Ranking Methods in Machine Learning

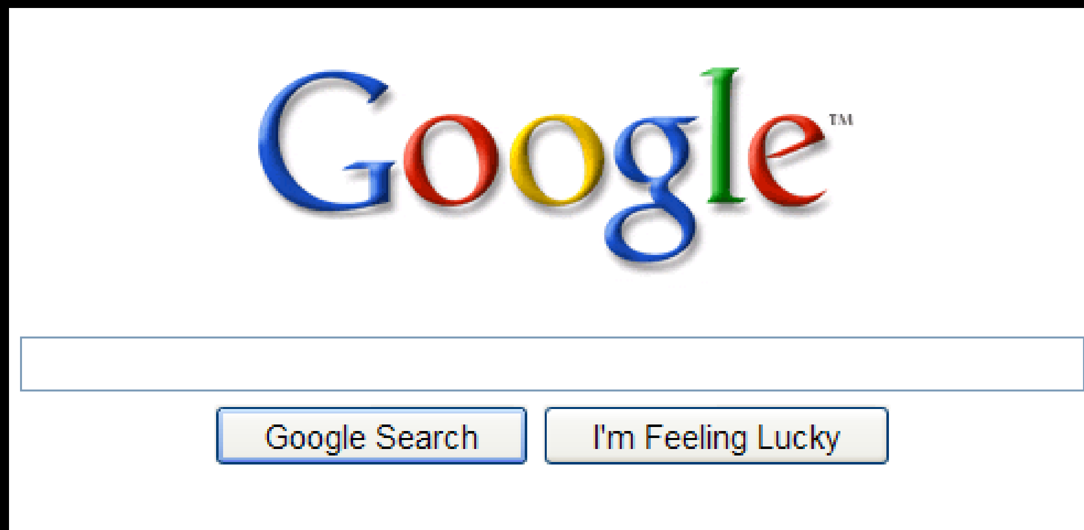
A Tutorial Introduction

Shivani Agarwal

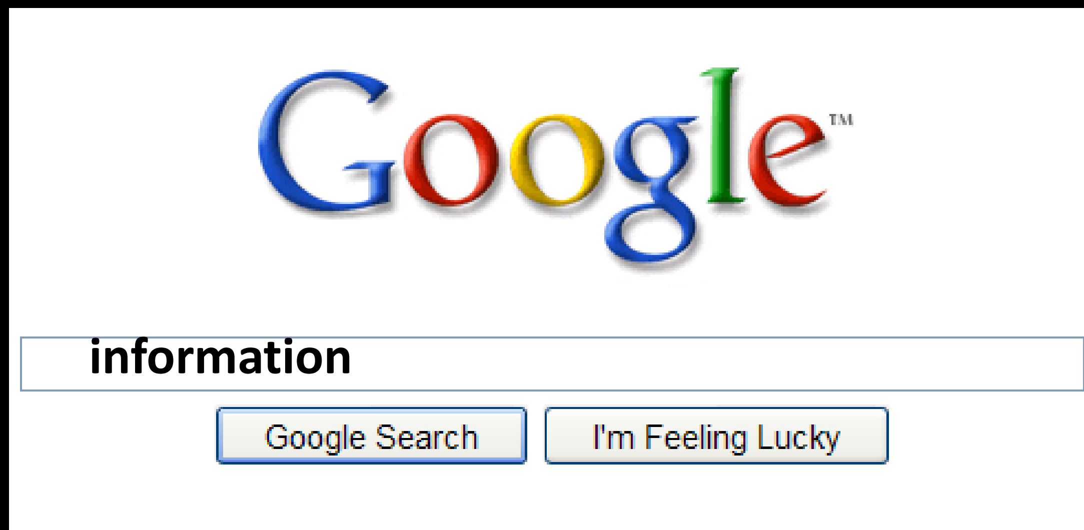
Computer Science & Artificial Intelligence Laboratory
Massachusetts Institute of Technology

Example 1: Recommendation Systems

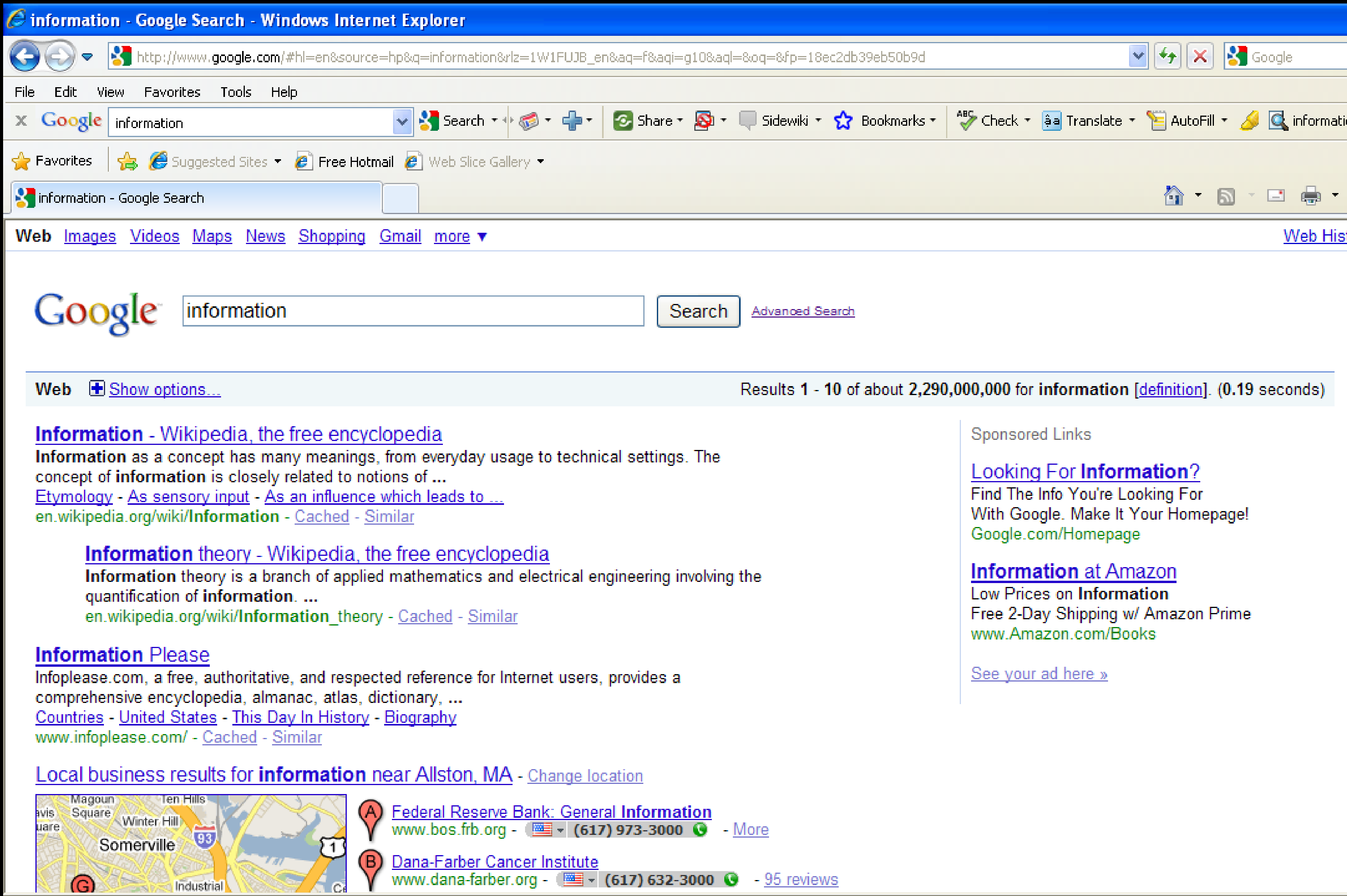
Example 2: Information Retrieval



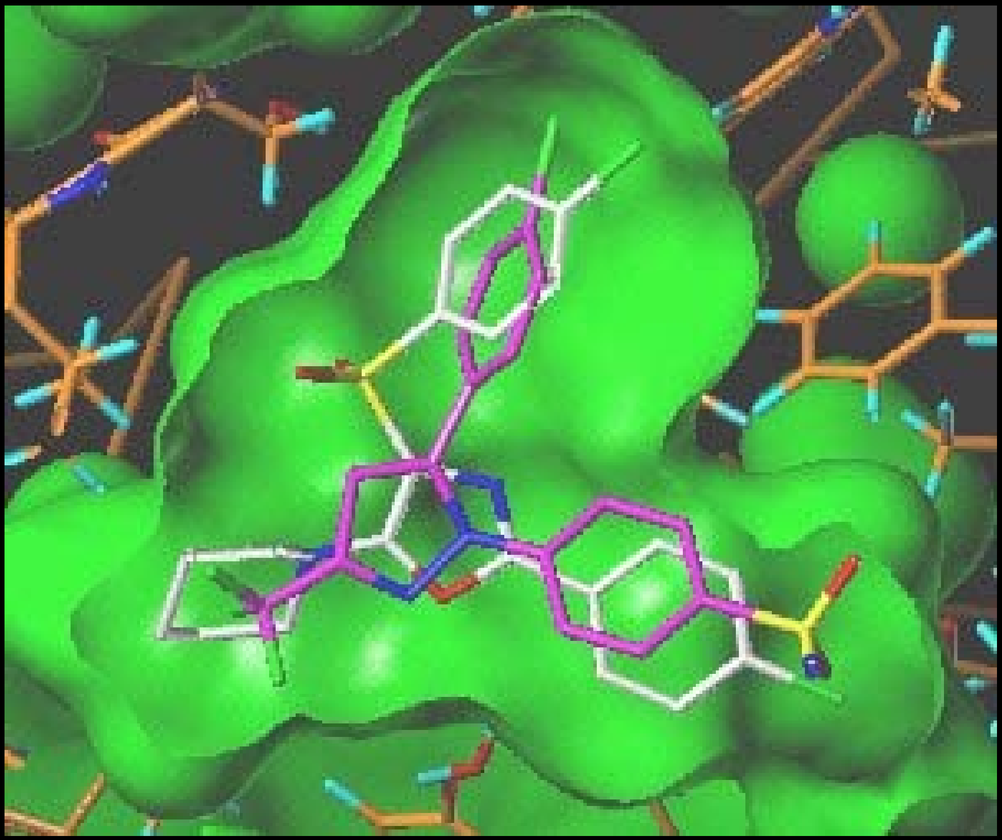
Example 2: Information Retrieval

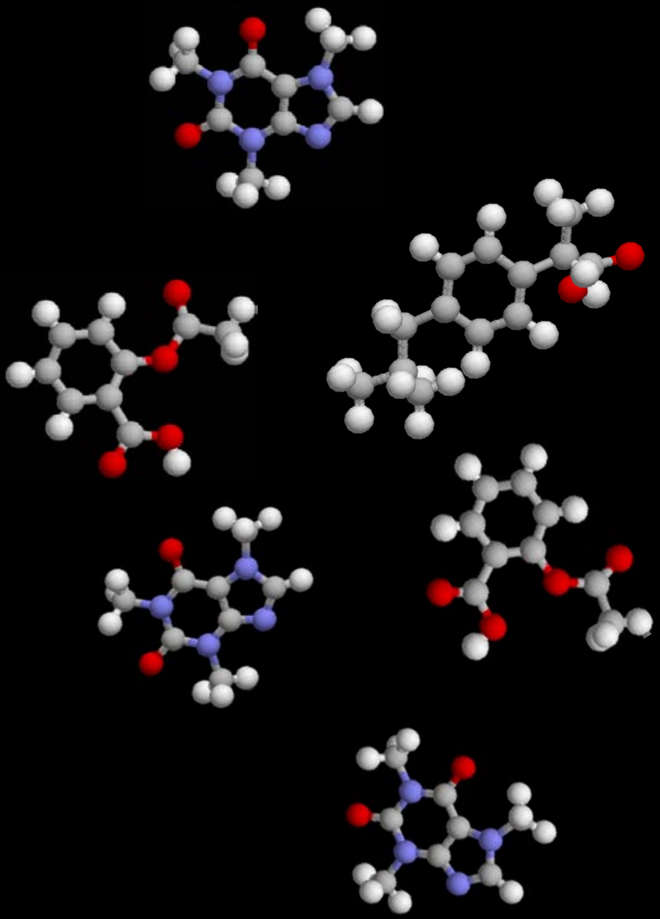


Example 2: Information Retrieval



Example 3: Drug Discovery





Problem: Millions of structures in a chemical library. How do we identify the most promising ones?

Example 4: Bioinformatics

nature

Searching for genetic determinants in the new millennium

N.J. Risch

Human genetics is now at a critical juncture. The molecular methods used successfully to identify the genes underlying rare mendelian syndromes are failing to find the numerous genes causing more common, familial, non-mendelian diseases . . .

Nature **405**:847–856, 2000

Example 4: Bioinformatics

nature

Searching for genetic determinants in the new millennium

N.J. Risch

With the human genome sequence nearing completion, new opportunities are being presented for unravelling the complex genetic basis of nonmendelian disorders based on large-scale genomewide studies . . .

Nature **405**:847–856, 2000

Types of Ranking Problems

Instance Ranking

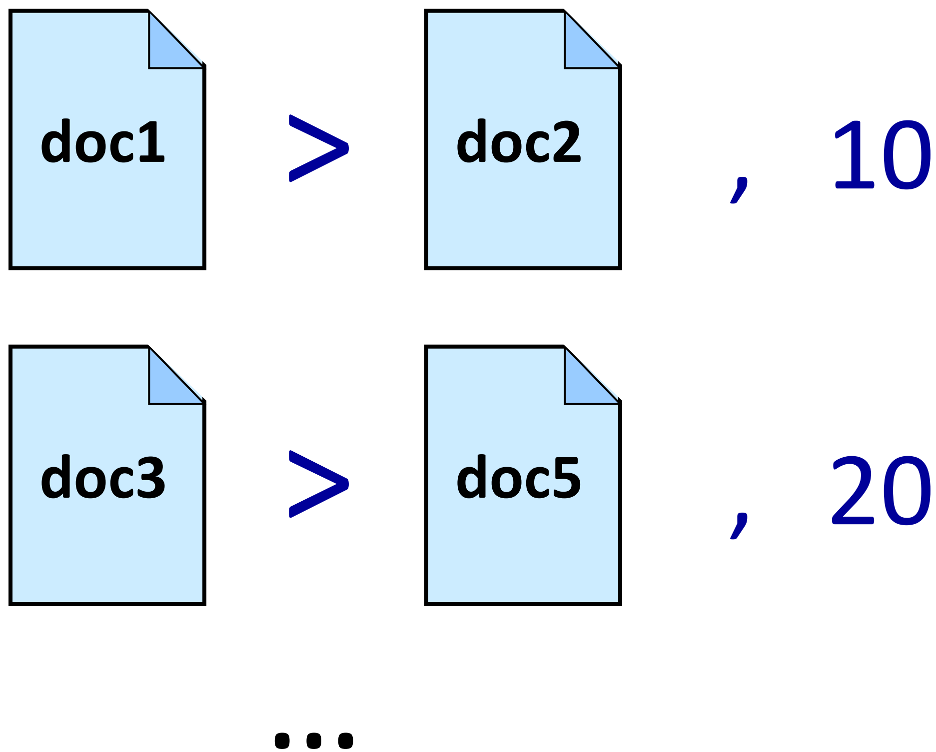
Label Ranking

Subset Ranking

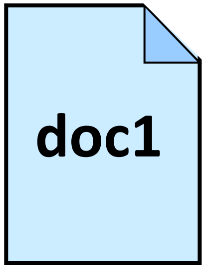
Rank Aggregation

?

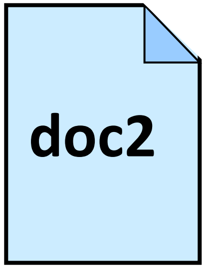
Instance Ranking



Label Ranking



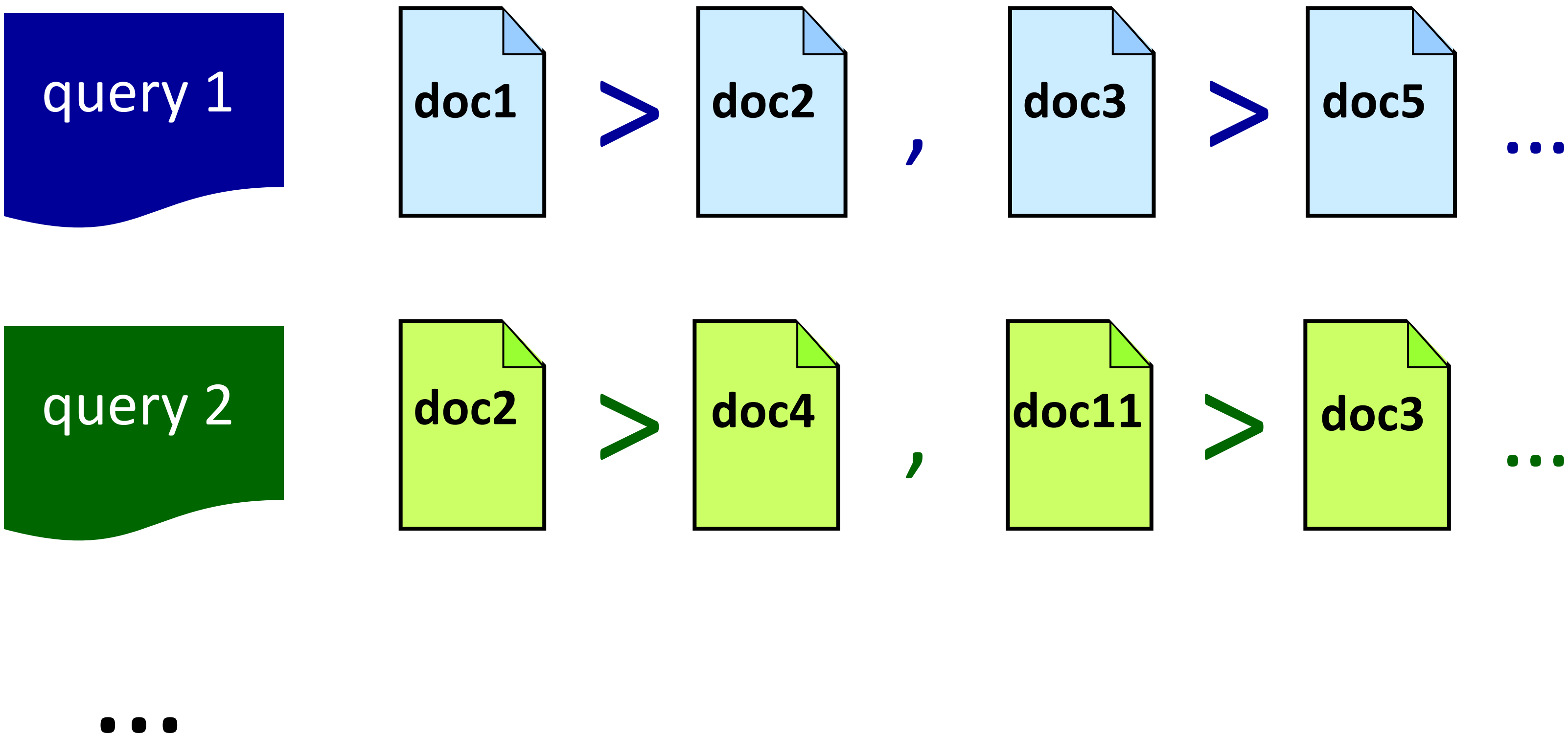
sports > politics
health > money
...



