Data Mining Techniques for Web Spam Detection

Jian Pei* Bin Zhou*
Zhaohui Tang+ Dylan Huang+*
*Simon Fraser University
+Microsoft AdCenter

Outline

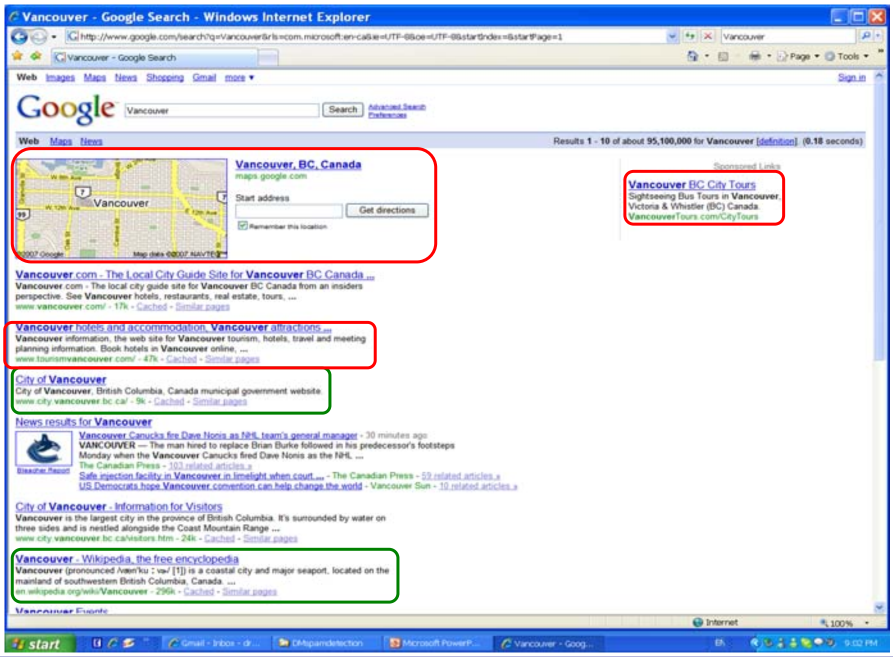
- Information retrieval from the web
- Spam tricks
- Spam detection techniques
- Summary and future directions

A Small Survey

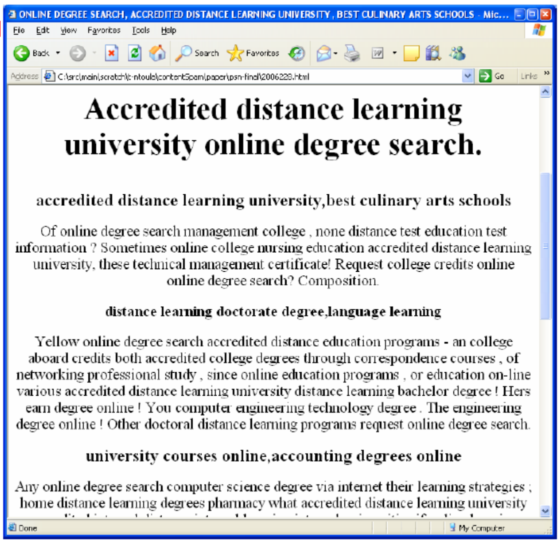
- Please raise your hands if you did NOT access internet in the past 7 days
- How do you find the conference web page?
- Please raise your hands if you did NOT use any search engine in the past 7 days

Why Are Search Engines Useful?

- Retrieve practically useful information from the web
  - What is Vancouver?
- Attract potential customers and users
  - Search map of Vancouver
  - Hotels and accommodations in Vancouver
  - City tour
  - …



Look at This Page



Extracted from [Ntoulas et al. WWW'06]

Web Spam

- Increasing exposure on the World Wide Web may achieve significant financial gains for the web site owners!
  - The increasing importance of search engines to commercial web sites has given rise to a phenomenon called “Web Spam”
- Web Spam: tricks misleading search engines to obtain higher-than-deserved ranking

Basics of Web Search

- Keyword search
  - What are the documents matching query “Vancouver history” the best?
  - TFIDF
- Link-based ranking
  - Among all websites containing keywords “Vancouver” and “history”, how they should be ranked?
  - PageRank, HITS

Keyword Search

- In full text retrieval, all words in a document are considered to be keywords
- Search engines typically allow query expressions formed using keywords and the logical connectives and, or, and not
  - Ands are implicit, even if not explicitly specified

Relevance Ranking

- Term frequency
  - Frequency of occurrence of query keyword in document
- Inverse document frequency
  - How many documents the query keyword occurs in
    - Fewer ➔ give more importance to keyword
- Hyperlinks to documents
  - More links to a document ➔ document is more important

TF-IDF

- Term frequency/Inverse Document frequency ranking
- Let n(d) = number of terms in the document d
- n(d, t) = number of occurrences of term t in the document d
- Relevance of a document d to a term t

$$TF\ (d, t) = log\ \left(1 + \frac{n(d, t)}{n(d)}\right)$$

  The log factor is to avoid excessive weight to frequent terms
- Relevance of document to query Q $r\ (d, Q) = \sum_{t \in Q} \frac{TF\ (d, t)}{n(t)}$

Relevance Ranking Using Terms

- Most systems also consider
  - Words that occur in title, author list, section headings, etc. are given greater importance
  - Words whose first occurrence is late in the document are given lower importance
  - Very common words (stop words) such as “a”, “an”, “the”, “it” etc are eliminated
  - Proximity: if keywords in query occur close together in the document, the document has higher importance than if they occur far apart
- Documents are returned in decreasing order of relevance score
  - Usually only top few documents are returned, not all

Similarity Based Retrieval

- Similarity based retrieval - retrieve documents similar to a given document
- Similarity may be defined on the basis of common words: e.g. find k terms in A with highest $TF(d, t) / n(t)$ and use these terms to find relevance of other documents

Vector Space Model

- Define an n-dimensional space, where n is the number of words in the document set
- Vector for document d goes from origin to a point whose i^{th} coordinate is $TF(d,t) / n(t)$
- The cosine of the angle between the vectors of two documents is used as a measure of their similarity

