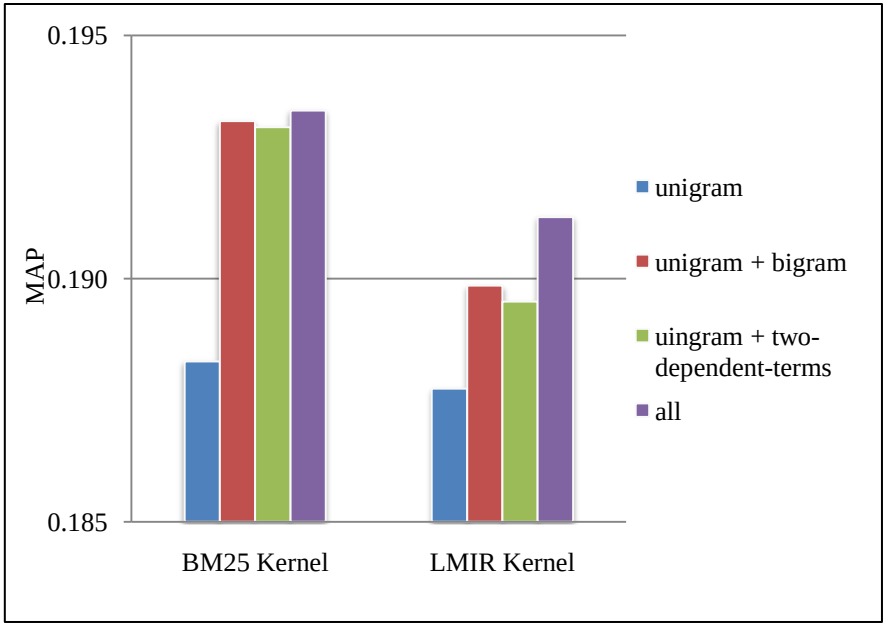
$$\phi_{Q,t}^{BM25}(q)_x = \frac{(k_3+1)\times f_t(x,q)}{k_3+f_t(x,q)}$$

$$\phi_{D,t}^{BM25}(d)_x = \mathrm{IDF}_t(x) \cdot \frac{(k_1+1)\times f_t(x,d)}{k_1\left(1-b+b\cdot\frac{f_t(d)}{avgDocLen_t}\right)+f_t(x,d)}$$