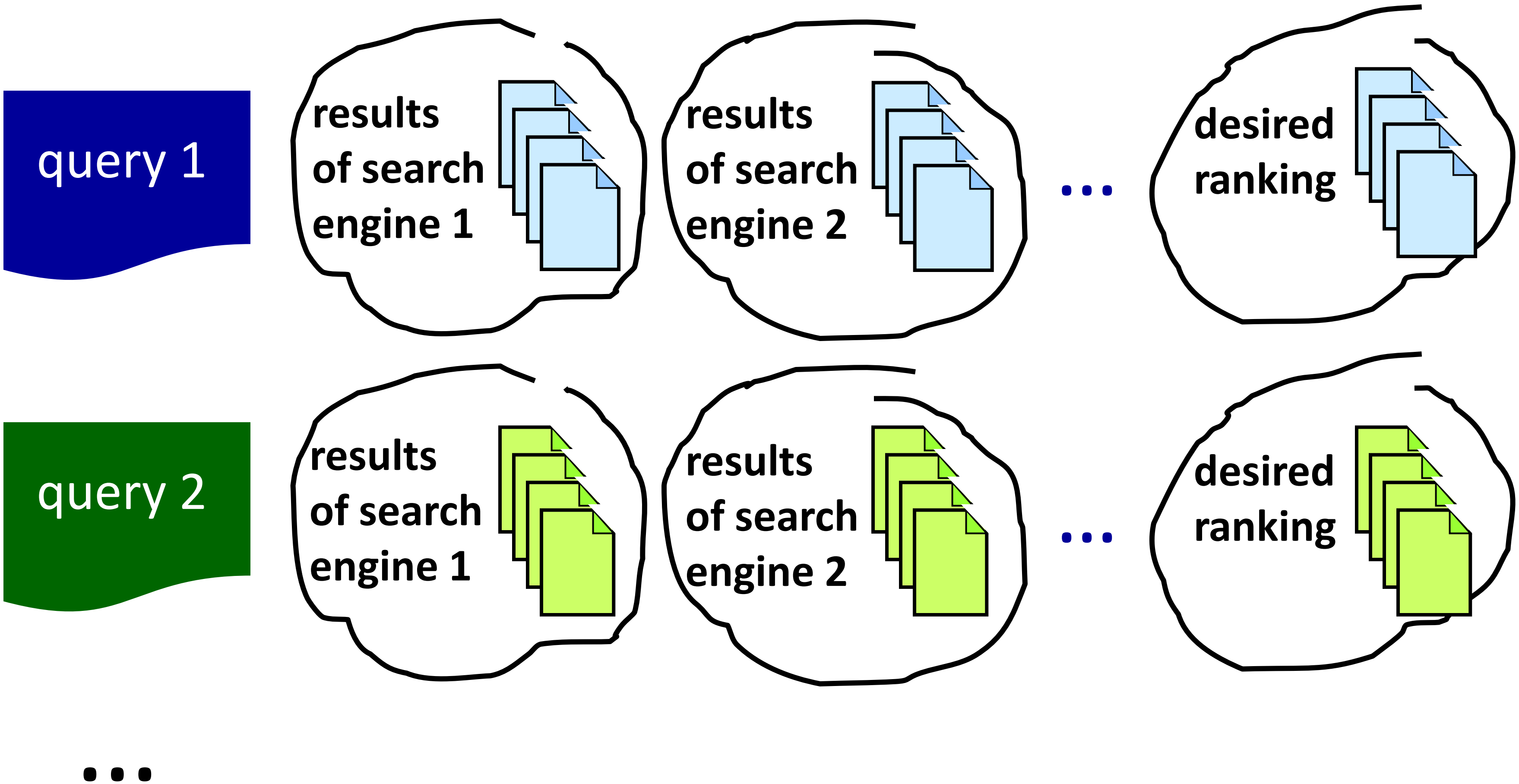
science > sports
money > politics
...

...

Subset Ranking



Rank Aggregation



Types of Ranking Problems

Instance Ranking

Label Ranking

Subset Ranking

Rank Aggregation

?

This tutorial

Tutorial Road Map

Part I: Theory & Algorithms

Bipartite Ranking
k-partite Ranking
Ranking with Real-Valued Labels
General Instance Ranking

RankSVM
RankBoost
RankNet

Part II: Applications

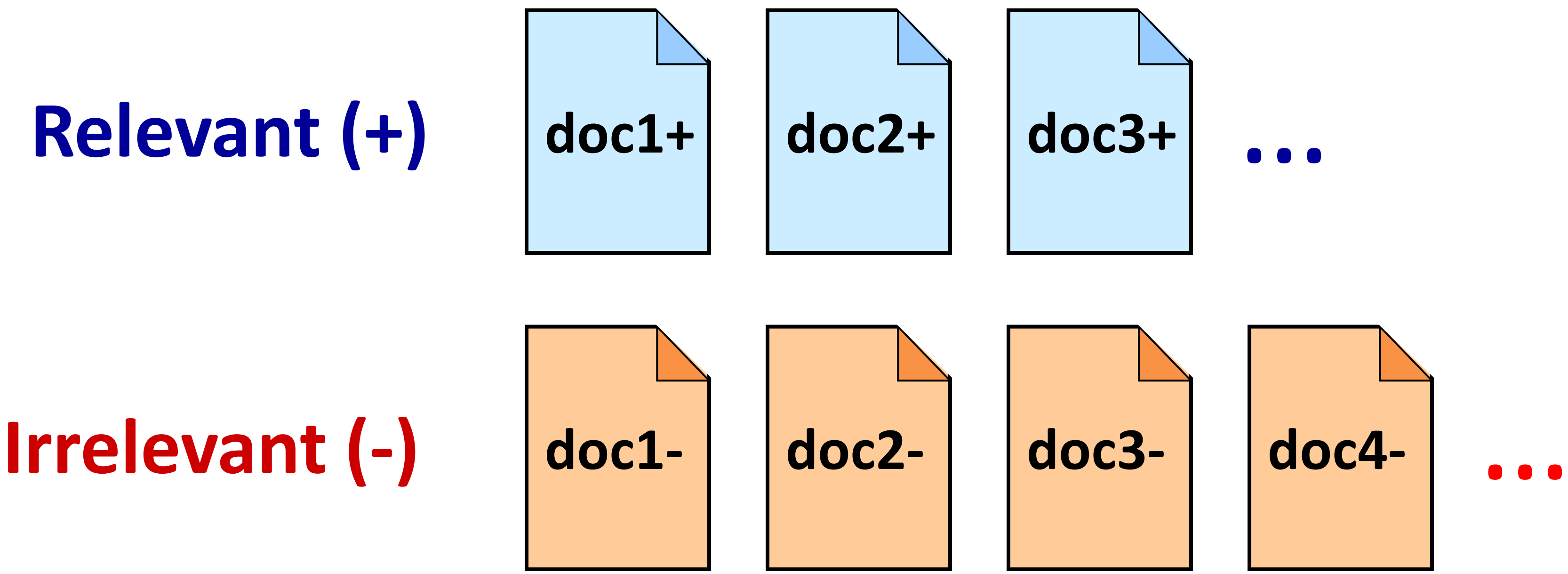
Applications to Bioinformatics
Applications to Drug Discovery
Subset Ranking and Applications to Information Retrieval

Further Reading & Resources

Part I
Theory & Algorithms

[for Instance Ranking]

Bipartite Ranking

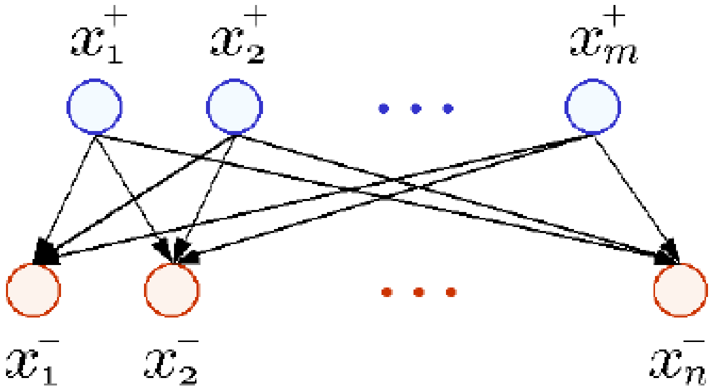


Bipartite Ranking

- Instance space X
- Input: Training sample $S = (S_+, S_-)$:

$$S_+ = (x_1^+, \ldots, x_m^+) \in X^m \quad \text{(positive examples)}$$
$$S_- = (x_1^-, \ldots, x_n^-) \in X^n \quad \text{(negative examples)}$$

- Output: Ranking function $f : X \rightarrow \mathbb{R}$



Bipartite Ranking

- Instance space X
- Input: Training sample $S = (S_+, S_-)$:

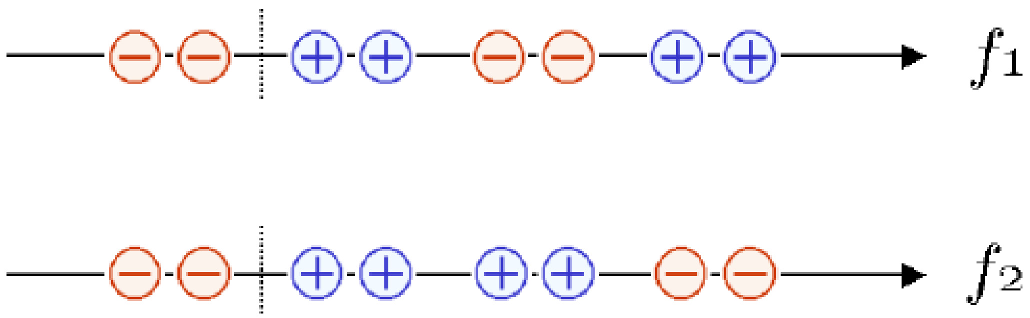
$$S_+ = (x_1^+, \ldots, x_m^+) \in X^m \quad \text{(positive examples)}$$
$$S_- = (x_1^-, \ldots, x_n^-) \in X^n \quad \text{(negative examples)}$$

- Output: Ranking function $f : X \rightarrow \mathbb{R}$

- Expected error: $\mathbf{er}(f) = \mathbf{P}_{(x,x')\sim\mathcal{D}_+\times\mathcal{D}_-}\left[f(x) < f(x')\right]$
- Empirical error: $\widehat{\mathbf{er}}_S(f) = \dfrac{1}{mn}\displaystyle\sum_{i=1}^{m}\sum_{j=1}^{n} \mathbf{1}(f(x_i^+) < f(x_j^-))$

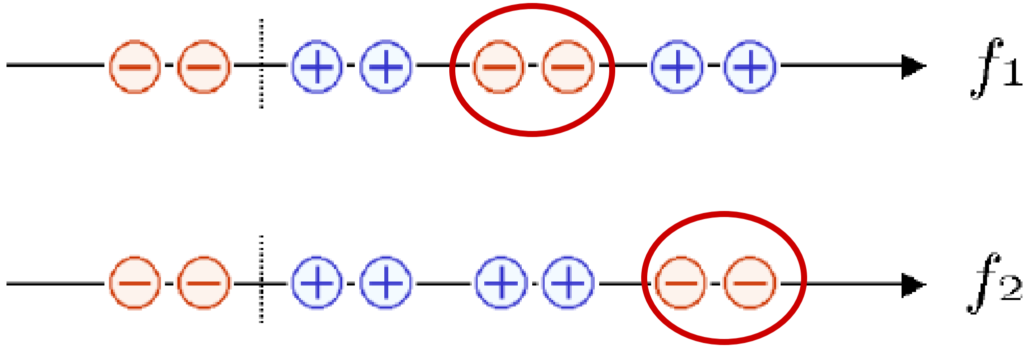
Is Bipartite Ranking Different from Binary Classification?

Example 1



Is Bipartite Ranking Different from Binary Classification?

Example 1

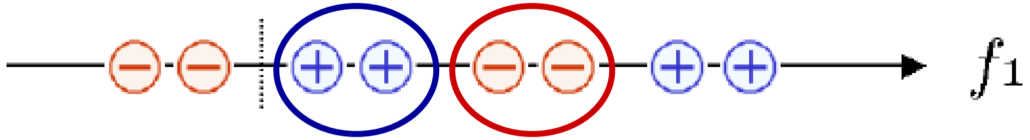


Classification error = $\frac{1}{4}$

Classification error = $\frac{1}{4}$

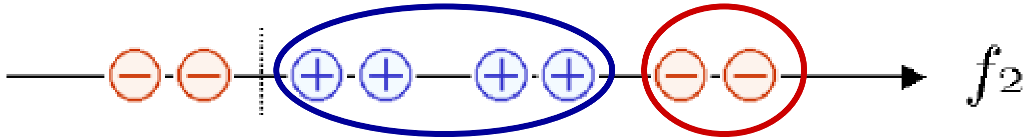
Is Bipartite Ranking Different from Binary Classification?

Example 1



Classification error = $\frac{1}{4}$

Ranking error = $\frac{1}{4}$

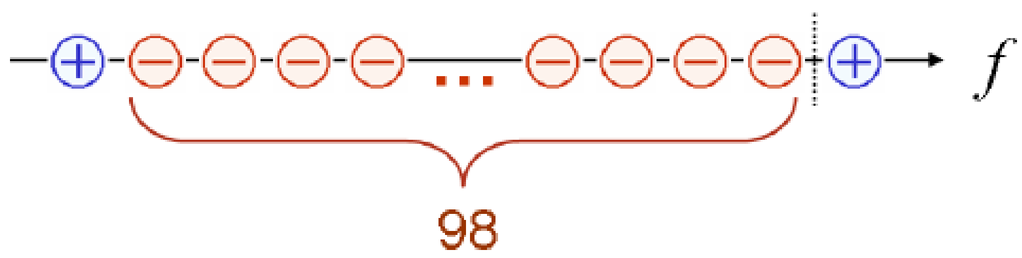


Classification error = $\frac{1}{4}$

Ranking error = $\frac{1}{2}$

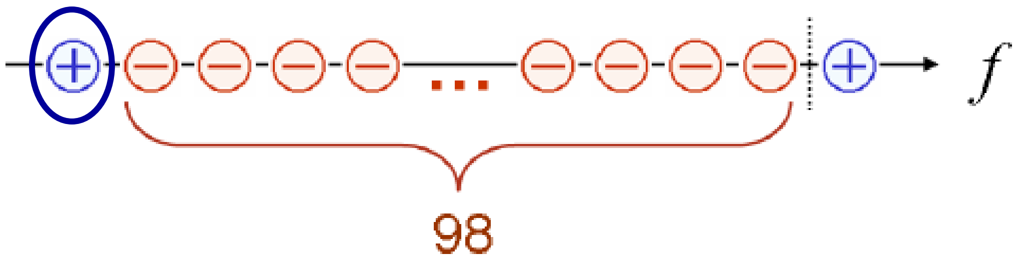
Is Bipartite Ranking Different from Binary Classification?

Example 2



Is Bipartite Ranking Different from Binary Classification?

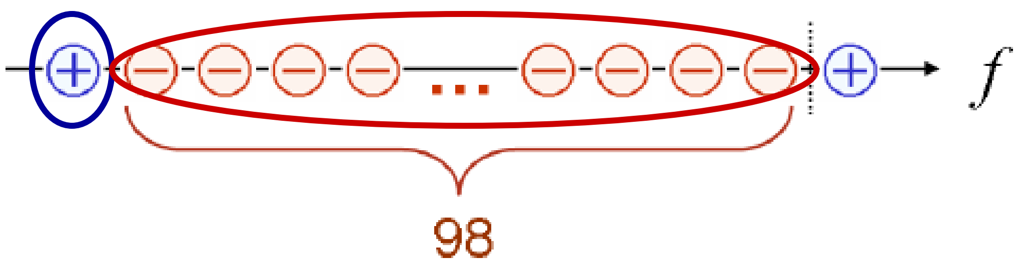
Example 2



Classification error = $\frac{1}{100}$

Is Bipartite Ranking Different from Binary Classification?