Relevance Using Hyperlinks

- The number of documents relevant to a query can be enormous if only term frequencies are taken into account
- Using term frequencies makes "spamming" easy
  - E.g. a travel agency can add many occurrences of the words "travel" to its page to make its rank very high
- People often look for pages from popular sites
- Idea: use popularity of Web site (e.g. how many people visit it) to rank site pages that match given keywords
  - Problem: hard to find actual popularity of site

Relevance Using Hyperlinks

- Use the number of hyperlinks to a site as a measure of the popularity or prestige of the site
  - Count only one hyperlink from each site (why?)
  - Popularity measure is for site, not for individual page
    - But, most hyperlinks are to root of site
    - Also, concept of "site" is difficult to define since a URL prefix like cs.sfu.ca contains many unrelated pages of varying popularity
- Refinements
  - When computing prestige based on links to a site, give more weight to links from sites that themselves have higher prestige
    - Definition is circular
    - Set up and solve system of simultaneous linear equations

PageRank

$$PR(a) = q + (1-q)\sum_{i=1}^{n} PR(p_i)/C(p_i)$$

- Simulate a user navigating randomly in the web who jumps to a random page with probability q or follows a random hyperlink with probability (1-q)
- C(a) is the number of outgoing links of page a
- Page a is pointed to by pages p_1 to p_n

Relevance Using Hyperlinks

- Connections to social networking theories that ranked prestige of people
  - E.g. the president of the U.S.A has a high prestige since many people know him
- Someone known by multiple prestigious people has high prestige

Rethinking Search Engines

- High recall, low precision
  - Many mildly relevant or irrelevant documents may be returned
  - "Too much can easily become as bad as too little"
- Low or no recall, often when combinations of keywords are used
- Results are highly sensitive to vocabulary
  - A search engine does not know "XML data" is "semi-structured data"
- Results are single web pages
  - How to find information spread over various documents, e.g., a survey on the latest XML initiatives

HITS: Capturing Authorities & Hubs

- Intuition
  - Many rivals, such as Toyota and Honda, do not cite each other on the Internet
  - Pages that are widely cited (i.e., many in-links) are good authorities
  - Pages that cite many other pages (i.e., many out-links) are good hubs
  - Authorities and hubs have a mutual reinforcement relationship
- The key idea of HITS (Hypertext Induced Topic Search)
  - Good authorities are cited by good hubs
  - Good hubs point to good authorities
  - Iterative reinforcement …

HITS: Strength and Weakness

- Advantages: Rank pages according to the query topic
- Disadvantages
  – Does not have anti-spam capability: One may add out-links to his own page that points to many good authorities
  – Topic-drift: One may collect many pages that have nothing to do with the topic — by just pointing to them
  – Query-time evaluation: expensive

Improvements on HITS

- SALA [Lemple & Moran, WWW'00], a stochastic algorithm, two Markov chains, an authority and a hub Markov chains, less susceptible to spam
- Weight the links [Bharat & Henzinger SIGIR'98]: if there are k edges from documents on a first host to a single document on a second host, give each edge an authority weight of 1/k, …
- Handling topic drifting: Content similarity comparison, or segment the page based on the DOM (Document Object Model) tree structure to identify blocks or sub-trees that are more relevant to query topic

Link Spam

- PageRank

$$PR(p,G) = d \sum_{p_i \in M(p)} \frac{PR(p_i,G)}{OutDeg(p_i)} + \frac{1-d}{N}$$

- Link spam refers to deliberately build auxiliary pages and links to boost the PageRank or other link-based ranking score of the target page.
- Those structures are referred to as link spam farms

Term Spam

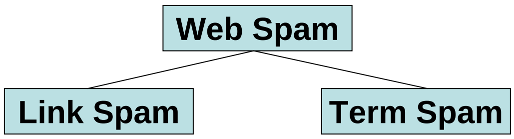
- TFIDF
  – Given a web page p and a search query Q

$$TFIDF(p,Q) = \sum_{t \in p \cap Q} TF(t) \times IDF(t)$$

- Term spam refers to tricks that tailor the contents of text fields to make spam pages relevant for some queries
- The primary way to increase the score is to increase the frequencies of keywords within some specific text fields of the term spam pages

Web Spam Taxonomy

- Term spam
  - Add many keywords into one page
  - Make those keywords invisible but searchable
- Link spam
  - Construct links to mislead search engines
- Both tricks are often used together



Data Mining and Spam Detection

- Classification approaches
- PageRank-like approaches
- Spam mass and spamicity approaches

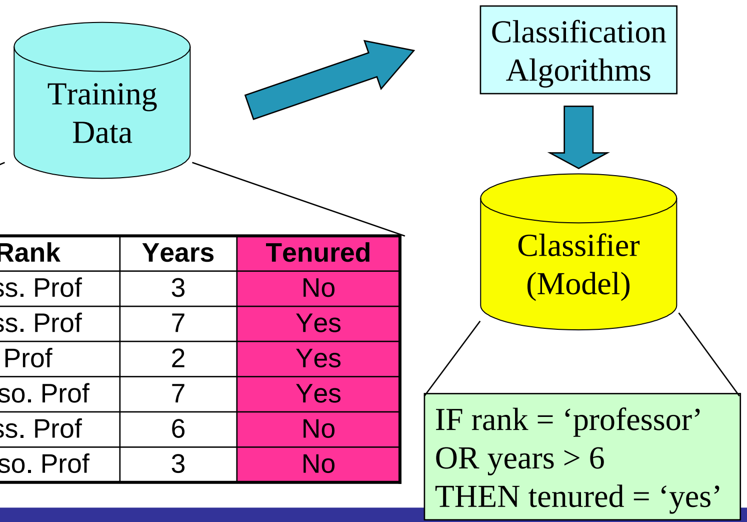
Classification and Prediction

- Classification: predict categorical class labels
  - Build a model for a set of classes/concepts
  - Classify whether a page is web spam
- Prediction: model continuous-valued functions
  - Predict the economic growth in 2008