Experimental Results

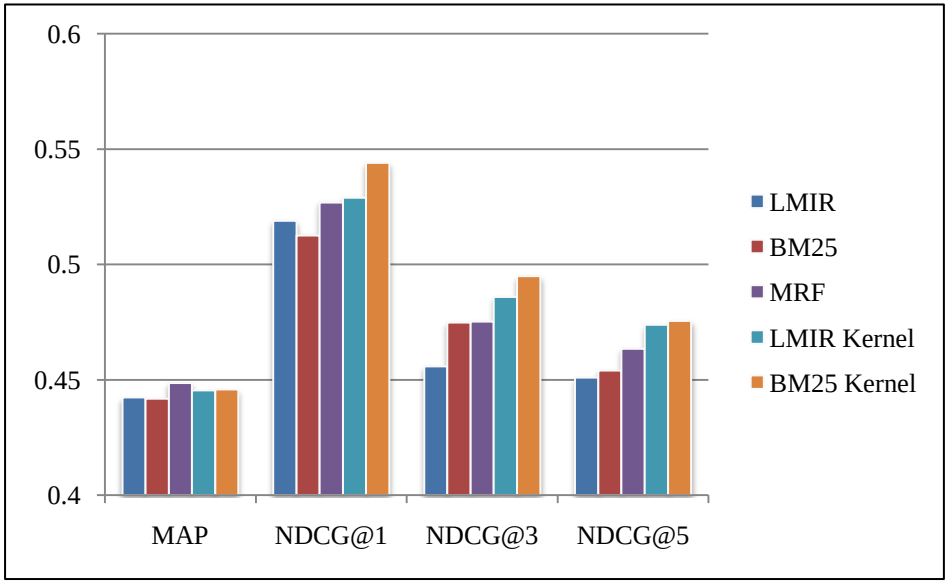


Ranking accuracies on web search data

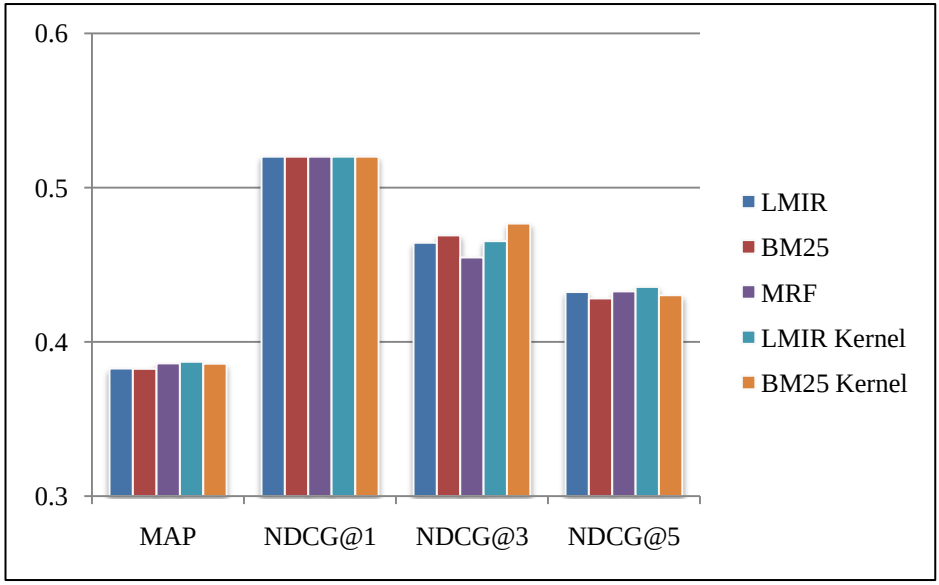


Kernel model with different term dependences

Experimental Results

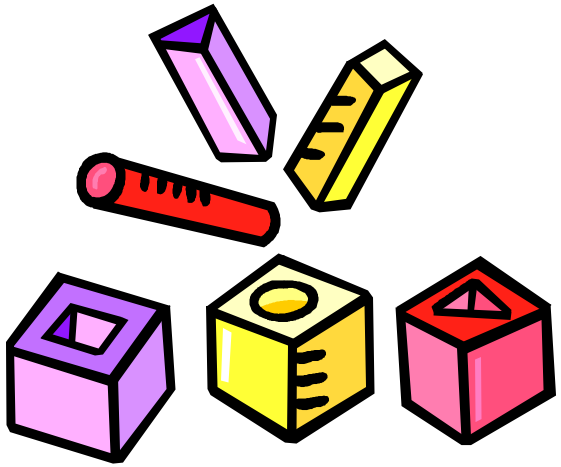


Ranking accuracies on OHSUMED



Ranking accuracies on AP

3. Learning of Robust IR Models as Asymmetric Kernels



Matching = Subset to Subset Matching

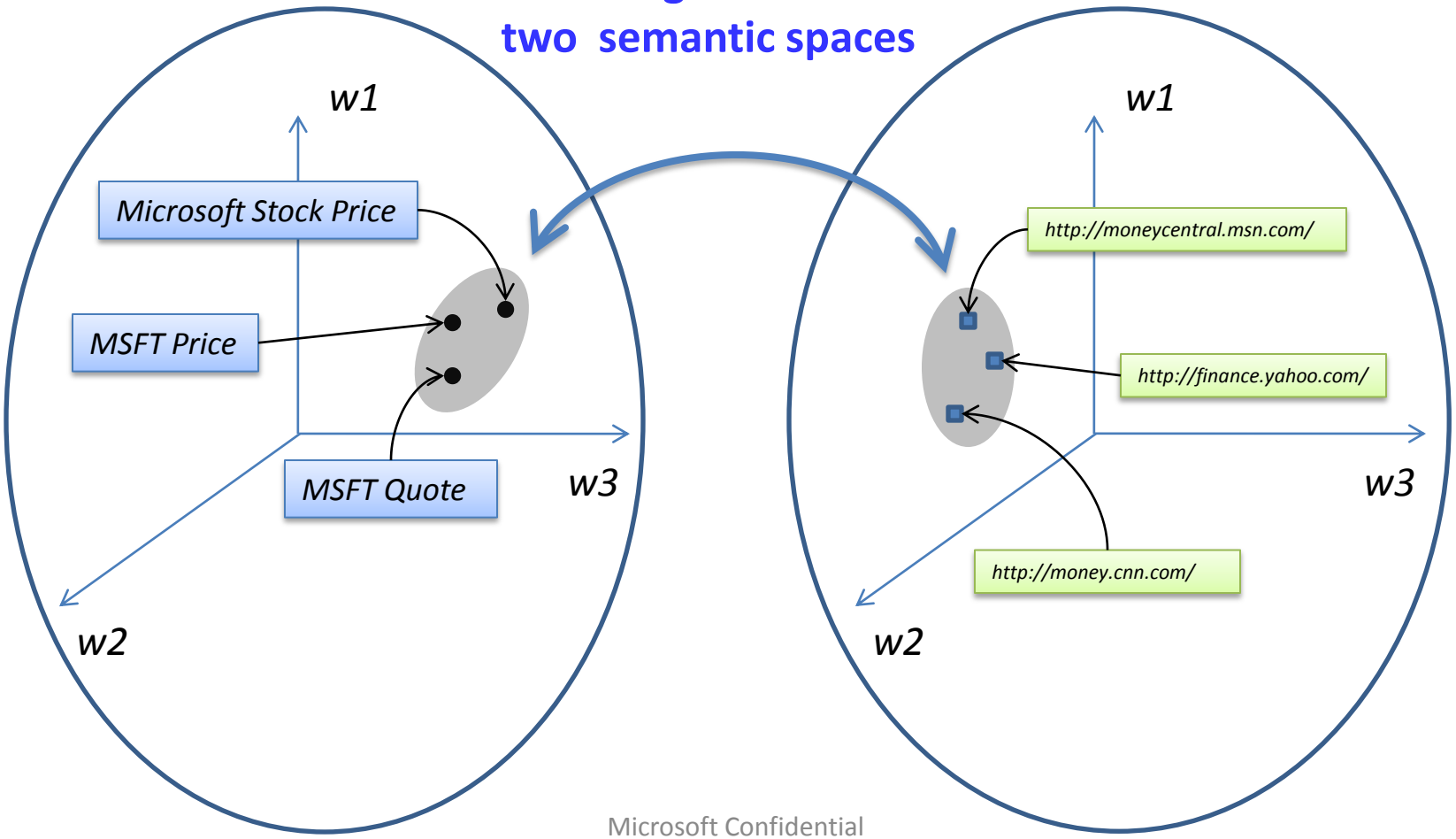