Example 2



Classification error = $\frac{1}{100}$

Ranking error = $\frac{1}{2}$

Bipartite Ranking: Basic Algorithmic Framework

Minimize a convex upper bound on the empirical ranking error, possibly with some regularization, over some class of ranking functions:

$$\min_{f \in \mathcal{F}} \left[\frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \ell(f, x_i^+, x_j^-) + \lambda N(f) \right]$$

where

$\ell(f, x_i^+, x_j^-)$	:	convex upper bound on $\mathbf{1}(f(x_i^+) < f(x_j^-))$
$N(f)$	:	regularizer
$\lambda > 0$	:	regularization parameter
$\mathcal{F}$	:	class of ranking functions

Bipartite RankSVM Algorithm

$$\min_{f \in \mathcal{F}_K} \left[\frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \ell_{\text{hinge}}(f, x_i^+, x_j^-) + \frac{\lambda}{2} \|f\|_K^2 \right]$$

$$\ell_{\text{hinge}}(f, x_i^+, x_j^-) = \left(1 - \left(f(x_i^+) - f(x_j^-)\right)\right)_+ \quad \left[u_+ = \max(u, 0)\right]$$

$\mathcal{F}_K$ = reproducing kernel Hilbert space (RKHS) with kernel function K

$$N(f) = \frac{\|f\|_K^2}{2}$$

[Herbrich et al, 2000; Joachims, 2002; Rakotomamonjy, 2004]

Bipartite RankSVM Algorithm

Introducing slack variables and taking the Lagrangian dual results in the following convex quadratic program (QP) over mn variables $\{\alpha_{ij} : 1 \leq i \leq m, 1 \leq j \leq n\}$:

$$\min_{\alpha} \left[\frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{m} \sum_{l=1}^{n} \alpha_{ij}\alpha_{kl}\, \phi(x_i^+, x_j^-, x_k^+, x_l^-) - \sum_{i=1}^{m} \sum_{j=1}^{n} \alpha_{ij} \right]$$

subject to $0 \leq \alpha_{ij} \leq C \quad \forall\, i, j$

where

$$\phi(x_i^+, x_j^-, x_k^+, x_l^-) = \left(K(x_i^+, x_k^+) - K(x_i^+, x_l^-) - K(x_j^-, x_k^+) + K(x_j^-, x_l^-) \right)$$

$$C = \frac{1}{\lambda m n}$$

Can be solved using a standard QP solver, or more efficient methods (e.g. Chapelle & Keerthi, 2010).

Bipartite RankBoost Algorithm

$$\min_{f \in \mathcal{L}(\mathcal{F}_{\text{base}})} \left[\frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \ell_{\text{exp}}(f, x_i^+, x_j^-) \right]$$

$$\ell_{\text{exp}}(f, x_i^+, x_j^-) \;=\; \exp\Big(-\Big(f(x_i^+) - f(x_j^-)\Big)\Big)$$

$\mathcal{L}(\mathcal{F}_{\text{base}})$ = linear combinations of functions in some base class $\mathcal{F}_{\text{base}}$

[Freund et al, 2003]

Bipartite RankBoost Algorithm

Input: $(S_+, S_-) \in X^m \times X^n$.

Initialize: $D_1(x_i^+, x_j^-) = \dfrac{1}{mn}$ for all $i \in \{1, \ldots, m\},\ j \in \{1, \ldots, n\}$.

For $t = 1, \ldots, T$:

- Train weak learner using distribution D_t; get weak ranker $f_t \in \mathcal{F}_{\text{base}}$.
- Choose $\alpha_t \in \mathbb{R}$.
- Update: $D_{t+1}(x_i^+, x_j^-) = \dfrac{1}{Z_t} D_t(x_i^+, x_j^-) \exp\left(-\alpha_t \left(f_t(x_i^+) - f_t(x_j^-)\right)\right)$

  where $Z_t = \displaystyle\sum_{i=1}^{m} \sum_{j=1}^{n} D_t(x_i^+, x_j^-) \exp\left(-\alpha_t \left(f_t(x_i^+) - f_t(x_j^-)\right)\right)$.

Output final ranking: $f(x) = \displaystyle\sum_{t=1}^{T} \alpha_t f_t(x)$.

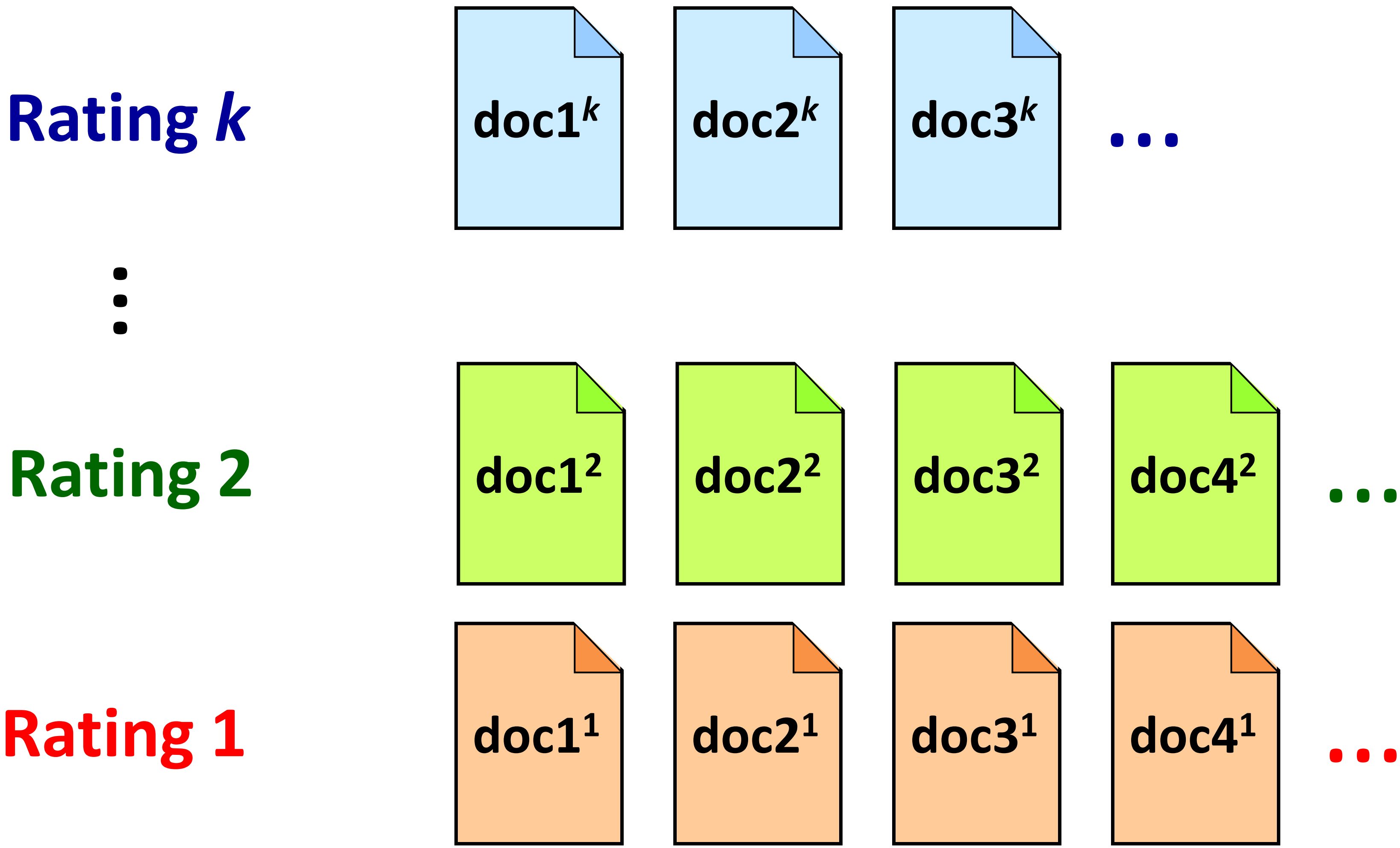
Bipartite RankNet Algorithm

$$\min_{f \in \mathcal{F}_{\text{neural}}} \left[\frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \ell_{\text{logistic}}(f, x_i^+, x_j^-) \right]$$

$$\begin{aligned} \ell_{\text{logistic}}(f, x_i^+, x_j^-) &= \log\left(1 + \exp\left(-\left(f(x_i^+) - f(x_j^-)\right)\right)\right) \\ \mathcal{F}_{\text{neural}} &= \text{functions represented by some class of} \\ & \quad \text{neural networks} \end{aligned}$$

[Burges et al, 2005]

k-partite Ranking



k-partite Ranking

- Instance space X
- Input: Training sample $S = (S_1, S_2, \ldots, S_k)$:

$$S_k = (x_1^k, \ldots, x_{n_k}^k) \in X^{n_k} \quad \text{(examples of rating } k\text{)}$$
$$\vdots$$
$$S_2 = (x_1^2, \ldots, x_{n_2}^2) \in X^{n_2} \quad \text{(examples of rating } 2\text{)}$$
$$S_1 = (x_1^1, \ldots, x_{n_1}^1) \in X^{n_1} \quad \text{(examples of rating } 1\text{)}$$

- Output: Ranking function $f : X \rightarrow \mathbb{R}$

- Empirical error:

$$\widehat{\mathbf{er}}_S(f) = \left(\frac{1}{\sum\limits_{1 \le a < b \le k} n_a n_b} \right) \sum_{1 \le a < b \le k} \sum_{i=1}^{n_b} \sum_{j=1}^{n_a} (b-a)\, \mathbf{1}\left(f(x_i^b) < f(x_j^a)\right)$$

k-partite Ranking: Basic Algorithmic Framework

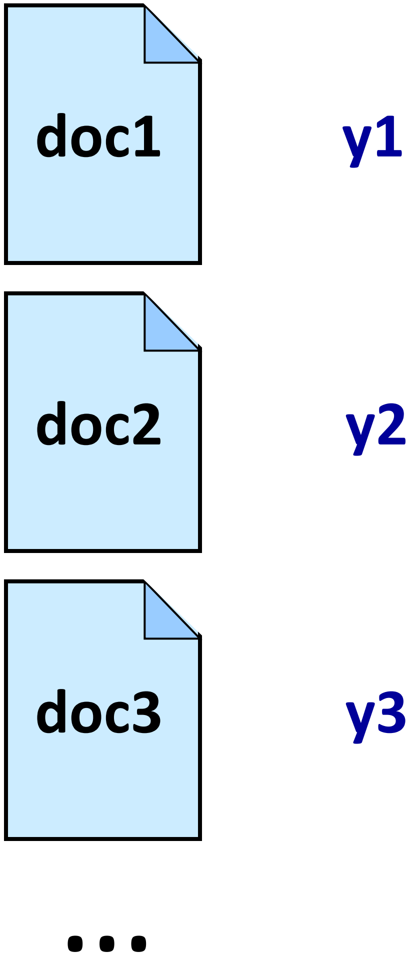
Minimize a convex upper bound on the empirical ranking error, possibly with some regularization, over some class of ranking functions:

$$\min_{f \in \mathcal{F}} \left[\left(\frac{1}{\sum_{1 \leq a < b \leq k} n_a n_b} \right) \sum_{1 \leq a < b \leq k} \sum_{i=1}^{n_b} \sum_{j=1}^{n_a} \ell(f, x_i^b, x_j^a, (b-a)) + \lambda N(f) \right]$$

where

$$\begin{aligned}
\ell(f, x_i^b, x_j^a, (b-a)) \quad &: \quad \text{convex upper bound on } (b-a)\,\mathbf{1}(f(x_i^b) < f(x_j^a)) \\
N(f) \quad &: \quad \text{regularizer} \\
\lambda > 0 \quad &: \quad \text{regularization parameter} \\
\mathcal{F} \quad &: \quad \text{class of ranking functions}
\end{aligned}$$

Ranking with Real-Valued Labels



Ranking with Real-Valued Labels

- Instance space X
- Real-valued labels $Y = \mathbb{R}$
- Input: Training sample $S = ((x_1, y_1), \ldots, (x_m, y_m)) \in (X \times \mathbb{R})^m$
- Output: Ranking function $f : X \to \mathbb{R}$
- Empirical error:

$$\widehat{\mathbf{er}}_S(f) = \frac{1}{\binom{m}{2}} \sum_{1 \leq i < j \leq m} |y_i - y_j| \, \mathbf{1}\left((y_i - y_j)(f(x_i) - f(x_j)) < 0\right)$$

Ranking with Real-Valued Labels: Basic Algorithmic Framework

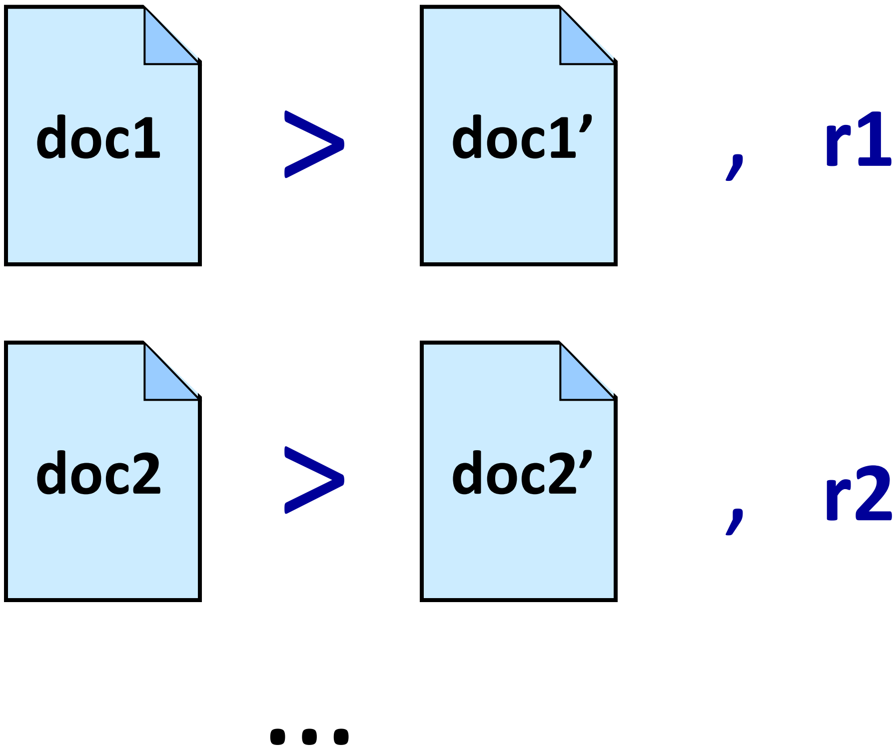
Minimize a convex upper bound on the empirical ranking error, possibly with some regularization, over some class of ranking functions:

$$\min_{f \in \mathcal{F}} \left[\frac{1}{\binom{m}{2}} \sum_{1 \le i < j \le m} \ell(f, (x_i, y_i), (x_j, y_j)) \; + \; \lambda N(f) \right]$$

where

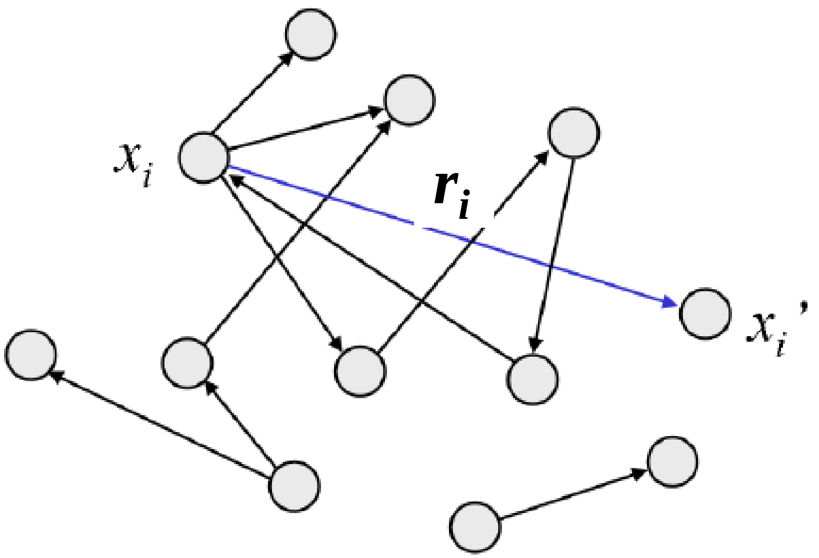
$$\begin{array}{rcl}
\ell(f, (x_i, y_i), (x_j, y_j)) & : & \text{convex upper bound on} \\
 & & |y_i - y_j| \, \mathbf{1}\left((y_i - y_j)(f(x_i) - f(x_j)) < 0\right) \\
N(f) & : & \text{regularizer} \\
\lambda > 0 & : & \text{regularization parameter} \\
\mathcal{F} & : & \text{class of ranking functions}
\end{array}$$

General Instance Ranking



General Instance Ranking

- Instance space X
- Input: Training sample $S = ((x_1, x'_1, r_1), \ldots, (x_m, x'_m, r_m)) \in (X^2 \times \mathbb{R}_+)^m$
- Output: Ranking function $f : X \rightarrow \mathbb{R}$



General Instance Ranking

- Instance space X
- Input: Training sample $S = ((x_1, x'_1, r_1), \ldots, (x_m, x'_m, r_m)) \in (X^2 \times \mathbb{R}_+)^m$
- Output: Ranking function $f : X \to \mathbb{R}$

- Empirical error: $\widehat{\mathbf{er}}_S(f) = \frac{1}{m} \sum_{i=1}^{m} r_i \, \mathbf{1}(f(x_i) < f(x'_i))$

General Instance Ranking: Basic Algorithmic Framework

Minimize a convex upper bound on the empirical ranking error, possibly with some regularization, over some class of ranking functions:

$$\min_{f \in \mathcal{F}} \left[\frac{1}{m} \sum_{i=1}^{m} \ell(f, x_i, x_i', r_i) + \lambda N(f) \right]$$

where

$\ell(f, x_i, x_i', r_i)$	:	convex upper bound on $r_i \mathbf{1}(f(x_i) < f(x_i'))$
$N(f)$	:	regularizer
$\lambda > 0$	:	regularization parameter
$\mathcal{F}$	:	class of ranking functions