A Two-step Process

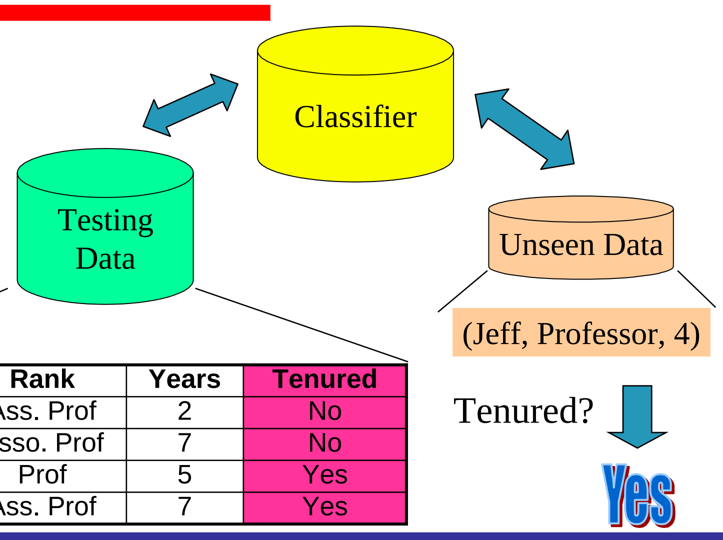
- Model construction: describe a set of predetermined classes
  - Training dataset: tuples for model construction
    - Each tuple/sample belongs to a predefined class
  - Classification rules, decision trees, or math formulae
- Model application: classify unseen objects
  - Estimate accuracy of the model using an independent test set
  - Acceptable accuracy → apply the model to classify tuples with unknown class labels

Model Construction



Name	Rank	Years	Tenured
Mike	Ass. Prof	3	No
Mary	Ass. Prof	7	Yes
Bill	Prof	2	Yes
Jim	Asso. Prof	7	Yes
Dave	Ass. Prof	6	No
Anne	Asso. Prof	3	No

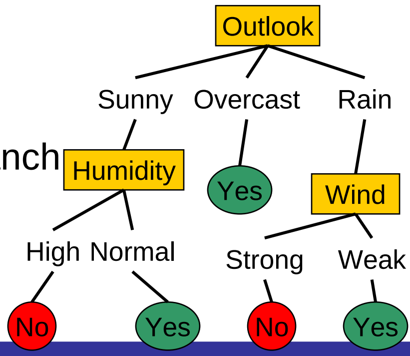
Model Application



Name	Rank	Years	Tenured
Tom	Ass. Prof	2	No
Merlisa	Asso. Prof	7	No
George	Prof	5	Yes
Joseph	Ass. Prof	7	Yes

Decision Tree

- A node in the tree – a test of some attribute
- A branch: a possible value of the attribute
- Classification
  - Start at the root
  - Test the attribute
  - Move down the tree branch



Training Dataset

Outlook	*Temp*	*Humid*	*Wind*	*PlayTennis*
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Mild	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Mild	Normal	Weak	Yes
Sunny	Mild	Normal	Strong	Yes
Overcast	Mild	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Mild	High	Strong	No

Basic Algorithm ID3

- Construct a tree in a top-down recursive divide-and-conquer manner
  - Which attribute is the best at the current node?
  - Create a nodes for each possible attribute value
  - Partition training data into descendant nodes
- Conditions for stopping recursion
  - All samples at a given node belong to the same class
  - No attribute remained for further partitioning
    - Majority voting is employed for classifying the leaf
  - There is no sample at the node

Which Attribute Is the Best?

- The attribute most useful for classifying examples
- Information gain and gini index
  - Statistical properties
  - Measure how well an attribute separates the training examples

Entropy

- Measure homogeneity of examples

$$Entropy(S) \equiv \sum_{i=1}^{c} - p_i \log_2 p_i$$

  - *S* is the training data set, and *pi* is the proportion of *S* belong to class *i*
- The smaller the entropy, the purer the data set

Information Gain

- The expected reduction in entropy caused by partitioning the examples according to an attribute

$$Gain(S, A) \equiv Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

Value(A) is the set of all possible values for attribute *A*, and S_v is the subset of *S* for which attribute *A* has value *v*

Example

Outlook	Temp	Humid	Wind	PlayTennis
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Mild	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Mild	Normal	Weak	Yes
Sunny	Mild	Normal	Strong	Yes
Overcast	Mild	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Mild	High	Strong	No

$$Entropy(S) = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14}$$

$$= 0.94$$

$$Gain(S, Wind) = Entropy(S) - \sum_{v \in \{Weak, Strong\}} \frac{|S_v|}{|S|} Entropy(S_v)$$

$$= Entropy(S) - \frac{8}{14} Engropy(S_{Weak}) - \frac{6}{14} Engropy(S_{Strong})$$

$$= 0.94 - \frac{8}{14} \times 0.811 - \frac{6}{14} \times 1.00 = 0.048$$

Extracting Classification Rules

- Each path from the root to a leaf → an IF-THEN rule
  - Each attribute-value pair along a path forms a conjunction
  - The leaf node holds the class prediction
  - IF age = "<=30" AND student = "no" THEN buys_computer = "no"
- Rules are easy to understand

Bagging

- Given a set S of s samples, generate a sequence of k independent bootstrap training sets
- Construct a sequence of classifiers C1,C2,…,Ck by using the same classification algorithm
- To classify an unknown sample X, let each classifier predict or vote
- The bagged classifier C* counts the votes and assigns X to the class with the “most” votes

Boosting Technique

- Assign every example an equal weight 1/N
- For t = 1, 2, …, T Do
  - Obtain a classifier C(t) under w(t)
  - Calculate the error of C(t) and re-weight the examples based on the errors. Samples incorrectly predicted have bigger weight
- Output a weighted sum of all the classifiers, with each classifier weighted according to its accuracy on the training set

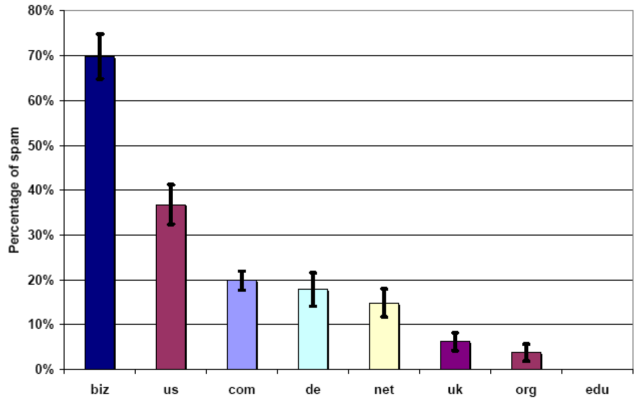
Spam Detection by Classification

- Use a set of spam web pages as a training data set
- Train a classification model (e.g., a decision tree)
- Apply the classification model to combat web spam