Query Space

Semantically related queries are close to each other

Document Space

Semantically related documents are close to each other

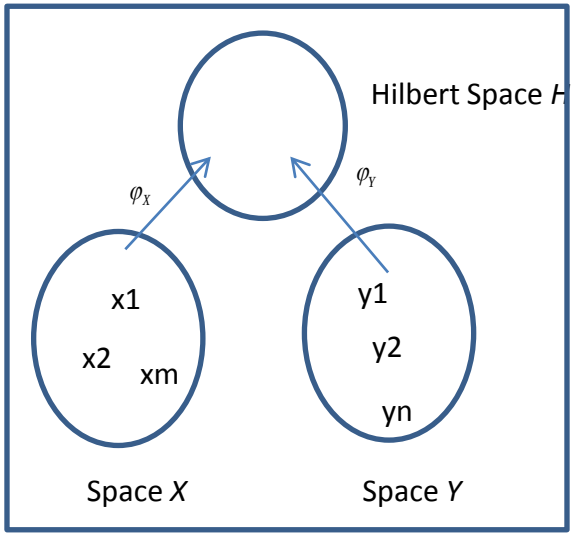
Matching between two semantic spaces



Asymmetric Kernel Learning

- Asymmetric Kernel: $k(x, y) = \langle \varphi(x), \varphi(y) \rangle_{\mathcal{H}}$
- Input
  - Training data $S = \{(x_i, y_i), t_i\}_{1 \le i \le N}$
- Output
  - Asymmetric kernel function
- Optimization