General RankSVM Algorithm

$$\min_{f \in \mathcal{F}_K} \left[\frac{1}{m} \sum_{i=1}^{m} \ell_{\text{hinge}}(f, x_i, x_i', r_i) + \frac{\lambda}{2} \|f\|_K^2 \right]$$

$$\ell_{\text{hinge}}(f, x_i, x_i', r_i) = \Big(r_i - \big(f(x_i) - f(x_i') \big) \Big)_+ \quad \big[u_+ = \max(u, 0) \big]$$

$\mathcal{F}_K$ = reproducing kernel Hilbert space (RKHS) with kernel function K

$$N(f) = \frac{\|f\|_K^2}{2}$$

[Herbrich et al, 2000; Joachims, 2002]

General RankBoost Algorithm

$$\min_{f \in \mathcal{L}(\mathcal{F}_{\text{base}})} \left[\frac{1}{m} \sum_{i=1}^{m} \ell_{\text{exp}}(f, x_i, x_i', r_i) \right]$$

$$\begin{aligned} \ell_{\text{exp}}(f, x_i, x_i', r_i) &= r_i \exp\left(-\left(f(x_i) - f(x_i')\right)\right) \\ \mathcal{L}(\mathcal{F}_{\text{base}}) &= \text{linear combinations of functions in some} \\ & \quad \text{base class } \mathcal{F}_{\text{base}} \end{aligned}$$

[Freund et al, 2003]

General RankNet Algorithm

$$\min_{f \in \mathcal{F}_{\text{neural}}} \left[\frac{1}{m} \sum_{i=1}^{m} \ell_{\text{logistic}}(f, x_i, x_i', r_i) \right]$$

$$\begin{aligned} \ell_{\text{logistic}}(f, x_i, x_i', r_i) &= r_i \log\left(1 + \exp\left(-\left(f(x_i) - f(x_i')\right)\right)\right) \\ \mathcal{F}_{\text{neural}} &= \text{functions represented by some class of} \\ &\quad \text{neural networks} \end{aligned}$$

[Burges et al, 2005]

Tutorial Road Map

Part I: Theory & Algorithms

Bipartite Ranking
k-partite Ranking
Ranking with Real-Valued Labels
General Instance Ranking

RankSVM
RankBoost
RankNet

Part II: Applications

Applications to Bioinformatics
Applications to Drug Discovery
Subset Ranking and Applications to Information Retrieval

Further Reading & Resources

Part II
Applications

[and Subset Ranking]

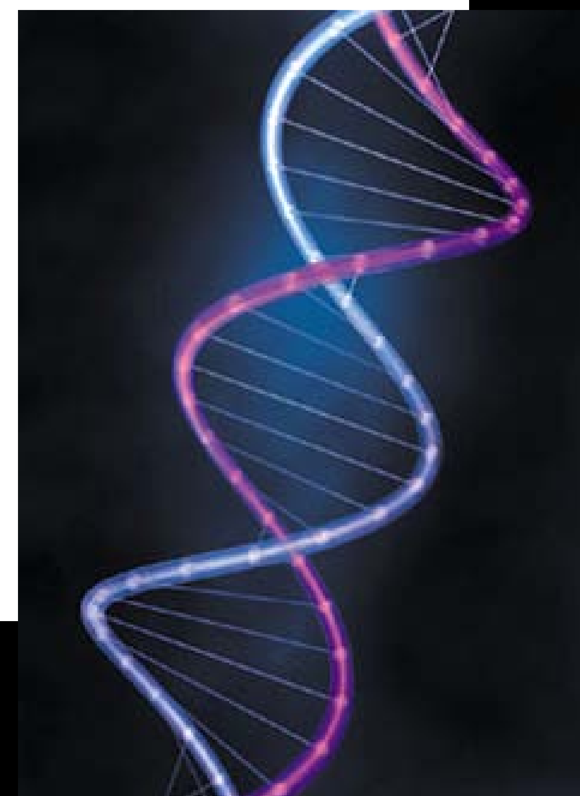
Application to Bioinformatics

nature

Searching for genetic determinants in the new millennium

N.J. Risch

Human genetics is now at a critical juncture. The molecular methods used successfully to identify the genes underlying rare mendelian syndromes are failing to find the numerous genes causing more common, familial, non-mendelian diseases . . .



Nature **405**:847–856, 2000

Application to Bioinformatics

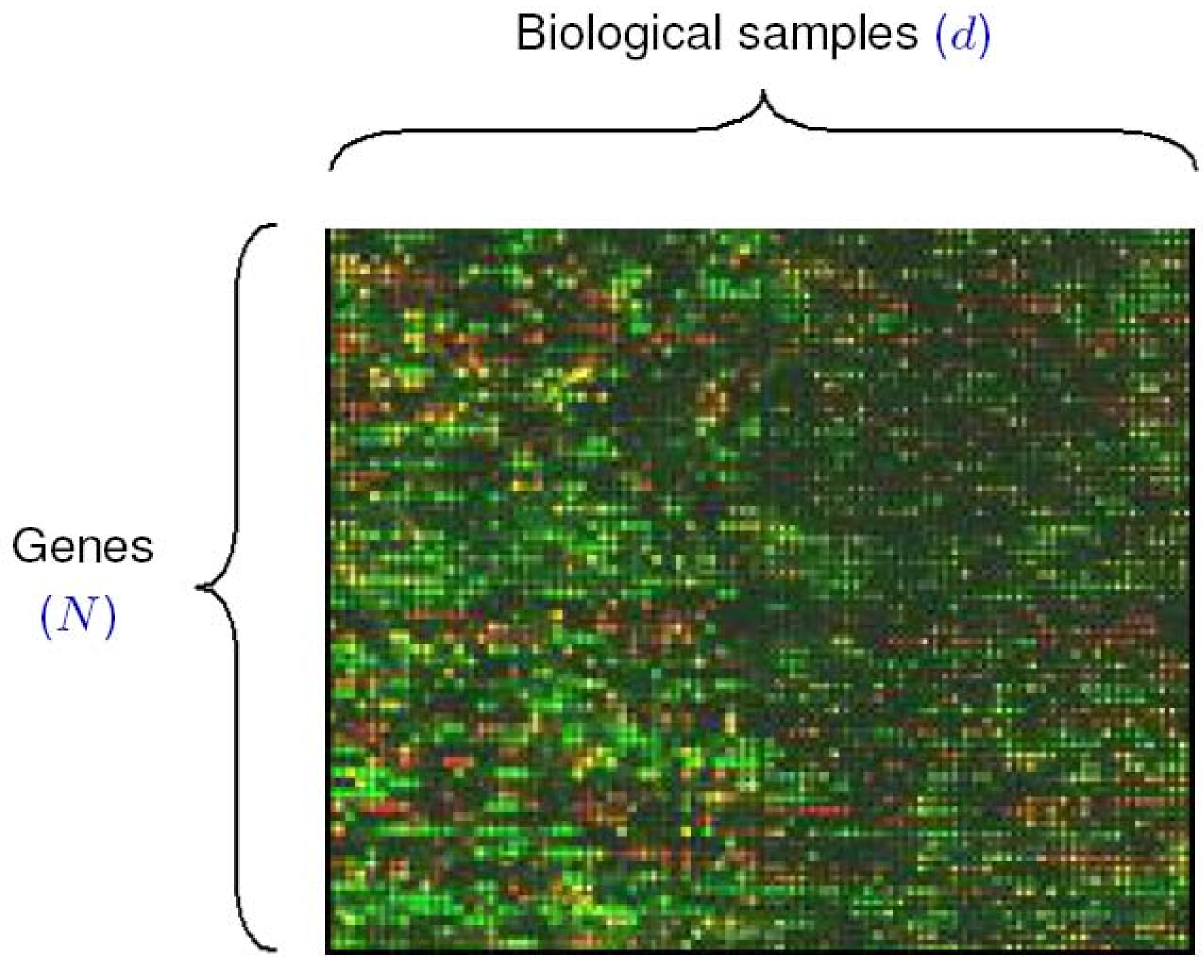
Searching for genetic determinants in the new millennium

N.J. Risch

With the human genome sequence nearing completion, new opportunities are being presented for unravelling the complex genetic basis of nonmendelian disorders based on large-scale genomewide studies . . .