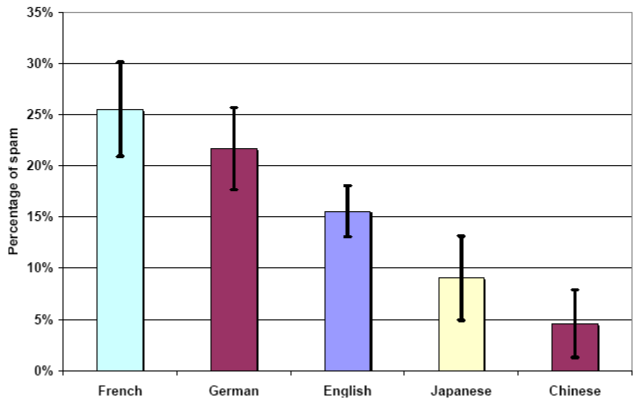
Heuristic Feature Selection

- Web page top domains
- Languages
- Number of words (body and title)
- Average word length
- Anchor words
- Visibility of content
- Repeating keywords
- The most common keywords
- N-gram likelihood
- [Ntoulas et al. WWW'06]

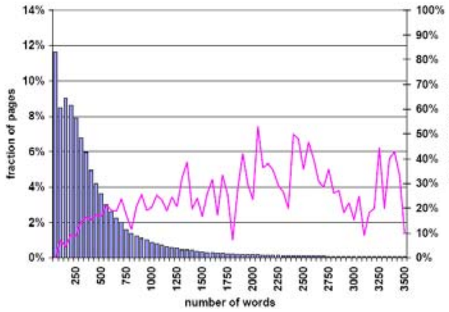
Web Page Top Domains

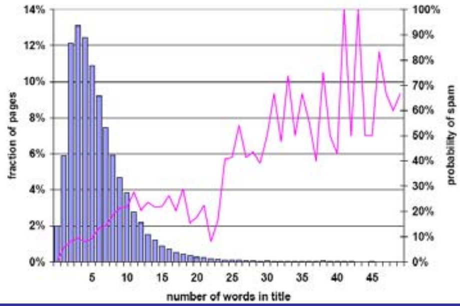


Languages

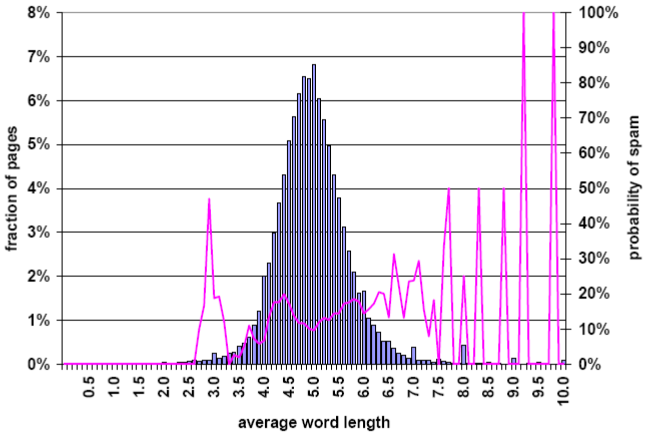


Number of Words



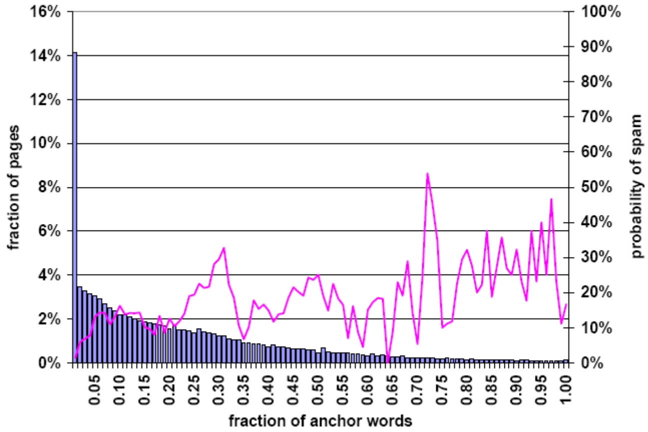


Average Word Length

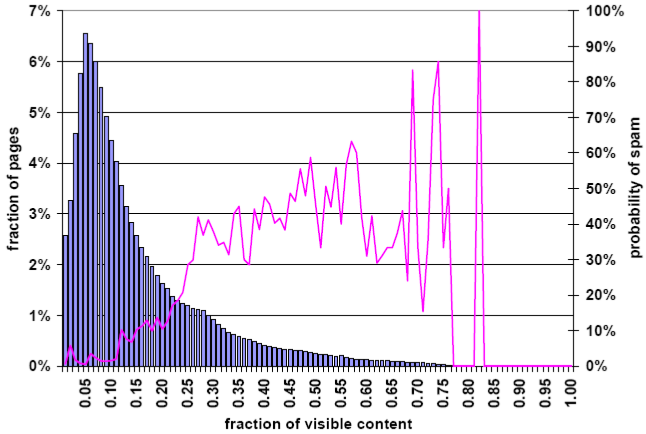


Fraction of Anchor Words

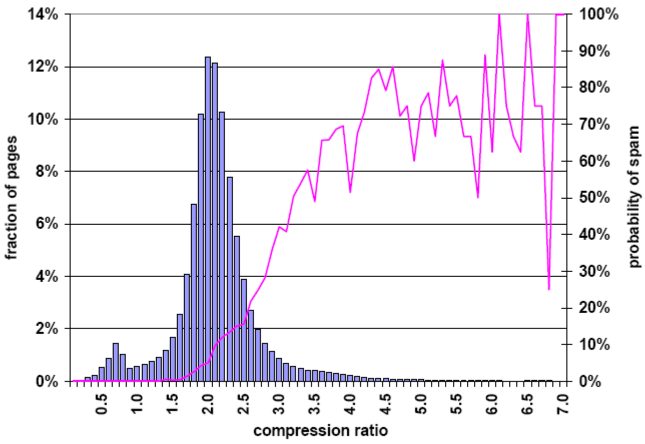
- Anchor words: words for hyperlinks



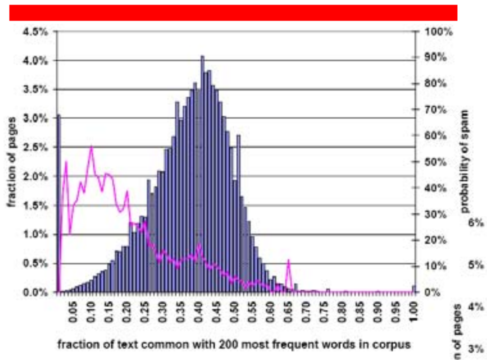
Visibility of Content

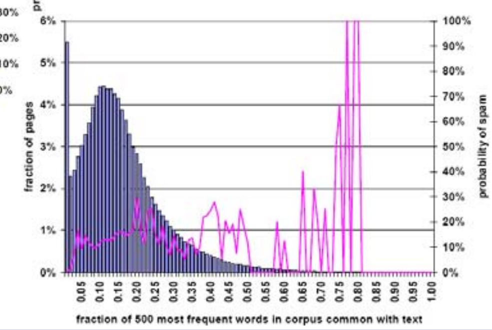


Repeating Keywords



Most Common Keywords





Using C4.5 to Combine Features

- Using bagging and boosting

class	recall	precision
spam	82.1%	84.2%
non-spam	97.5%	97.1%

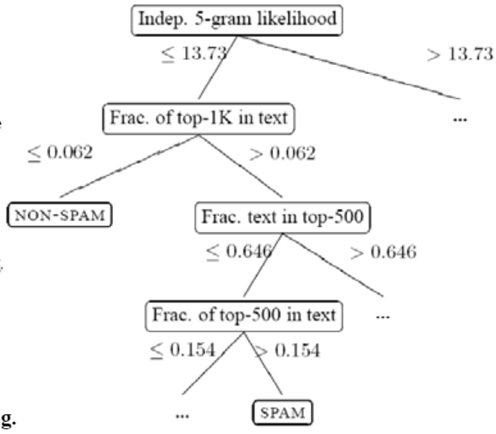
Table 1: Recall and precision of our classifie

class	recall	precision
spam	84.4%	91.2%
non-spam	98.7%	97.5%

Table 2: Recall and precision after bagging.

class	recall	precision
spam	86.2%	91.1%
non-spam	98.7%	97.8%

Table 3: Recall and precision after boosting.