$$\min_{k \in \mathcal{K} \subseteq \mathcal{A}} \frac{1}{N} \sum_{i=1}^{N} l(k(x_i, y_i), t_i) + \Omega(k)$$



Asymmetric Kernel Learning Using Kernel Methods

- Assumption
  - Space of asymmetric kernels is RKHS generated by positive-definite kernel $\bar{k}$: $(\mathcal{X} \times \mathcal{Y}) \times (\mathcal{X} \times \mathcal{Y})$
  - Hyper Asymmetric Kernel (HAK)
- Optimization

$$\min_{k \epsilon \mathcal{K}} \frac{1}{N} \sum_{i=1}^{N} l(k(x_i, y_i), t_i) + \frac{\lambda}{2} \|k\|_{\mathcal{K}}{}^2$$

- Solution
  - By representer theorem $k^*(x, y) = \sum_{i=1}^{N} \alpha_i \bar{k}((x_i, y_i), (x, y))$
- Hyper Asymmetric Kernel

$$\bar{k}\big((x, y), (x', y')\big) = g(x, y) k_{\mathcal{X}}(x, x') k_{\mathcal{Y}}(y, y') g(x', y')$$

Learning Robust BM25

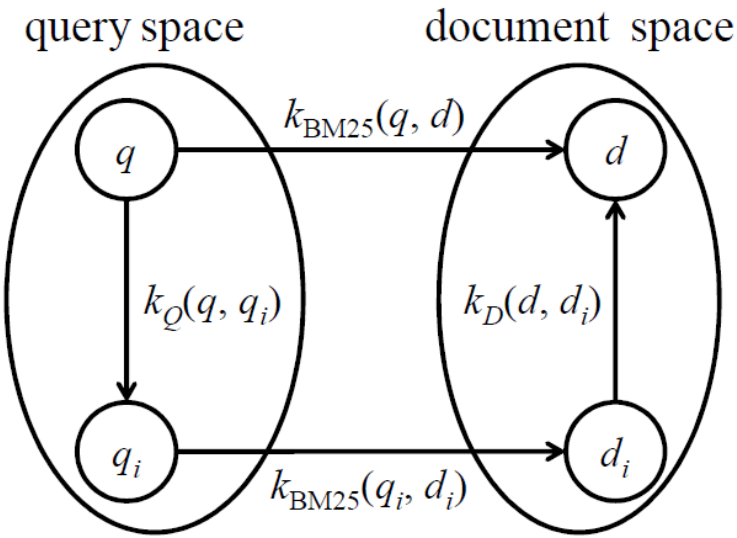
- BM25 = asymmetric kernel $k_{BM25}(q,d)$
- HAK

$$\bar{k}\big((q,d),(q',d')\big) = k_{BM25}(q,d)k_Q(q,q')k_D(d,d')k_{BM25}(q',d')$$