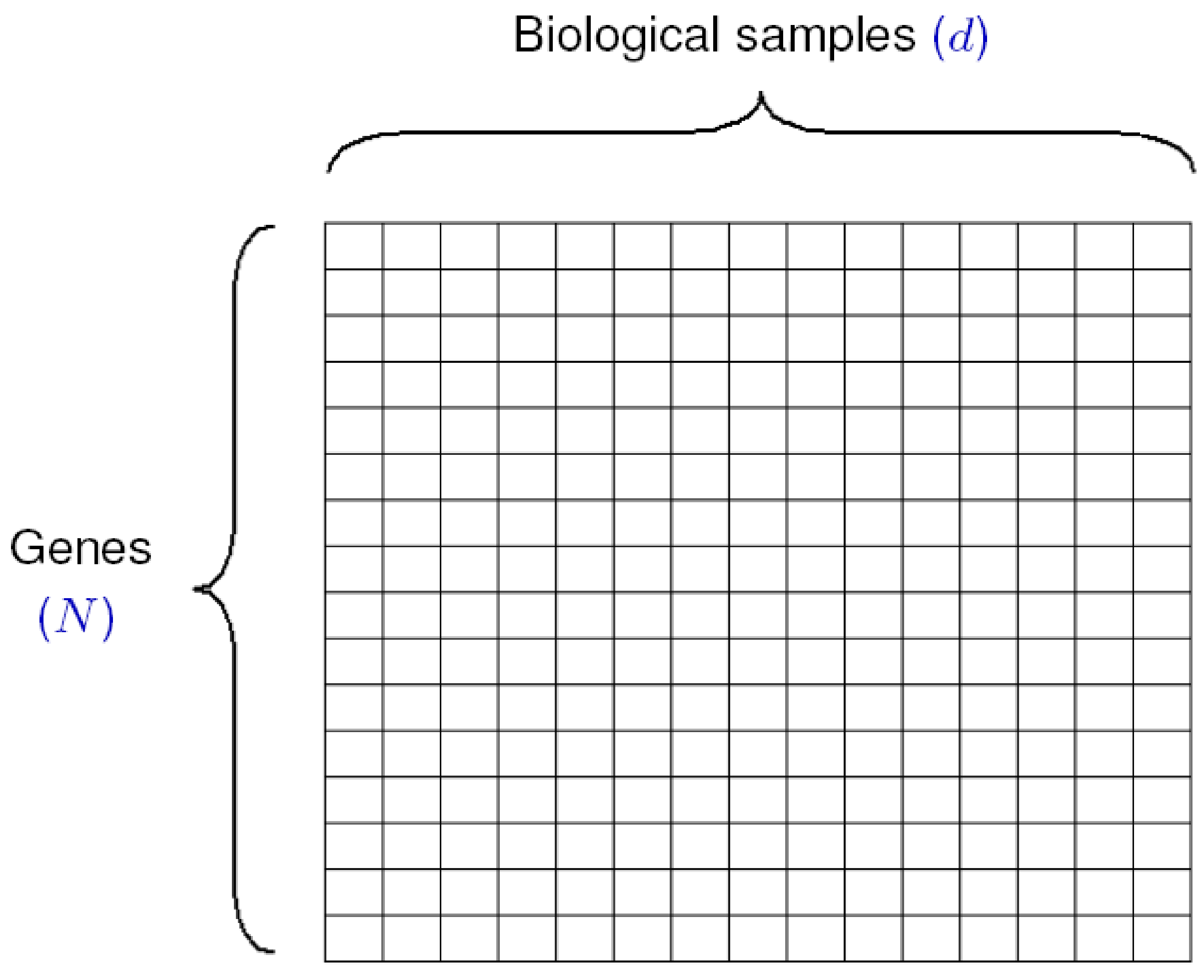
Nature **405**:847–856, 2000

Identifying Genes Relevant to a Disease Using Microarrray Gene Expression Data



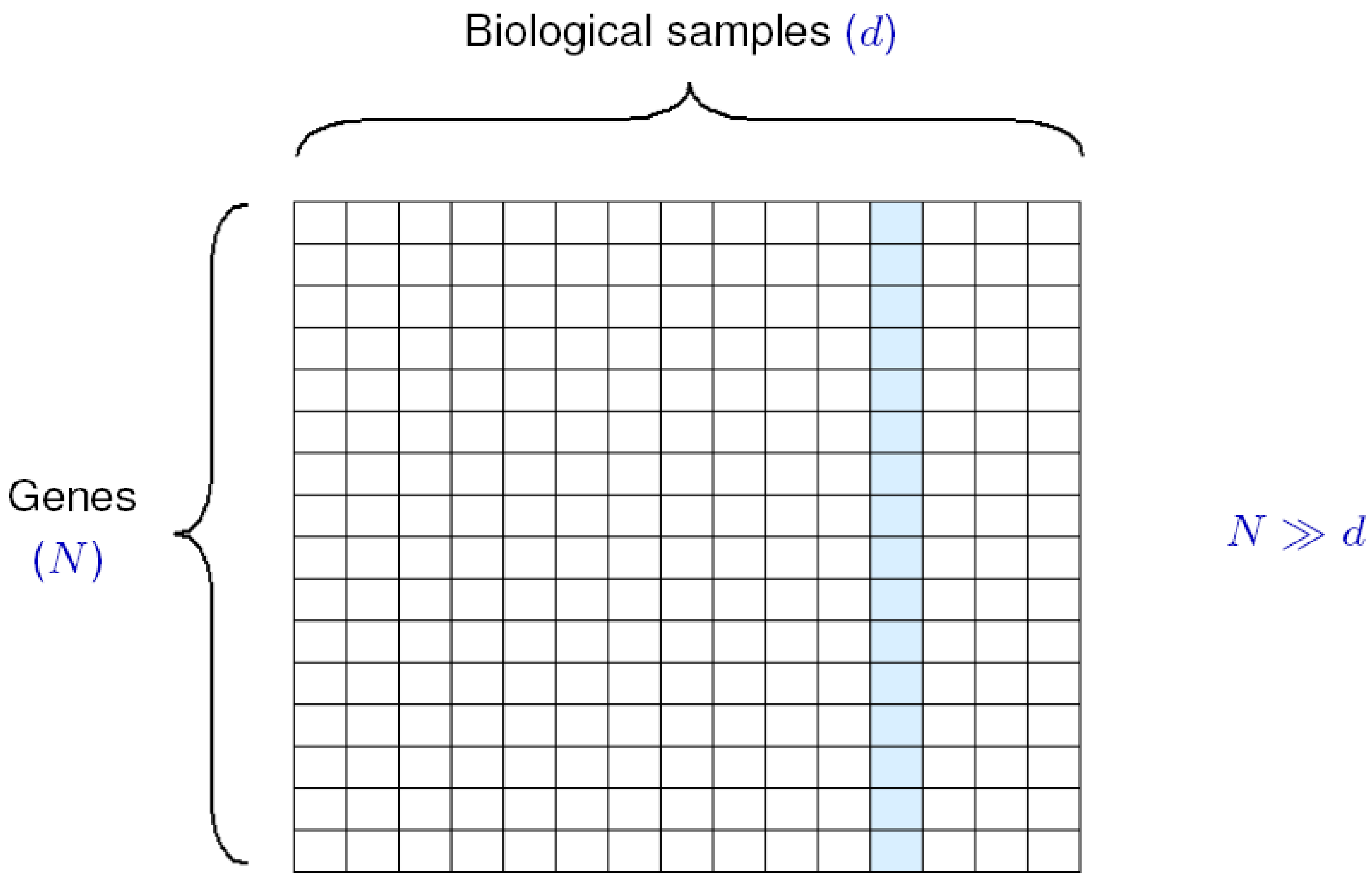
$N \gg d$

Identifying Genes Relevant to a Disease Using Microarrray Gene Expression Data

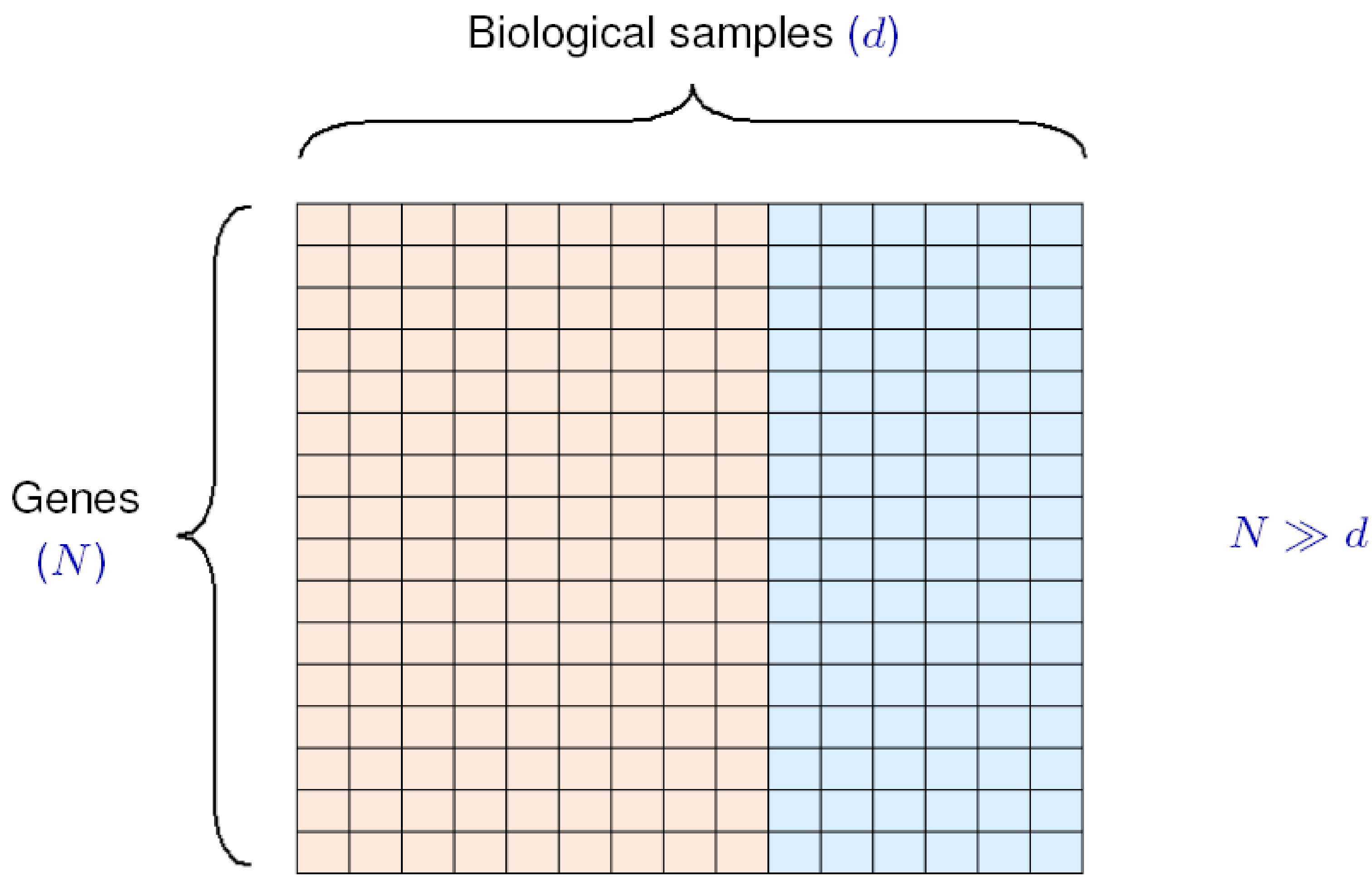


$N \gg d$

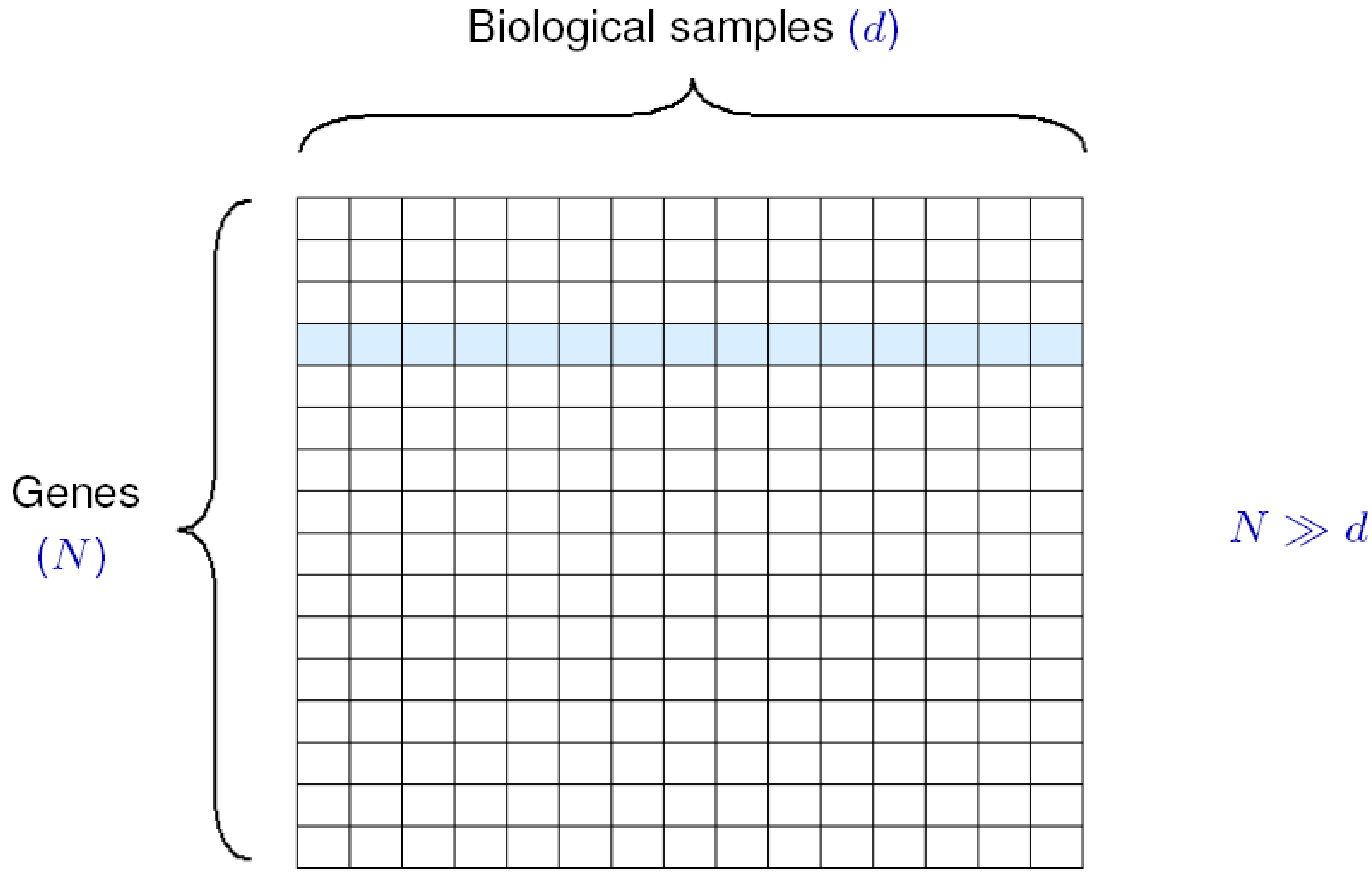
Identifying Genes Relevant to a Disease Using Microarrray Gene Expression Data



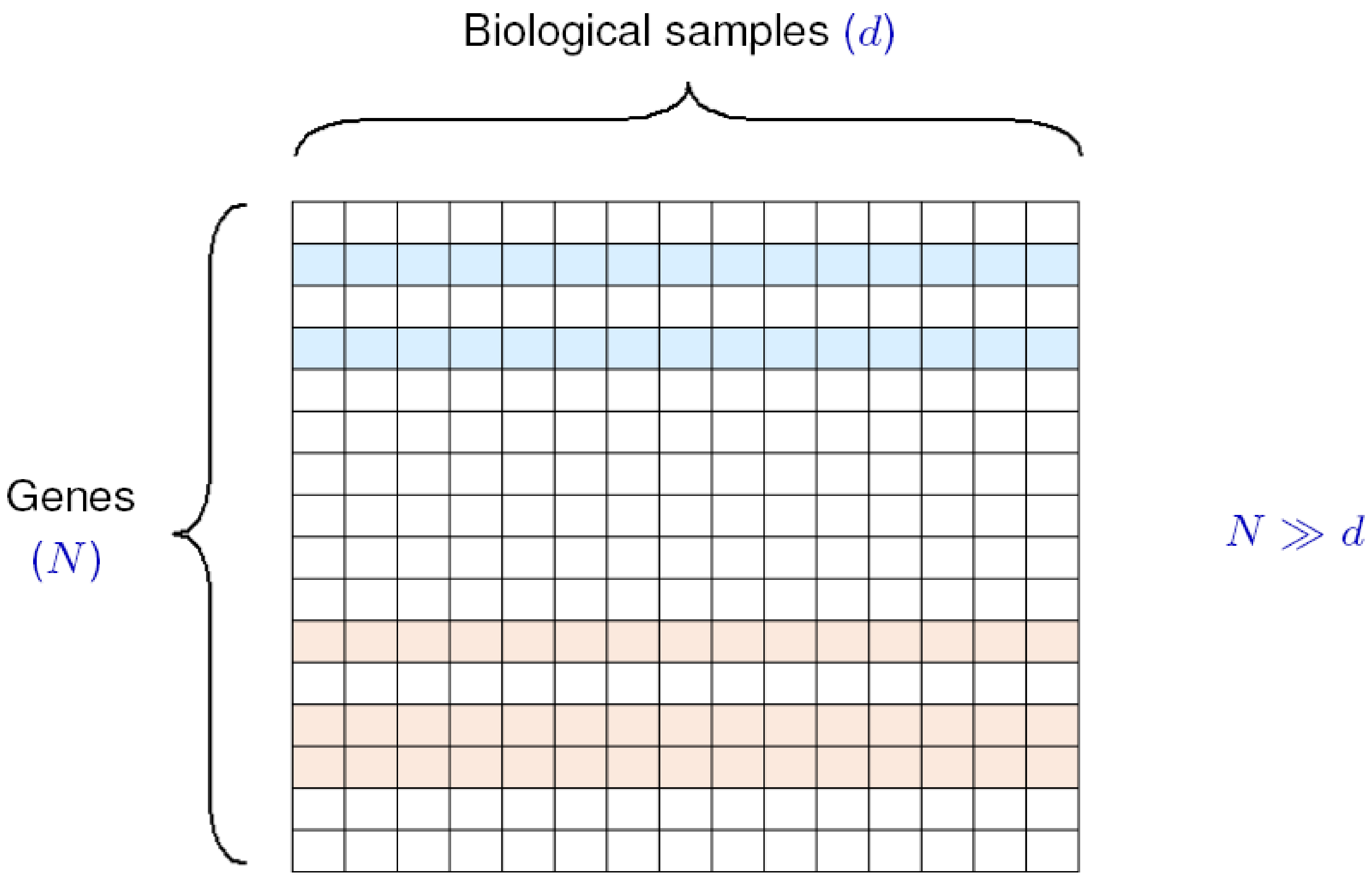
Identifying Genes Relevant to a Disease Using Microarrray Gene Expression Data



Identifying Genes Relevant to a Disease Using Microarrray Gene Expression Data

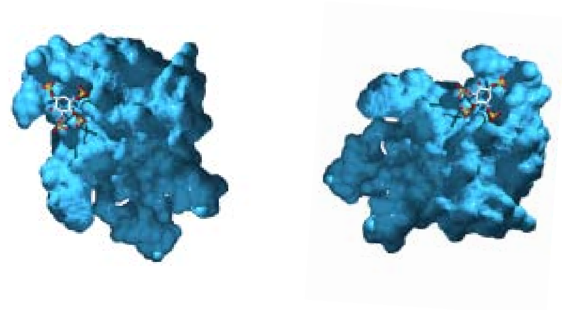


Identifying Genes Relevant to a Disease Using Microarrray Gene Expression Data



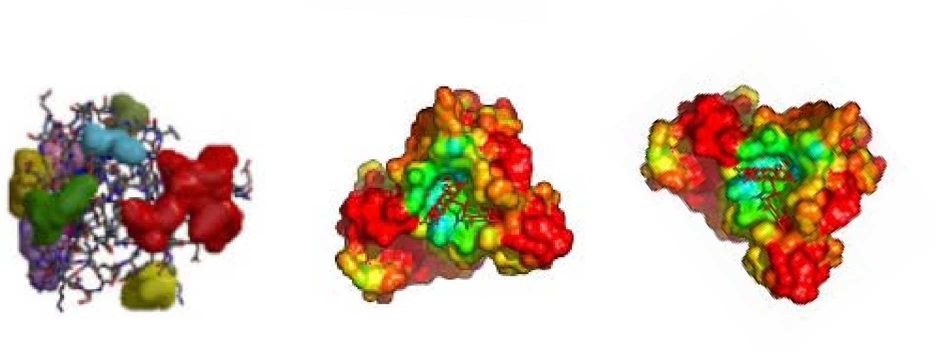
Formulation as a Bipartite Ranking Problem

Relevant



...

Not relevant

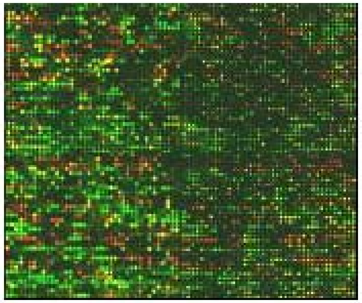


...

Microarray Gene Expression Data Sets

[Golub et al, 1999; Alon et al, 1999]

Data Set	No. of Genes	No. of Tissue Samples	Notes
Leukemia	7129	72	25 AML / 47 ALL
Colon cancer	2000	62	40 tumor / 22 normal



Selection of Training Genes

Leukemia	Colon cancer
Positive genes: Markers for AML/ALL	**Positive genes: Markers for colon cancer**
Myeloperoxidase CD13 CD33 HOXA9 Homeo box A9 V-myb CD19 CD10 (CALLA) TCL1 (T cell leukemia) C-myb Deoxyhypusine synthase	Phospholipase A2 Keratin 6 isoform PTP-H1 TF-IIIA V-raf oncogene MAPK kinase 1 CEA Oncoprotein 18 PEP carboxykinase ERK kinase 1
Negative genes	**Negative genes**
157 genes involved in physiological cellular functions	56 genes involved in physiological cellular functions

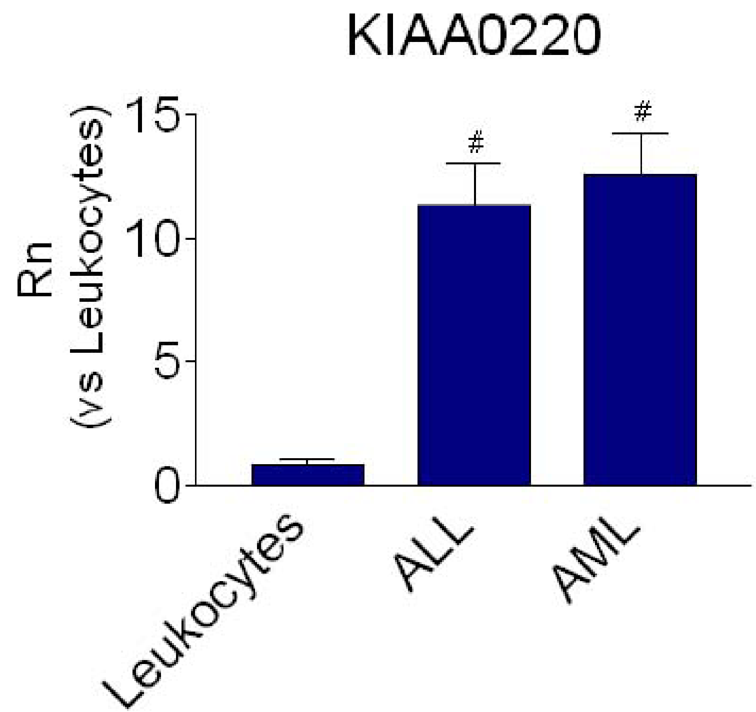
Top-Ranking Genes for Leukemia Returned by RankBoost

♦ Known marker; 🔹 Potential marker;
🟥 Known therapeutic target; 🟦 Potential therapeutic target;
x No link found.

	Gene	Relevance Summary	t-Statistic Rank	Pearson Rank
1.	KIAA0220	🟦	6628	2461
2.	G-gamma globin	♦	3578	3567
3.	Delta-globin	♦	3663	3532
4.	Brain-expressed HHCPA78 homolog	🟦	6734	2390
5.	Myeloperoxidase	♦	139	6573
6.	Disulfide isomerase precursor	🟦	6650	575
7.	Nucleophosmin	♦	405	1115
8.	CD34	♦	6732	643
9.	Elongation factor-1β	x	4460	3413
10.	CD24	♦	81	1
11.	60S ribosomal protein L23	🟦	1950	73
12.	5-aminolevulinic acid synthase	🟦	4750	3351

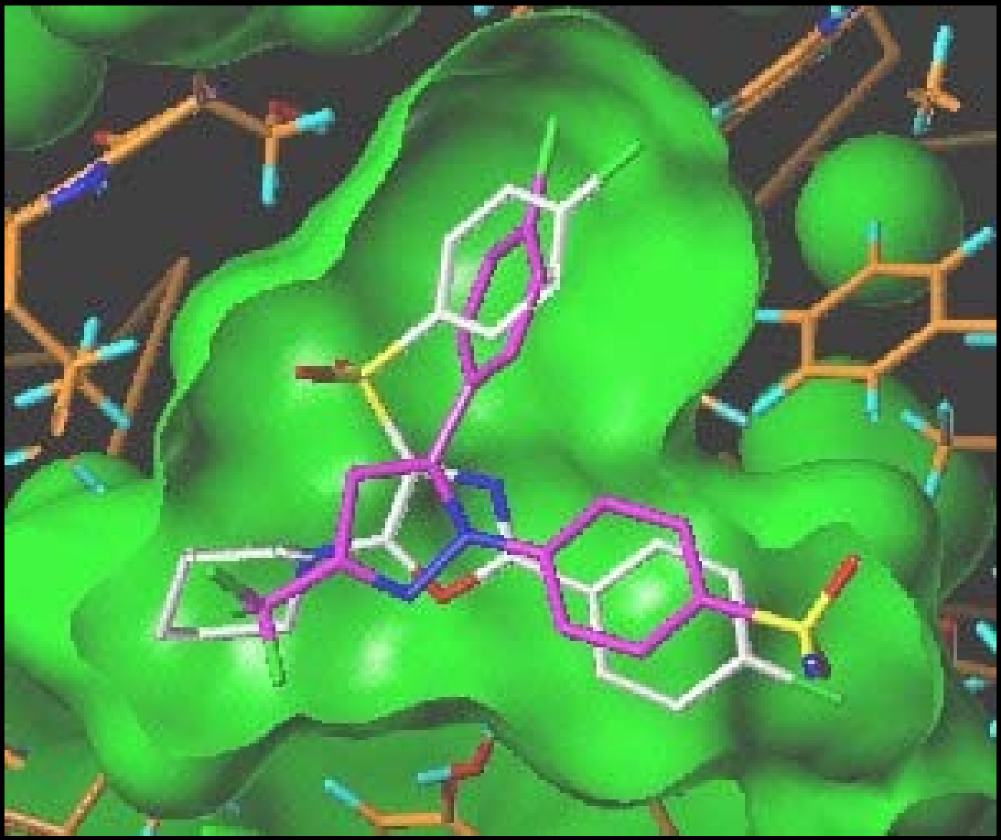
[Agarwal & Sengupta, 2009]