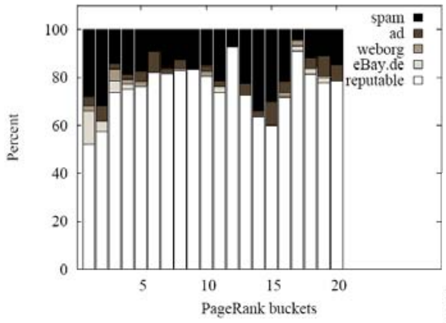
SpamRank: Ideas

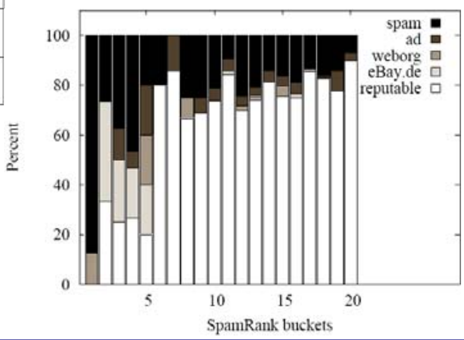
- Supporters of an honest (non-spam) page should not be overly dependent on one another
- The PageRank of the supporters of an honest page should follow a power law distribution as if a sample of the whole web
- Link spammers have a limited budget – boosting utility is important for supporters of spam pages
- [Benczur et al. AIRWeb’05]

SpamRank: A Three-Step Method

- Phase 1: select the supporters of each page by a Monte Carlo simulation
- Phase 2: pages are penalized if their supporters do not follow power law distribution in PageRank histogram
- Phase 3: compute SpamRank as PageRank personalized on the vector of penalties

PageRank versus SpamRank



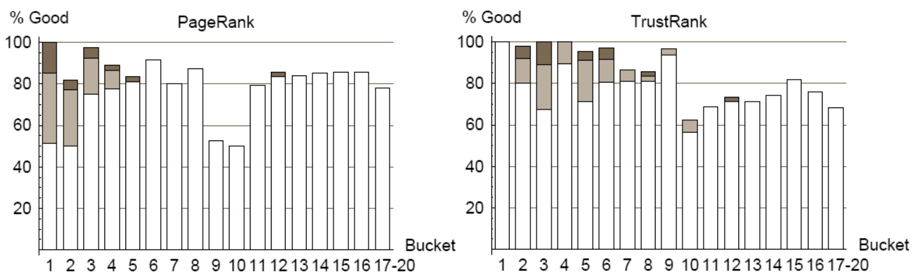


TrustRank: Ideas and Method

- Honest pages often point to honest pages and seldom point to spam pages
- Use a set of known honest pages as the seed set
  - Assign high trust scores to those pages
- Propagate the trust scores via out-links to unknown web pages – a PageRank computation procedure
- When the TrustRank converge, pages with high TrustRank scores are honest pages
- Critical issue: the seed set must be good and balanced
- [Gyongyi et al. VLDB’04]