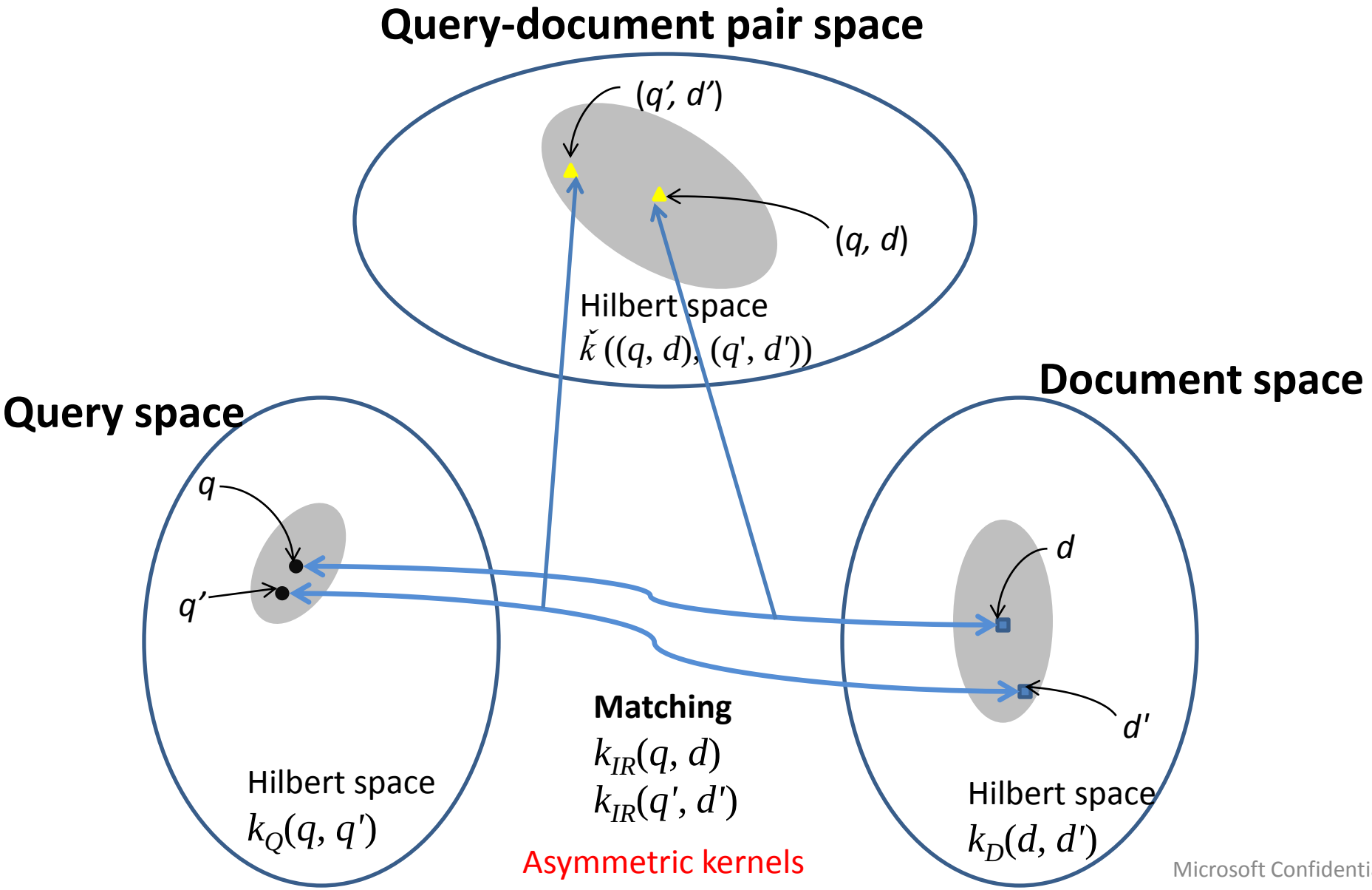
- Solution (called Robust BM25)

$$k_{RBM25}(q,d) = k_{BM25}(q,d) \cdot \sum_{i=1}^{N} \alpha_i k_Q(q,q_i)k_D(d,d_i)k_{BM25}(q_i,d_i)$$

- Deal with term mismatch



Mapping to Space of Query Document Pairs - KNN in New Space



Experimental Results

Table 5: Ranking accuracies on web search and enterprise search data.