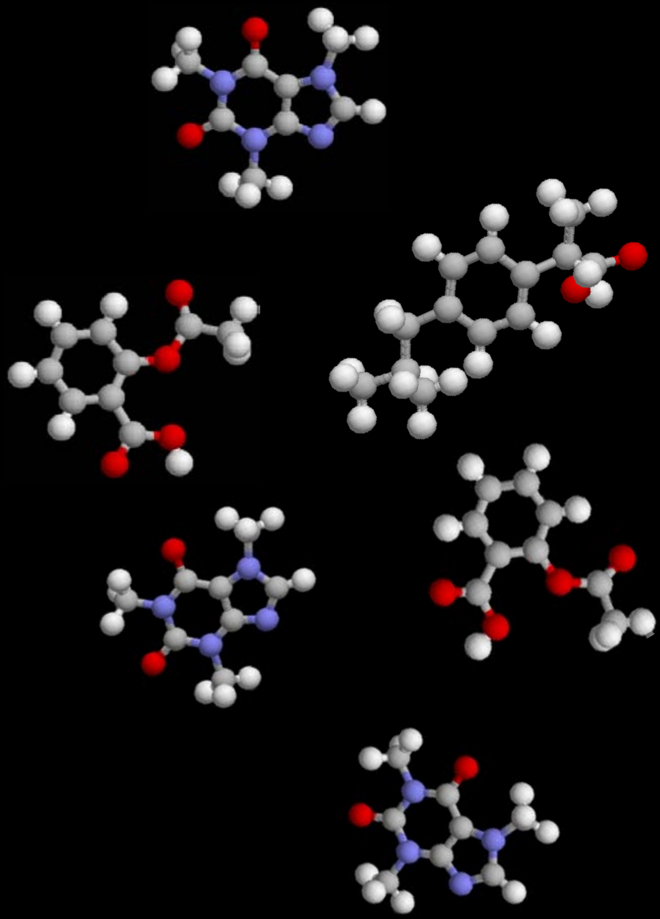
Biological Validation



[Agarwal et al, 2010]

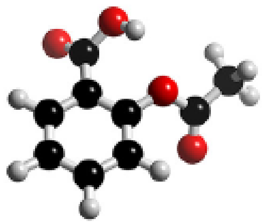
Application to Drug Discovery



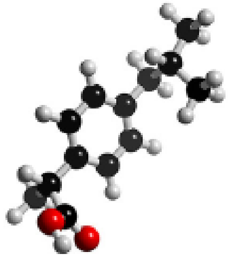


Problem: Millions of structures in a chemical library. How do we identify the most promising ones?

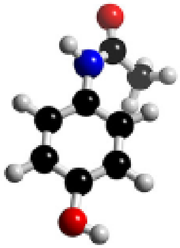
Formulation as a Ranking Problem with Real-Valued Labels



$\mathrm{pIC}_{50} = 5.6718$



$\mathrm{pIC}_{50} = 8.2991$



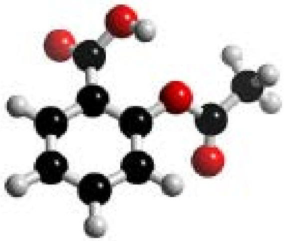
$\mathrm{pIC}_{50} = 4.1317$

...

Cheminformatics Data Sets

[Sutherland et al, 2004]

Data Set	No. of Compounds	No. of Chemical (2.5D) Descriptors	pIC_{50} Values
DHFR inhibitors	361	70	3.3 – 9.8
COX2 inhibitors	292	74	4.0 – 9.0



DHFR Results Using RankSVM

2.5D chemical descriptors
Gaussian kernel

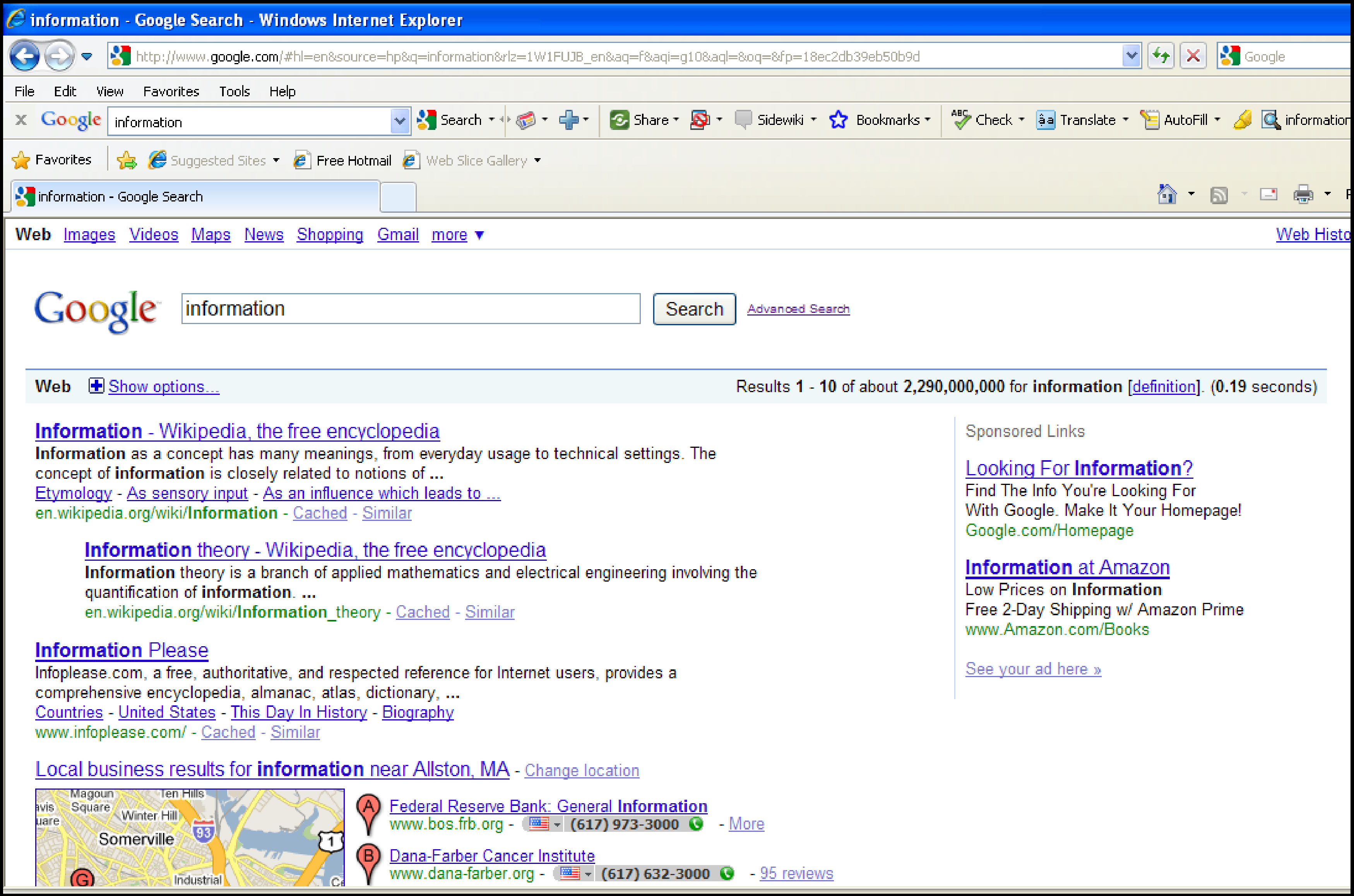
Training size	Ranking error	
	SVR	RankSVM
24	0.4755	**0.4601**
48	**0.3430**	0.3509
72	0.2840	**0.2726**
96	0.2483	**0.2351**
120	0.2171	**0.2121**
144	**0.2023**	0.2032
168	0.2019	**0.1817**
192	0.1808	**0.1749**
216	0.1816	**0.1722**
237	0.1714	**0.1681**

FP2 molecular fingerprints
Tanimoto kernel

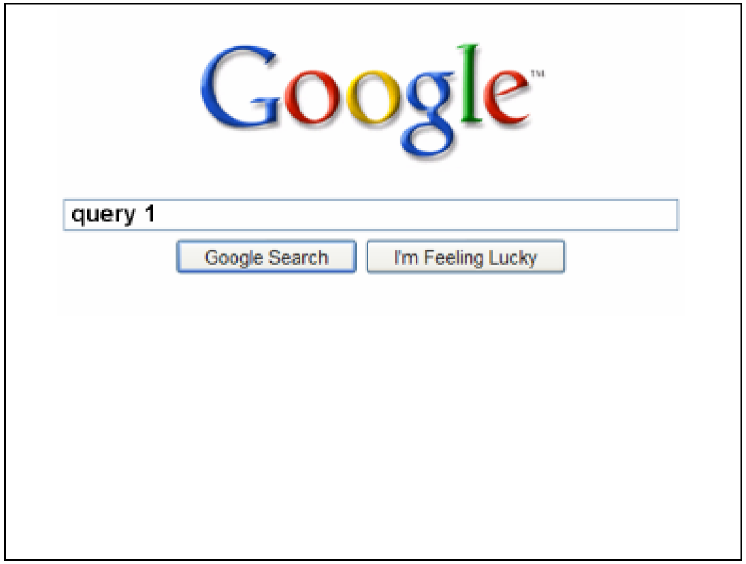
Training size	Ranking error	
	SVR	RankSVM
24	0.3793	**0.3546**
48	0.2905	**0.2896**
72	0.2517	**0.2421**
96	0.2343	**0.2201**
120	0.2147	**0.2052**
144	0.2166	**0.1988**
168	0.2096	**0.1966**
192	0.2056	**0.1962**
216	0.1907	**0.1787**
237	0.1924	**0.1798**

[Agarwal et al, 2010]

Application to Information Retrieval (IR)