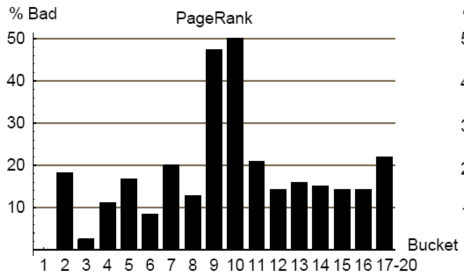
PageRank versus TrustRank

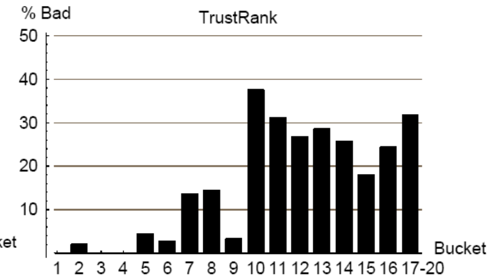
- Good pages



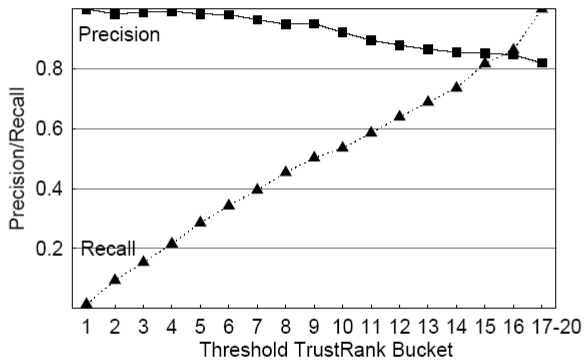
PageRank versus TrustRank

- Bad pages





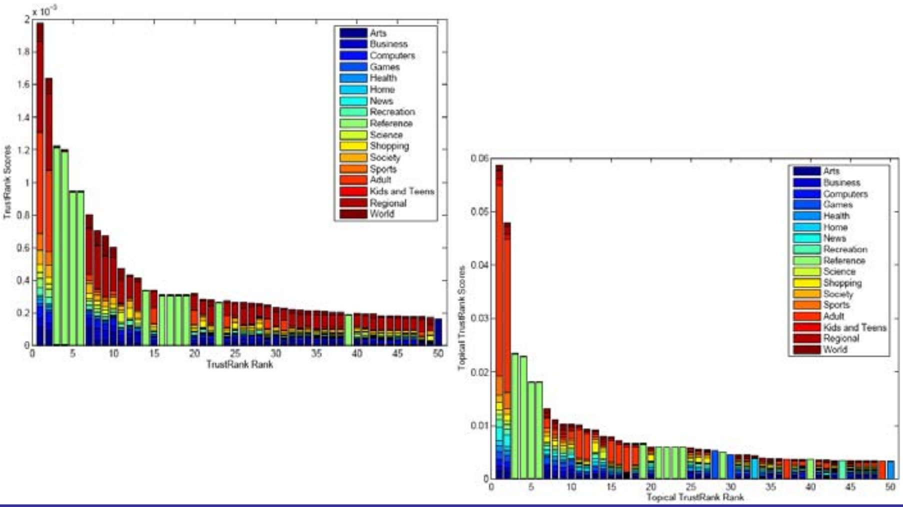
Precision and Recall



Topical TrustRank

- General TrustRank has a bias towards heavily represented communities in the seed set
- Use pages in well-maintained topic directories such as dmoz Open Directory Project as the seed set
  – Partition the seed set into topics
- Compute TrustRank score vectors on topics
- [Wu et al. WWW’06]

TrustRank versus Topical TrustRank

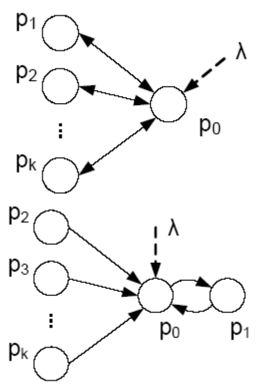


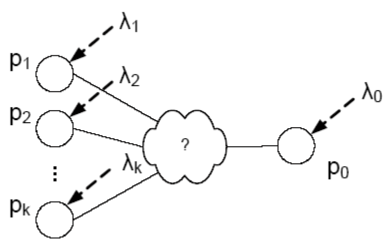
Spam Farms

- The set of pages supporting a spam page
- Three components
  - A single target page to be boosted by the spammer
  - A reasonable number of boosting pages that deliberately push the ranking of the target page
  - Some external links accumulated from pages outside the spam farm
- [Gyongyi and Garcia-Molina, VLDB'05]

Spam Farms

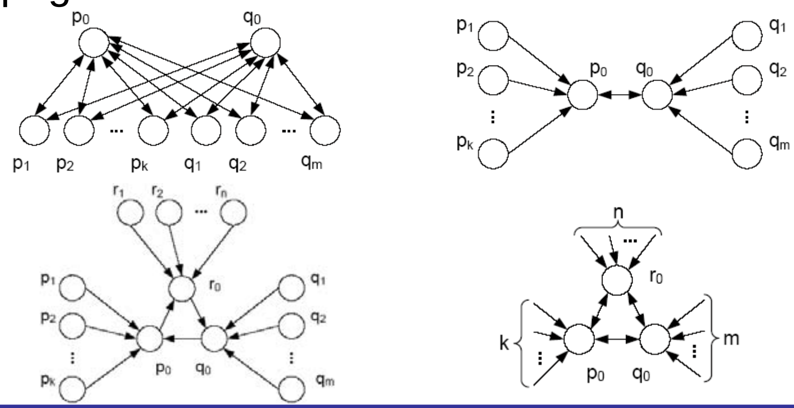
- Optimal structure for single target page
- General structure



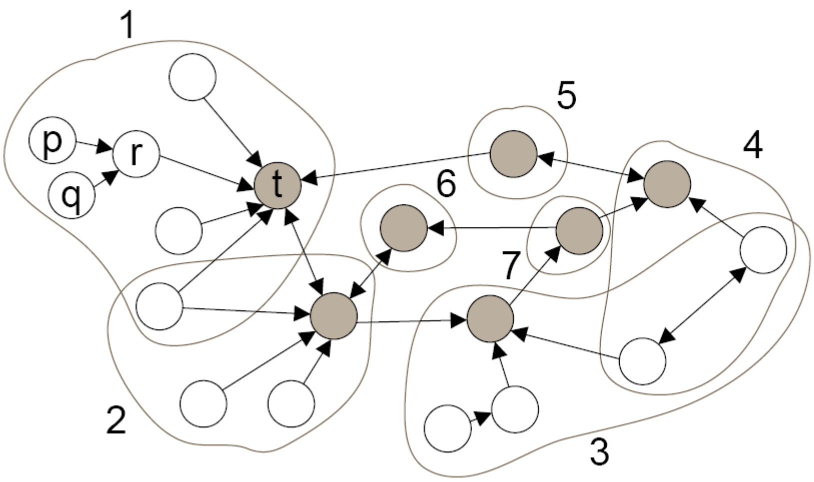


Spam Alliance

- A spam farm may boost multiple target pages



Irregular Spam Alliance



Questions Remained

- How can we derive spam farms in the real web?
- A spam page may play both link spam and content spam tricks?
- Is spamming as simple as black-and-white?