		MAP	NDCG@1	NDCG@3	NDCG@5
Web search	Robust BM25	0.1192	0.2480	0.2587	0.2716
	Pairwise Kernel	0.1123	0.2241	0.2418	0.2560
	Query Expansion	0.0963	0.1797	0.2061	0.2237
	BM25	0.0908	0.1728	0.2019	0.2180
Enterprise search	Robust BM25	0.3122	0.4780	0.5065	0.5295
	Pairwise Kernel	0.2766	0.4465	0.4769	0.4971
	Query Expansion	0.2755	0.4076	0.4712	0.4958
	BM25	0.2745	0.4246	0.4531	0.4741

General Principles for Constructing HAK

Theorem 4 (Power Series Construction) *Given two Mercer kernels* $k_X : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ *and* $k_Y : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, *for any asymmetric kernel* $g(x, y)$ *and* $\{c_i\}_{i=1}^n \subset \mathbb{R}^+$, $\bar{k}_P$ *defined below is a hyper asymmetric kernel.*

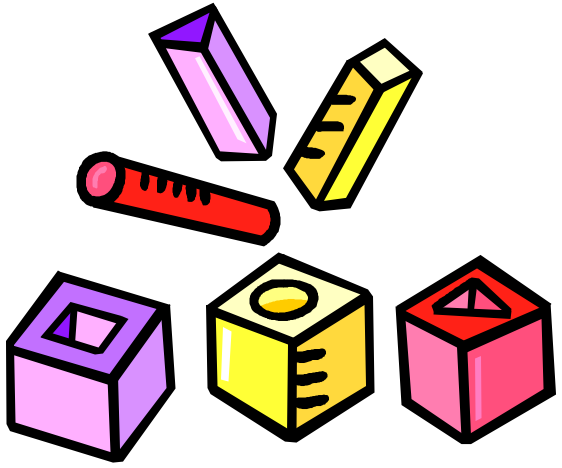
$$\bar{k}_P((x, y), (x', y')) = \sum_{i=0}^{\infty} c_i \cdot g(x, y) \left(k_X(x, x')k_Y(y, y')\right)^i g(x', y'), \tag{7}$$

where the convergence radius of $\sum_{i=0}^{\infty} c_i \xi^i$ *is* R, $|k_X(x, x')| < \sqrt{R}, |k_Y(y, y')| < \sqrt{R}$, *for any* x, x', y, y'.