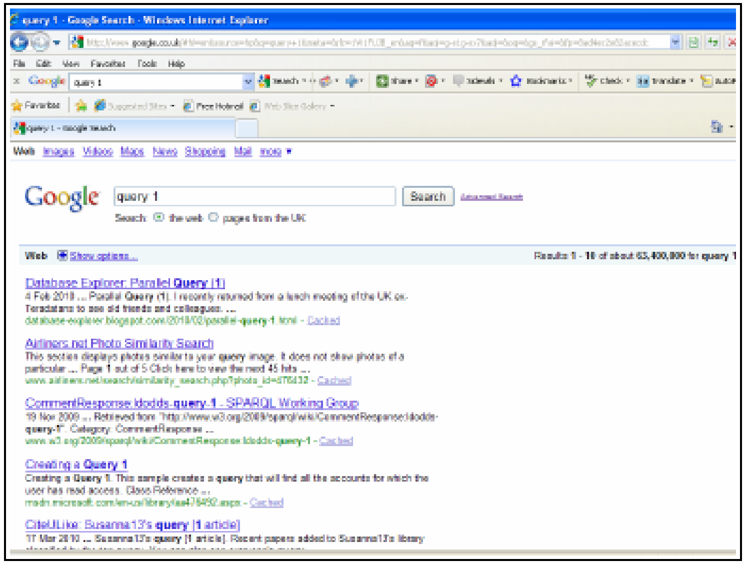


Learning to Rank in IR



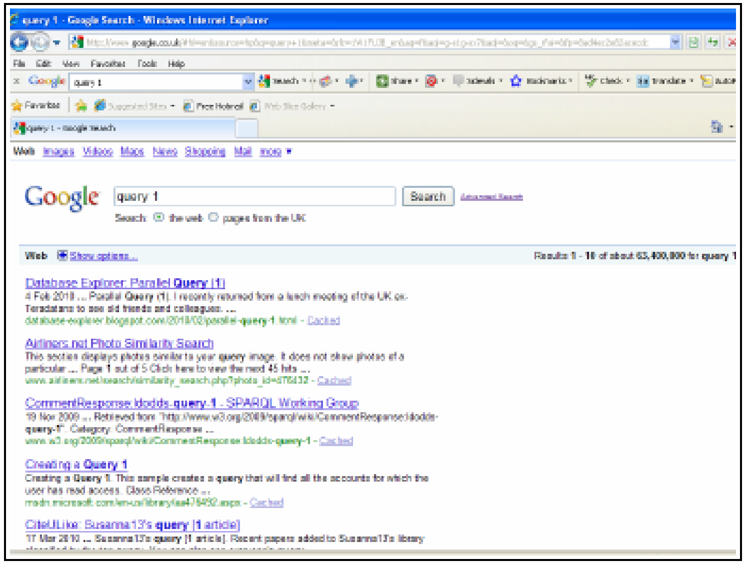
q1

Learning to Rank in IR



q1

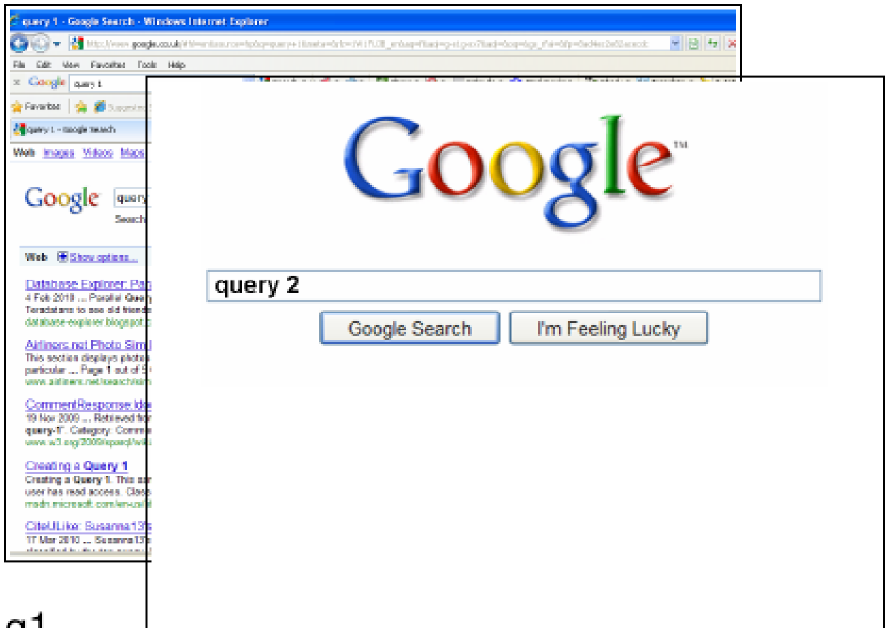
Learning to Rank in IR



q1

rel1

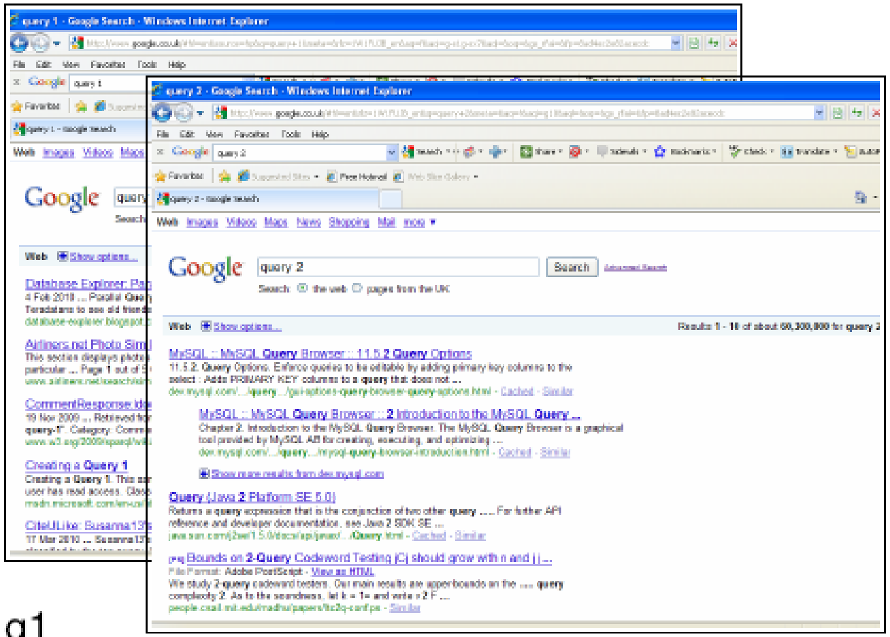
Learning to Rank in IR



q1

rel1 q2

Learning to Rank in IR

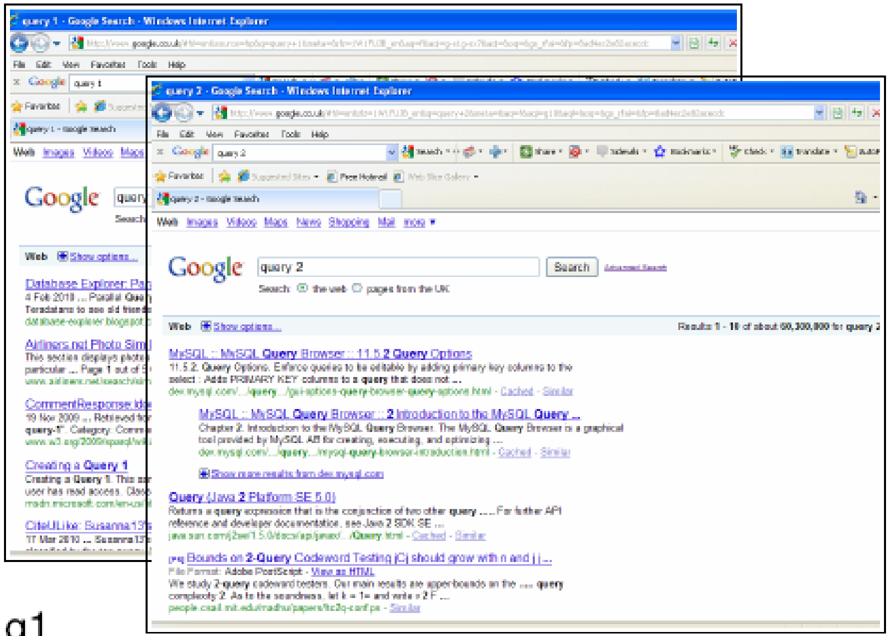


q1

rel1

q2

Learning to Rank in IR



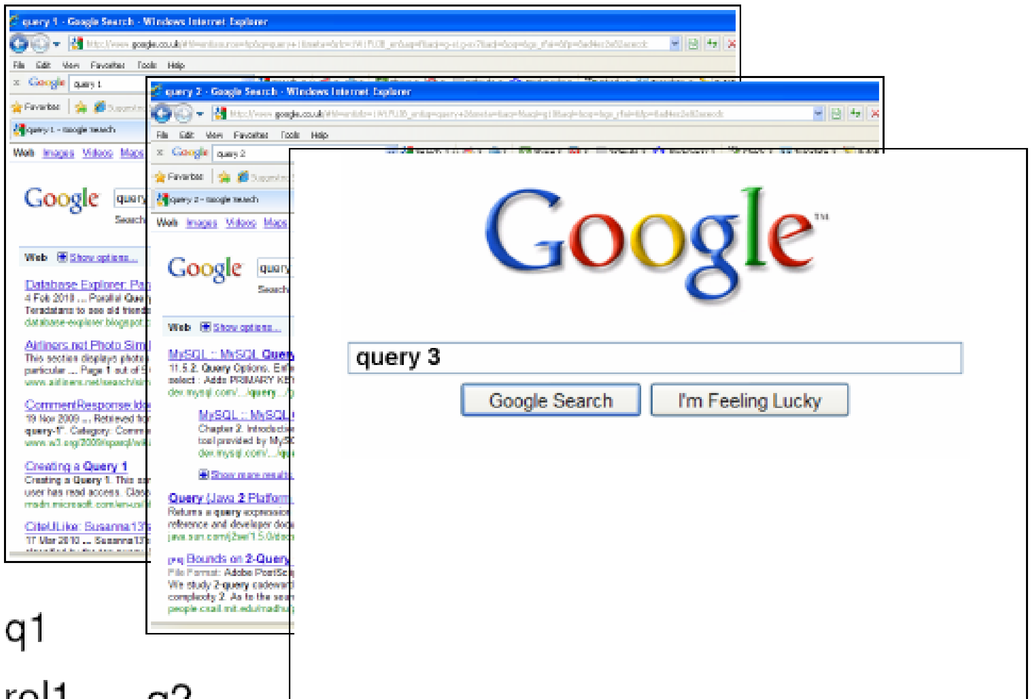
q1

rel1

q2

rel2

Learning to Rank in IR

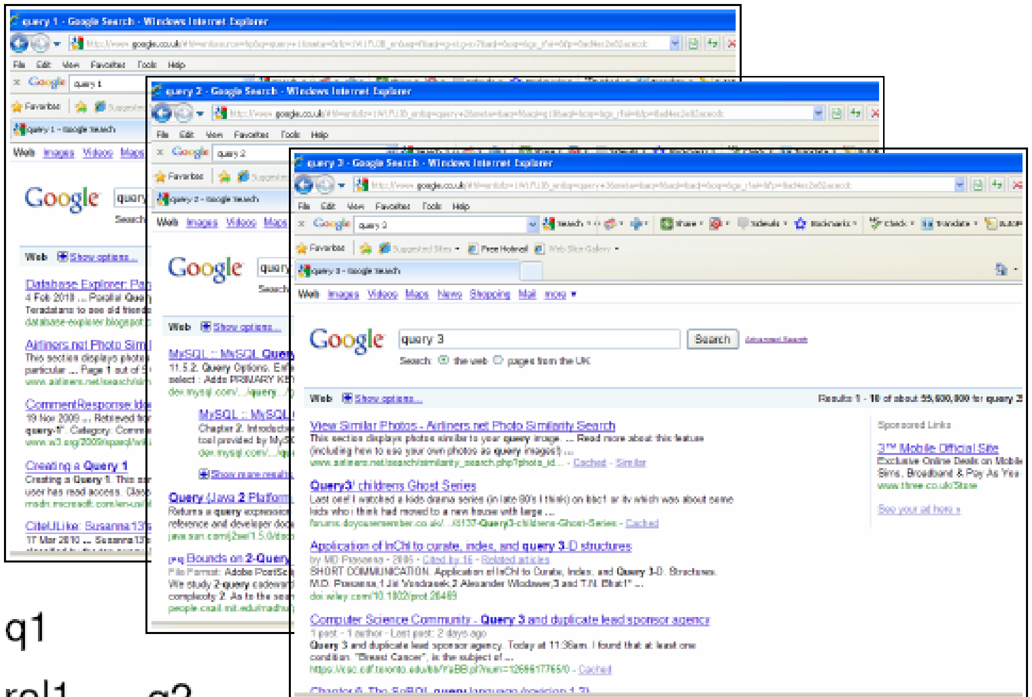


q1

rel1 q2

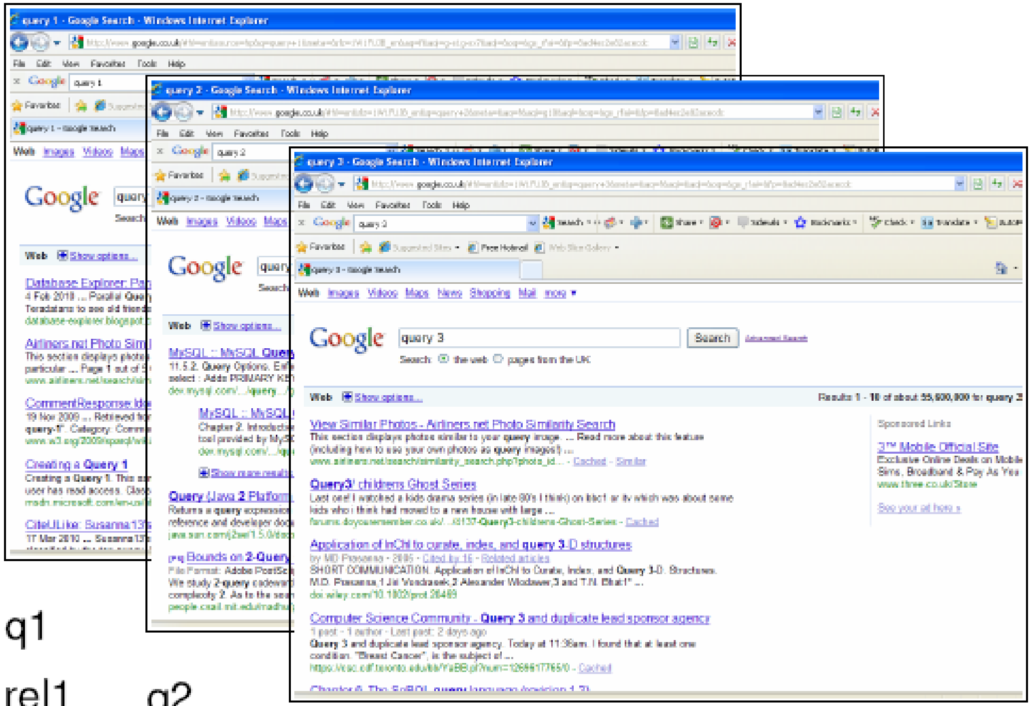
rel2 q3

Learning to Rank in IR



q1 rel1 q2 rel2 q3

Learning to Rank in IR



q1

rel1

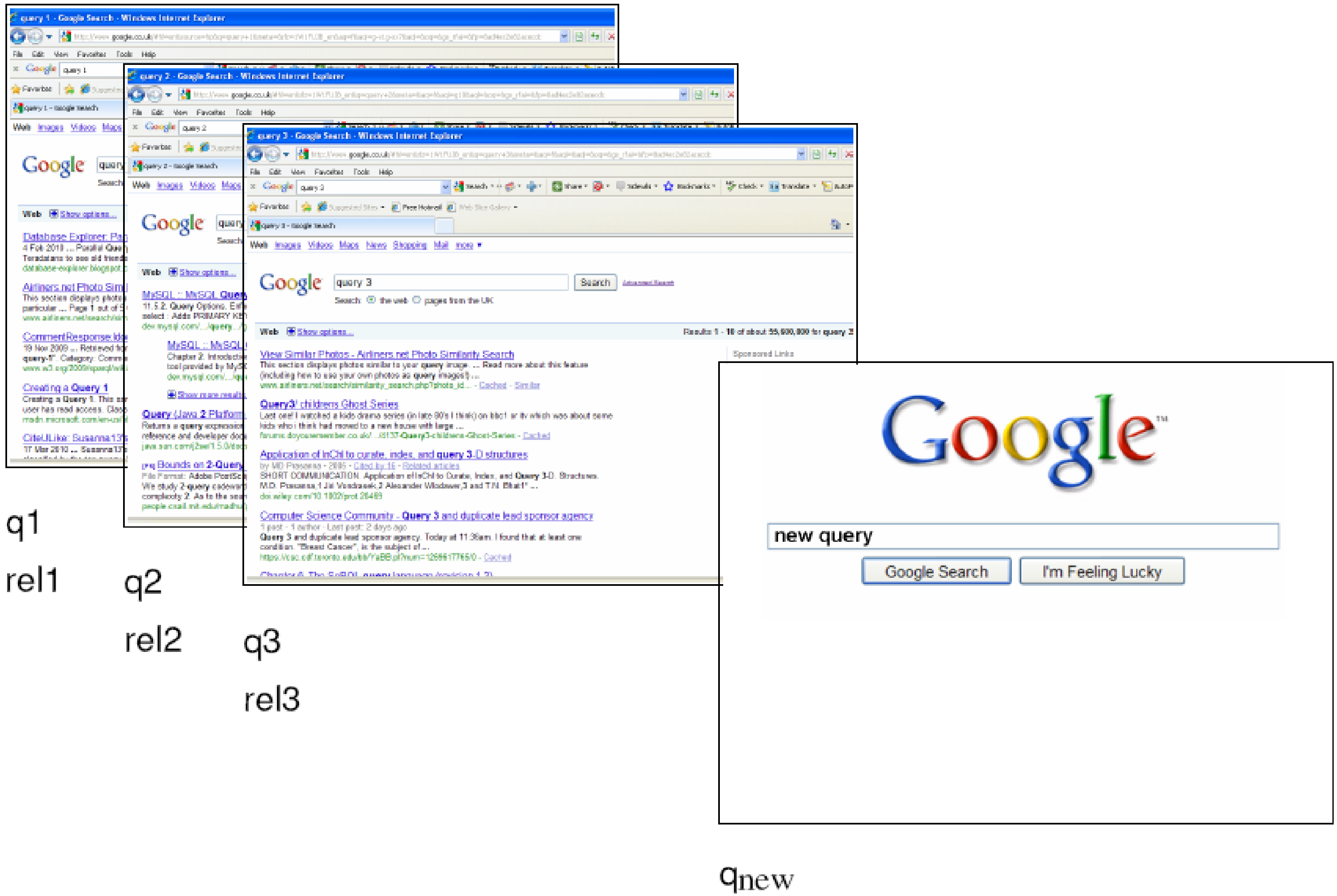
q2

rel2

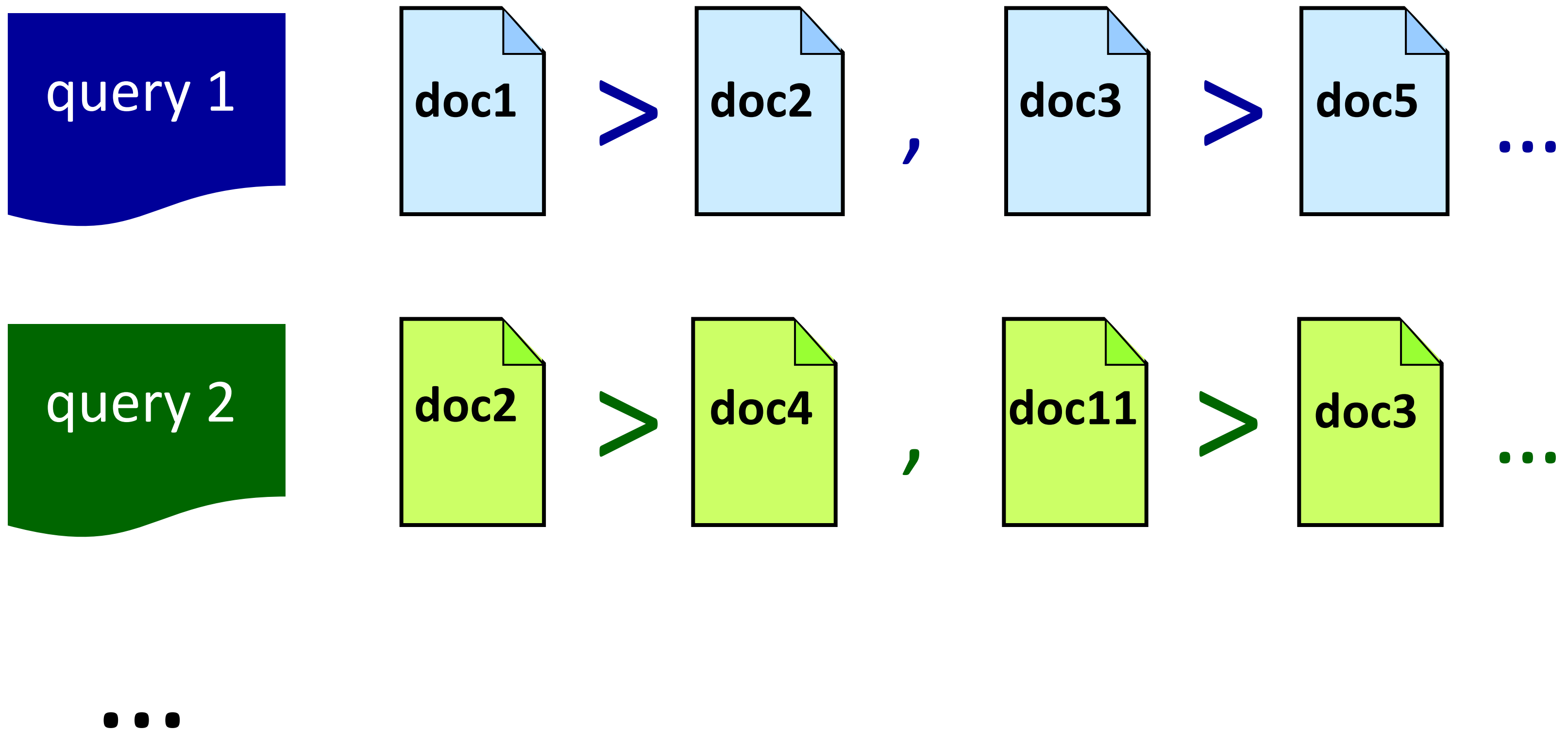
q3

rel3

Learning to Rank in IR



General Subset Ranking



General Subset Ranking

- Query space Q
- Document space D
- Query-document feature mapping $\phi : Q \times D \to \mathbb{R}^d$
- Input: Training sample $S = (S^1, \ldots, S^m)$:

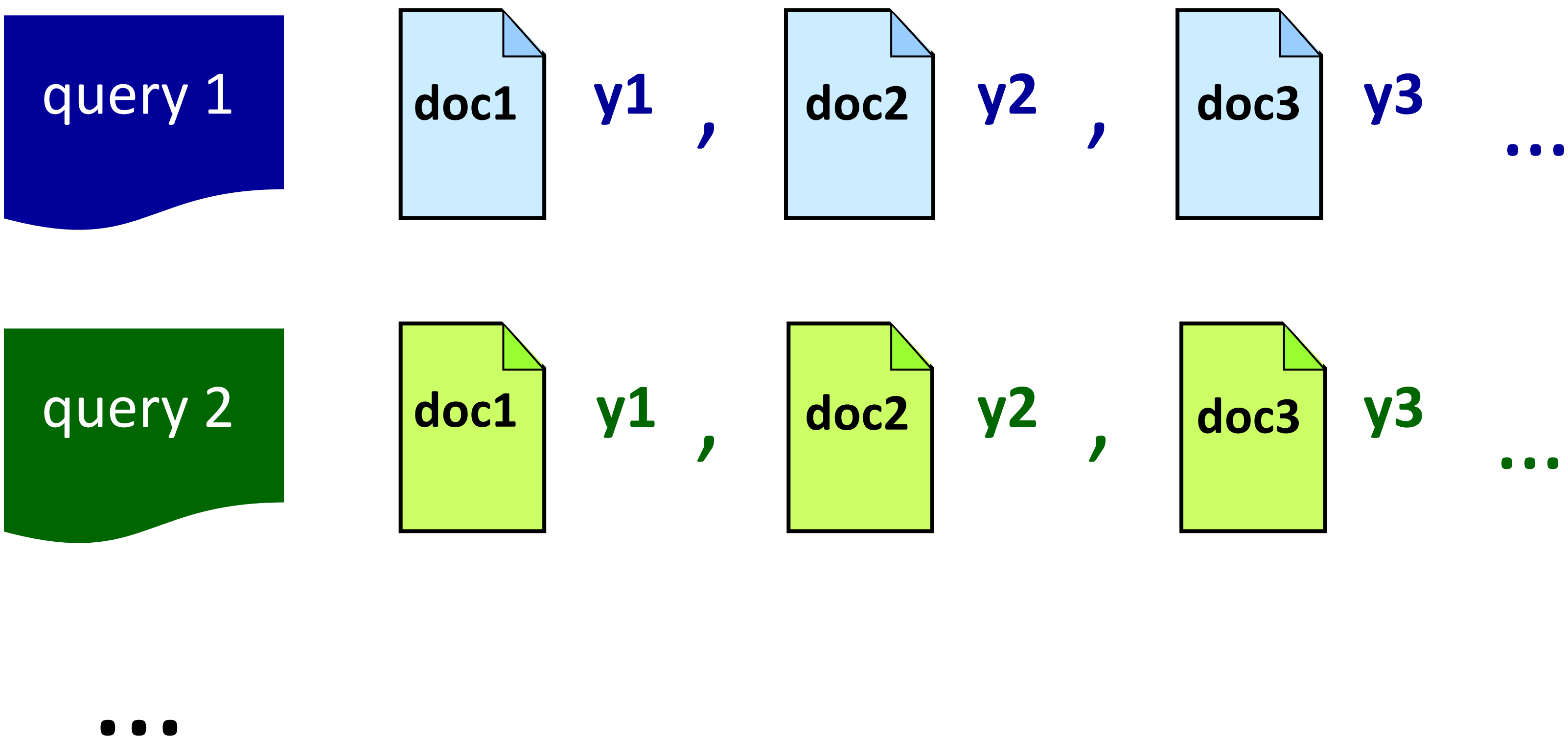
  $$S^i = \Big((\phi_1^i, {\phi_1^i}'), \ldots, (\phi_{n_i}^i, {\phi_{n_i}^i}') \Big) \in (\mathbb{R}^d \times \mathbb{R}^d)^{n_i}$$

  where

  $$\phi_j^i = \phi(q^i, d_j^i), \quad {\phi_j^i}' = \phi(q^i, {d_j^i}')$$

- Output: Ranking function $f : \mathbb{R}^d \to \mathbb{R}$

Subset Ranking with Real-Valued Relevance Labels



Subset Ranking with Real-Valued Relevance Labels

- Query space Q
- Document space D
- Query-document feature mapping $\phi : Q \times D \to \mathbb{R}^d$
- Input: Training sample $S = (S^1, \ldots, S^m)$:

  $$S^i = \Big((\phi_1^i, y_1^i), \ldots, (\phi_{n_i}^i, y_{n_i}^i) \Big) \in (\mathbb{R}^d \times \mathbb{R})^{n_i}$$

  where

  $$\phi_j^i = \phi(q^i, d_j^i), \quad y_j^i = \text{relevance of } d_j^i \text{ to } q^i$$