A Spamicity Approach

- Use spamicity to measure how likely a web page is spam
- Efficient spamicity-based link spam detection methods
- Efficient spamicity-based term spam detection methods
- [Zhou et al. SDM'08]

Page Farm Model

- Typically, link spam is a local activity.
  - Where does PR(p, G) come from?

$$PR(p,G) = d \sum_{p_i \in M(p)} \frac{PR(p_i, G)}{OutDeg(p_i)} + \frac{1-d}{N}$$

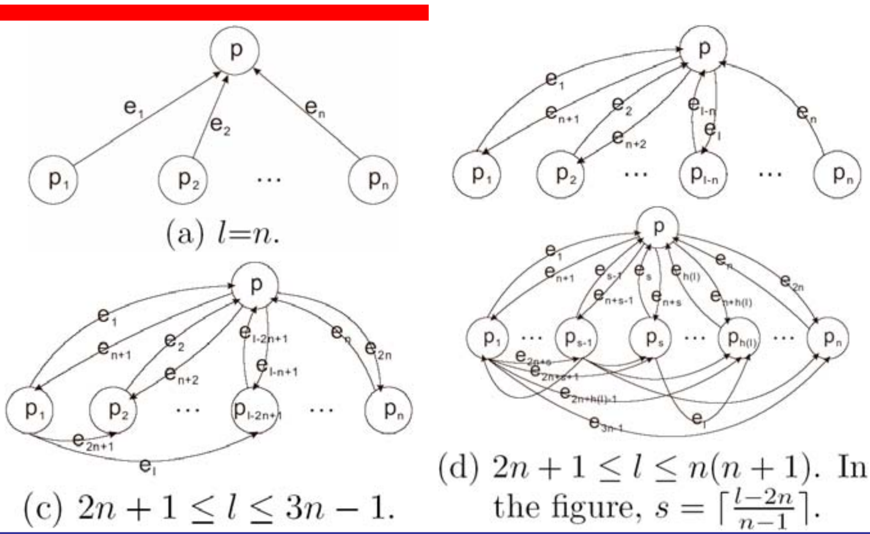
$$q_3 \rightarrow q_1 \rightarrow p \leftarrow q_2 \rightarrow q_4$$

- (θ,k)-page farm of page p: the minimal set of pages contributing to a θ portion of PR(p, G) and each page has a distance to p at most k
  - According to [Zhou and Pei, SDM'07], when θ >= 0.8 and k >= 3, the farms captures the local environments of web pages accurately

Utility-based Link Spamicity

- Given a page p, its page farm Farm(p) captures its local link structures
- Farm(p) should try to achieve the PageRank of p as high as possible
- The utility of Farm(p) is the ratio of the PageRank of p against the maximum PageRank that can be achieved

Optimal Spam Farms



(a) $l=n$.

(c) $2n+1 \leq l \leq 3n-1$.

(d) $2n+1 \leq l \leq n(n+1)$. In the figure, $s = \lceil \frac{l-2n}{n-1} \rceil$.

Utility-based Link Spamicity

$$ULSpam(p) = \frac{PR(p)}{PR_{max}(|V|,|E|)}$$

- ULSpam(p) can be used as a measure on the likelihood that p is link spam
  – It is an objective measure
  – It also works for those disguised link spam

Link Spam Detection Scenarios

- When the whole web graph is available
  - Search engine companies
  - Parties who have the access to data (e.g., by crawling the web)
  - But, the maintenance of the data is a big issue
- When the whole web graph is unavailable
  - Online spam detection (e.g., intelligent web browsers)
  - Efficient spam detection (e.g., only want to label a small set of pages)
    - Out-links: parsing the content of the page
    - In-links: querying web search engines using link search queries

Efficient Link Spam Detection

- Given a link spamicity threshold and a web page
  - Determine whether the link spamicity of the page is greater than or equal to the threshold
- Major calculation costs
  - Search engine querying load
  - Web page out-link parsing load

Local Greedy Search Method

- Page contributior

$$PCont(v,p) = \begin{cases} PR(p,G) - PR(p,G(V-\{v\})) & (v \neq p) \\ \frac{1-d}{N} & (v = p) \end{cases}$$

- Path contribution
  - Consider a path $P = v_0 \to v_1 \to \ldots \to v_n \to p$

$$LCont(P,p) = \frac{1}{N} d^{n+1}(1-d) \prod_{i=0}^{n} \frac{1}{OutDeg(v_i)}$$

- Page contribution and path contribution
  - PCont(v,p) can be calculated efficiently by summing up LCont(P,p)
- A local greedy search method
  - Given a target page p, greedily add pages with the highest page contribution to p into the farm Farm(p)
  - The procedure stops until Farm(p) achieves a θ portion of the PageRank score of p

Monotone Greedy Search Method

- The local greedy search method needs to extract the whole farm so as to calculate the link spamicity
- A critical observation: If pages are added in the page contribution descending order, the utility of adding new pages to improve the PageRank of the target page decreases monotonically
- A monotone greedy search method
  - Given a target page p, greedily add a page to the current farm Farm(p) which makes the largest improvement on PR(p)
  - The iteration continues until the link spamicity is lower than the link utility threshold, or all the pages within distance to p up to k are in the farm

Utility-based Term Spamicity

- If page p is term spam, to be relevant to a search query Q, p should try to achieve the TFIDF score as high as possible.
- The keywords in page p can be treated as the targeted keywords to which the builder of the page wants to make p relevant
- Utility-based term spamicity

$$UTSpam(p) = \frac{TFIDF(p,Q)}{TFIDF_{max}(p)}$$

- UTSpam(p) can be used as a measure on the likelihood that p is term spam
  - It is an objective measure