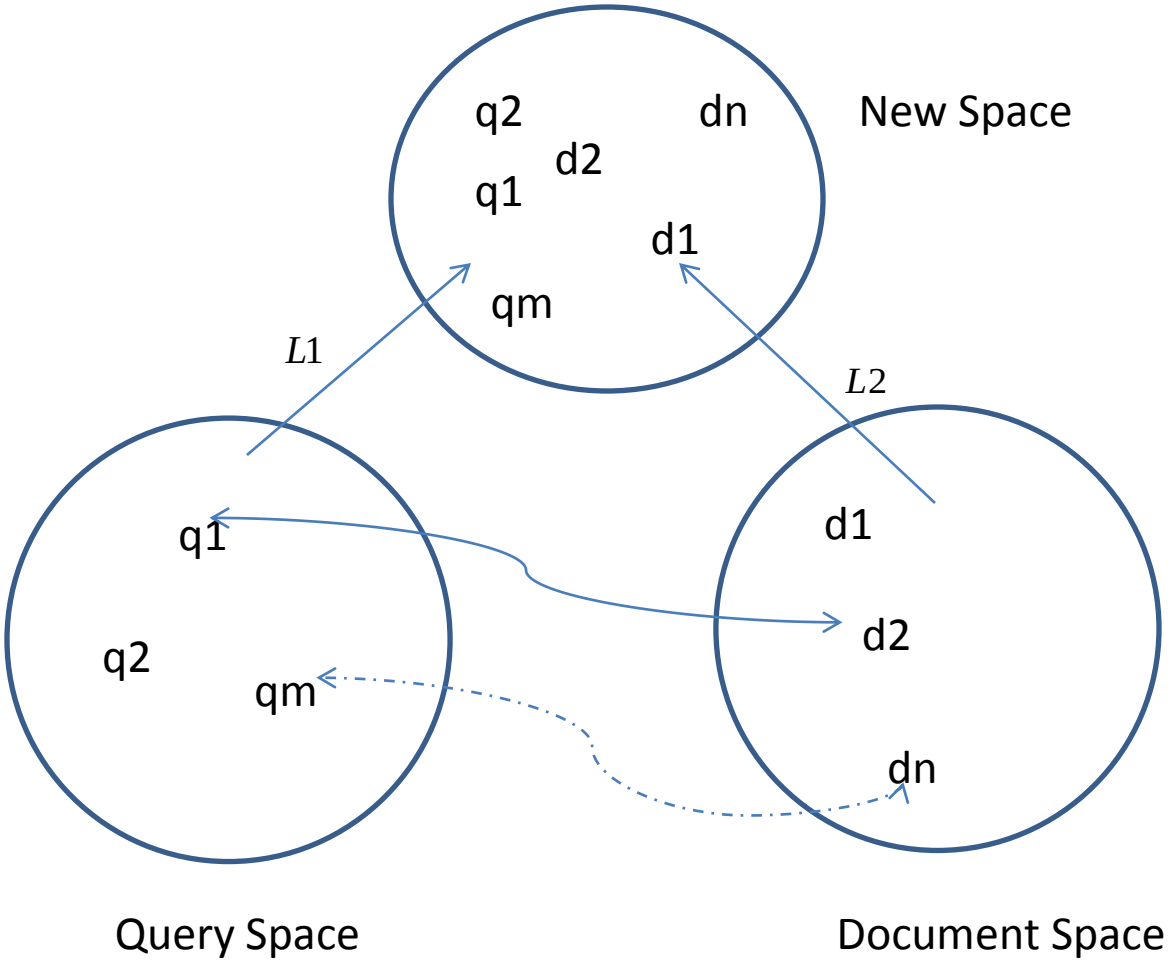
Theorem 5 (Multiple Kernel Construction) *Given two finite sets of Mercer kernels* $K_X = \left\{k_i^X(x, x')\right\}_{i=1}^n$ *and* $K_Y = \left\{k_i^Y(y, y')\right\}_{i=1}^n$. *For any asymmetric kernel* $g(x, y)$ *and* $\{c_i\}_{i=1}^n \subset \mathbb{R}^+$, $\bar{k}_M$ *defined below is a hyper asymmetric kernel.*

$$\bar{k}_M((x, y), (x', y')) = \sum_{i=1}^{n} c_i \cdot g(x, y)k_i^X(x, x')k_i^Y(y, y')g(x', y'). \tag{8}$$