- Output: Ranking function $f : \mathbb{R}^d \to \mathbb{R}$

RankSVM Applied to IR/Subset Ranking

Standard RankSVM

$$\min_{f \in \mathcal{F}_K} \left[\left(\frac{1}{\sum_{i=1}^{m} \binom{n_i}{2}} \right) \sum_{i=1}^{m} \sum_{1 \le j < k \le n_i} \ell_{\text{hinge}} \left(f, (\phi_j^i, y_j^i), (\phi_k^i, y_k^i) \right) + \frac{\lambda}{2} \|f\|_K^2 \right]$$

$$\ell_{\text{hinge}} \left(f, (\phi_j^i, y_j^i), (\phi_k^i, y_k^i) \right) = \left(1 - \left(\text{sign}(y_j^i - y_k^i) \cdot (f(\phi_j^i) - f(\phi_k^i)) \right) \right)_+ ,$$

convex upper bound on

$$\mathbf{1} \left((y_j^i - y_k^i)(f(\phi_j^i) - f(\phi_k^i)) < 0 \right)$$

[Joachims, 2002]

RankSVM Applied to IR/Subset Ranking

RankSVM with Query Normalization & Relevance Weighting

$$\min_{f \in \mathcal{F}_K} \left[\frac{1}{m} \sum_{i=1}^{m} \left[\frac{1}{\binom{n_i}{2}} \sum_{1 \leq j < k \leq n_i} \ell^{\text{rel}}_{\text{hinge}} \left(f, (\phi^i_j, y^i_j), (\phi^i_k, y^i_k) \right) + \frac{\lambda}{2} \|f\|^2_K \right] \right]$$

$$\ell^{\text{rel}}_{\text{hinge}} \left(f, (\phi^i_j, y^i_j), (\phi^i_k, y^i_k) \right) = \left(|y^i_j - y^i_k| - \left(\text{sign}(y^i_j - y^i_k) \cdot (f(\phi^i_j) - f(\phi^i_k)) \right) \right)_+ ,$$

convex upper bound on

$$|y^i_j - y^i_k| \, \mathbf{1} \left((y^i_j - y^i_k)(f(\phi^i_j) - f(\phi^i_k)) < 0 \right)$$

[Agarwal & Collins, 2010; also Cao et al, 2006]

Ranking Performance Measures in IR

Mean Average Precision (MAP)

Binary Labels: $y_j \in \{0, 1\}$

$$\text{MAP}_S(f) = \frac{1}{m}\sum_{i=1}^{m}\left[\frac{1}{\left|\{j : y_j^i = 1\}\right|}\sum_{j:y_j^i=1}\text{prec}^i_{r_j^i}(f)\right]$$

$$r_j^i = \text{rank of document } d_j^i \text{ for query } q^i$$

$$\text{prec}^i_r(f) = \text{fraction of positives in top } r \text{ documents for query } q^i$$

Ranking Performance Measures in IR

Normalized Discounted Cumulative Gain (NDCG)

General Real-Valued Labels: $y_j \in \mathbb{R}$

$$\mathrm{NDCG}_S(f) = \frac{1}{m}\sum_{i=1}^{m}\left[\frac{1}{Z_i}\sum_{r=1}^{n_i}\frac{2^{y^i_{\pi^i_r}}-1}{\log_2(r+1)}\right]$$

$$\pi^i_r = \text{index of document ranked at position } r \text{ for query } q^i$$

$$Z_i = \text{normalization constant}$$

$$\mathrm{NDCG@}k_S(f) = \frac{1}{m}\sum_{i=1}^{m}\left[\frac{1}{Z_i}\sum_{r=1}^{k}\frac{2^{y^i_{\pi^i_r}}-1}{\log_2(r+1)}\right]$$

Ranking Algorithms for Optimizing MAP/NDCG

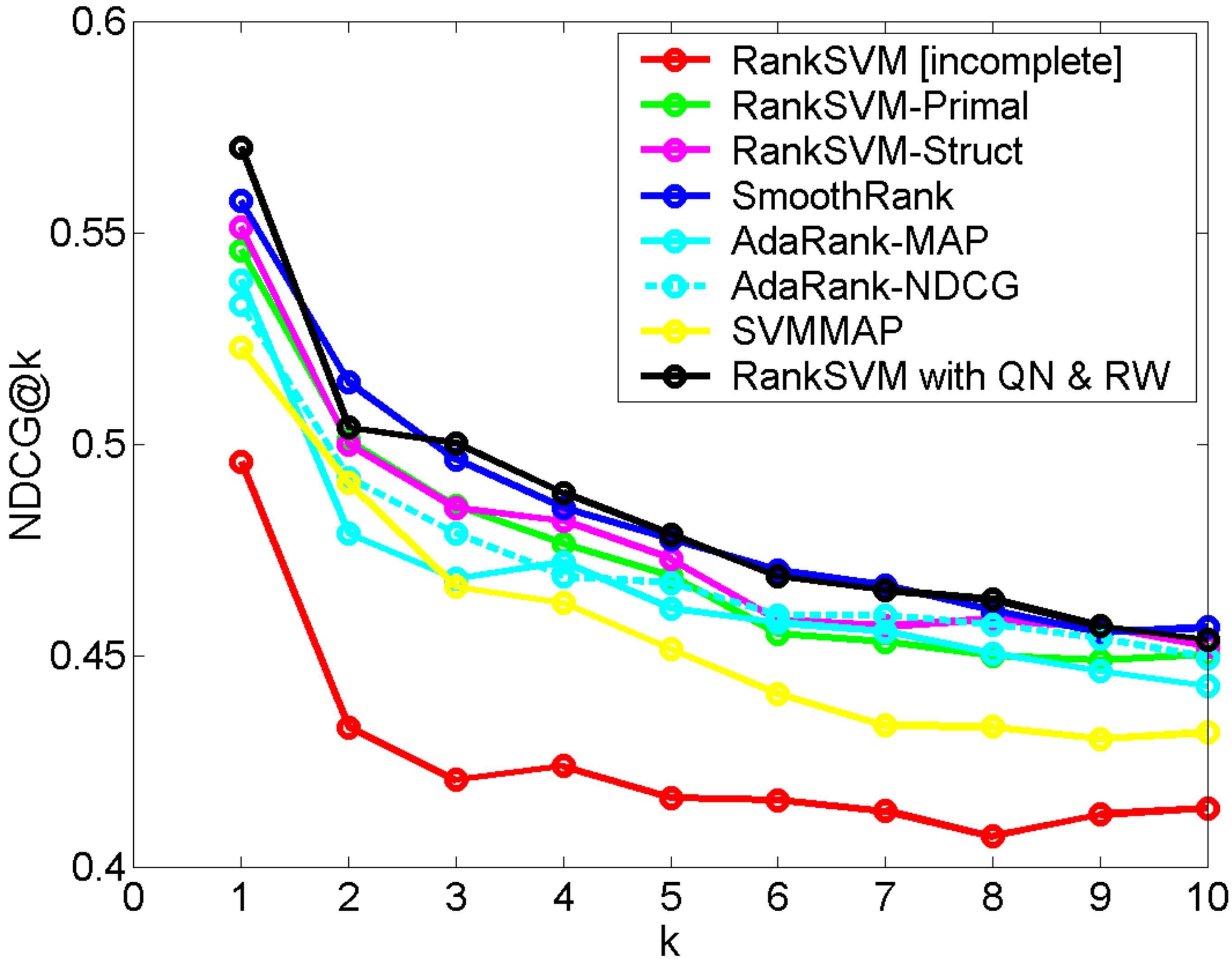
- SVMMAP [Yue et al. 2007]
- SVMNDCG [Chapelle et al. 2007]
- LambdaRank [Burges et al. 2007]
- AdaRank [Xu & Li 2007]
- Regression-based algorithm [Cossock & Zhang 2008]
- SoftRank [Taylor et al. 2008]
- SmoothRank [Chapelle & Wu 2010]

LETOR 3.0/OHSUMED Data Set

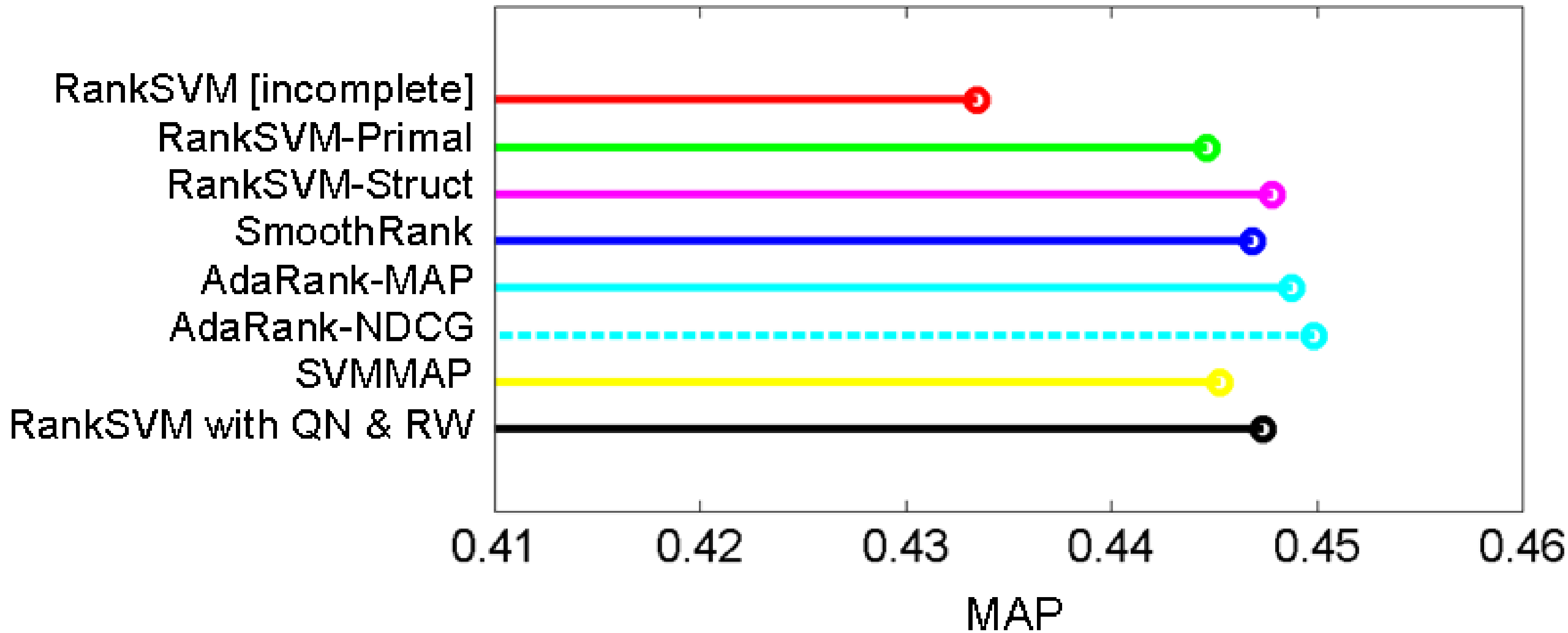
[Liu et al, 2007]

No. of Queries	Relevance Labels	Total no. of Query-Doc Pairs	Avg. no. of Docs/Query	No. of Features
106	2 : definitely relevant 1 : partially relevant 0 : not relevant	16, 140	152	45

OHSUMED Results – NDCG



OHSUMED Results – MAP



Further Reading & Resources

[Incomplete!]

Early Papers on Ranking

W. W. Cohen, R. E. Schapire, and Y. Singer, Learning to order things, *Journal of Artificial Intelligence Research*, 10:243–270, 1999.

R. Herbrich, T. Graepel, and K. Obermayer, Large margin rank boundaries for ordinal regression. *Advances in Large Margin Classifiers*, 2000.

T. Joachims, Optimizing search engines using clickthrough data, KDD 2002.

Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer, An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research*, 4:933–969, 2003.

C.J.C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, G. Hullender, Learning to rank using gradient descent, ICML 2005.

Generalization Bounds for Ranking

S. Agarwal, T. Graepel, R. Herbrich, S. Har-Peled, D. Roth, Generalization bounds for the area under the ROC curve, *Journal of Machine Learning Research*, 6:393—425, 2005.

S. Agarwal, P. Niyogi, Generalization bounds for ranking algorithms via algorithmic stability, *Journal of Machine Learning Research*, 10:441—474, 2009.

C. Rudin, R. Schapire, Margin-based ranking and an equivalence between AdaBoost and RankBoost, *Journal of Machine Learning Research,* 10: 2193—2232, 2009

Bioinformatics/Drug Discovery Applications

S. Agarwal and S. Sengupta, Ranking genes by relevance to a disease, CSB 2009.

S. Agarwal, D. Dugar, and S. Sengupta, Ranking chemical structures for drug discovery: A new machine learning approach. *Journal of Chemical Information and Modeling*, DOI 10.1021/ci9003865, 2010.

Other Applications

Natural Language Processing

M. Collins and T. Koo, Discriminative reranking for natural language parsing, *Computational Linguistics,* 31:25—69, 2005.

Collaborative Filtering

M. Weimer, A. Karatzoglou, Q. V. Le, and A. Smola, CofiRank - Maximum margin matrix factorization for collaborative ranking, NIPS 2007.

Manhole Event Prediction

C. Rudin, R. Passonneau, A. Radeva, H. Dutta, S. Ierome, and D. Isaac , A process for predicting manhole events in Manhattan, *Machine Learning*, DOI 10.1007/s10994-009-5166, 2010.

IR Ranking Algorithms

Y. Cao, J. Xu, T.-Y. Liu, H. Li, Y. Hunag, and H.W. Hon, Adapting ranking SVM to document retrieval, SIGIR 2006.

C.J.C. Burges, R. Ragno, and Q.V. Le, Learning to rank with non-smooth cost functions. NIPS 2006.

J. Xu and H. Li, AdaRank: A boosting algorithm for information retrieval. SIGIR 2007.

Y. Yue, T. Finley, F. Radlinski, and T. Joachims, A support vector method for optimizing average precision. SIGIR 2007.

M. Taylor, J. Guiver, S. Robertson, T. Minka, Softrank: optimizing non-smooth rank metrics. WSDM 2008.

IR Ranking Algorithms

S. Chakrabarti, R. Khanna, U. Sawant, and C. Bhattacharyya, Structured learning for nonsmooth ranking losses. KDD 2008.

D. Cossock and T. Zhang, Statistical analysis of Bayes optimal subset ranking, *IEEE Transactions on Information Theory,* 54:5140–5154, 2008.

T. Qin, X.D. Zhang, M.F. Tsai, D.S. Wang, T.Y. Liu, and H. Li. Query-level loss functions for information retrieval. *Information Processing and Management,* 44:838–855, 2008.

O. Chapelle and M. Wu, Gradient descent optimization of smoothed information retrieval metrics. *Information Retrieval* (To appear), 2010.

S. Agarwal and M. Collins, Maximum margin ranking algorithms for information retrieval, ECIR 2010.

NIPS Workshop 2005
Learning to Rank

SIGIR Workshops 2007-2009
Learning to Rank for Information Retrieval

NIPS Workshop 2009
Advances in Ranking

American Institute of Mathematics
Workshop in Summer 2010
The Mathematics of Ranking

Tutorial Articles & Books

Tie-Yan Liu, Learning to Rank for Information Retrieval, Foundations & Trends in Information Retrieval, 2009.

Shivani Agarwal, A Tutorial Introduction to Ranking Methods in Machine Learning, In preparation.

Shivani Agarwal (Ed.), Advances in Ranking Methods in Machine Learning, Springer-Verlag, In preparation.