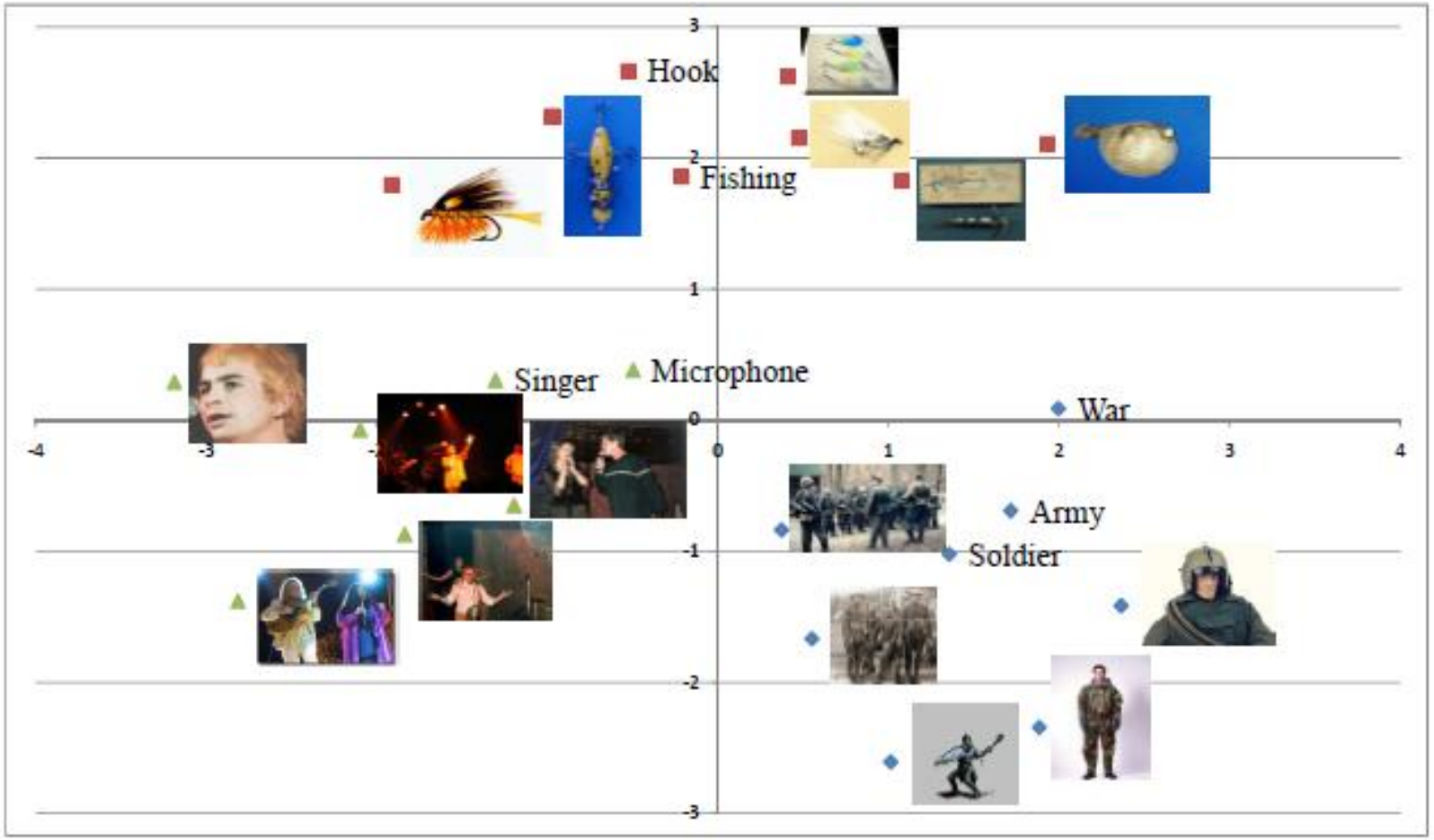
4. Learning of Matching Function



Learning Similarity Function between Objects in Two Spaces



Keywords and Images Represented in Same Space



Problem Formulation

- Setting
  - Two spaces: $\mathcal{X} \subset \mathbb{R}^m$ and $\mathcal{Y} \subset \mathbb{R}^n$.
- Input
  - Training data: $\{(x_i, y_i, t_i)\}_{1 \le i \le N}$
- 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

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

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

Our Solution

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

1: Input: training data $\{(x_i, y_i, r_i)\}_{i=1}^{N}$, parameter $k \leqslant min(n, m)$.
2: Calculate M_S and M_D through $\sum_{r_i=1} y_i x_i^\top$ and $\sum_{r_i=-1} y_i x_i^\top$, respectively.
3: Calculate SVD of $M_S - M_D$.
4: Choose left and right singular vectors $(u_1, \cdots, u_k)$ and $(v_1, \cdots, v_k)$ w.r.t the top k singular values.
5: Output: $L_{\mathcal{Y}} = (u_1, \cdots, u_k)$ and $L_{\mathcal{X}} = (v_1, \cdots, v_k)$.

Talk Outline

- Background
  1. Key Problem in Search = Matching between Query and Document
  2. IR Models (BM25) = Matching Functions
  3. Mismatch Problem
- Problems and Our Solutions
  1. Generic IR Model as Asymmetric Kernel
  2. Extension of IR Models as Asymmetric Kernels
  3. Learning of Robust IR Models as Asymmetric Kernels
  4. Learning of Matching Function

Microsoft Research
微软亚洲研究院

Thank You!