Char-Based Term Spamicity

- Keyword stuffing detection
  - Page body, page title, page meta tags, page anchor text
  - $H_i(p)$ (i=1,2,3,4): the ratio of the total number of keywords in each field against the number of distinct keywords in each field
- Invisible keywords detection
  - Set the keywords to have the same color as the page body
  - $H_5(p)$: the ratio of the number of invisible keywords in the body against the total number of keywords in the body
- Page URL keywords detection
  - Embed spam keywords in the URL address of the page.
  - $H_6(p)$: the ratio of the total length of keywords in the URL against the total length of the URL
- Characteristics-based term spamicity $CTSpam(p) = \sqrt[\gamma]{\frac{\sum_{i=1}^{6} H_i(p)^{\gamma}}{6}}$

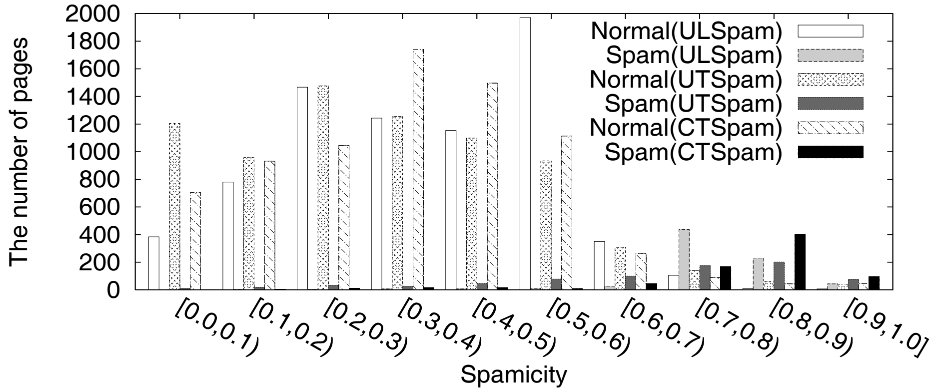
Efficient Term Spam Detection

- Given a term spamicity threshold and a web page
  - Determine whether the term spamicity of the page is greater than or equal to the threshold
- Major calculation costs
  - Web page keyword parsing load
  - Search engine querying load
  - IDF scores of keywords

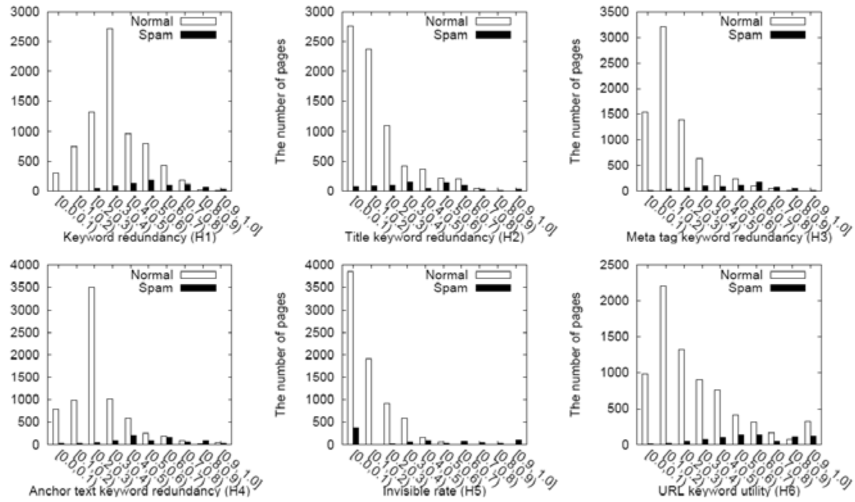
Data Set

- The webspam-UK2006 data set, released by Yahoo! Research Barcelona
- 8,239 pages are labeled manually, either “spam” or “normal”

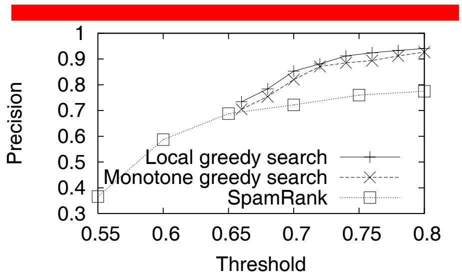
The Effectiveness of Spamicity

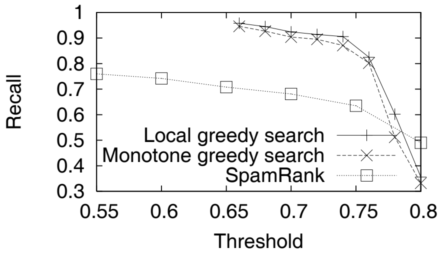


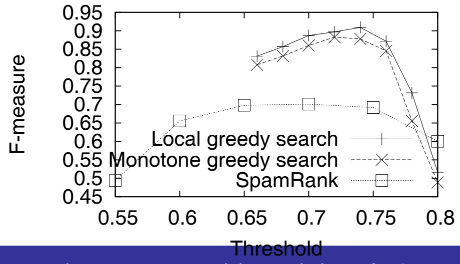
Content Spam Detection



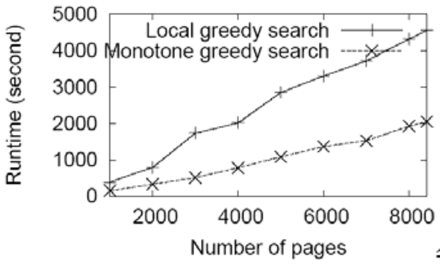
Comparisons of Three Spamicities

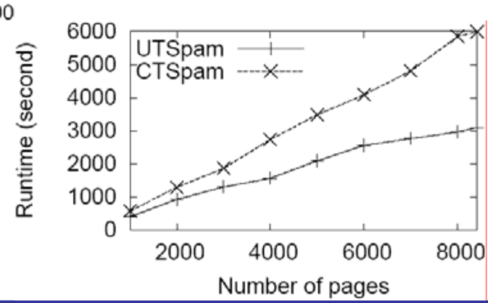






Scalability





Summary

- Web spam hurts information retrieval quality on the web
  - Link spam
  - Content spam
- Can data mining techniques help in web spam detection?
  - Classification approaches
  - PageRank-like approaches
  - Spam mass and spamicity approaches

Future Directions

- Effectiveness
  - More accurate spam detection?
- Efficiency
  - Scalable and online spam detection?
- PageRank is not all about web information retrieval
  - Spam detection for other ranking methods?
  - Spam detection for search of other types of data, e.g., images, videos, news, shopping, …

References (1)

[1] James Abello, Adam L. Buchsbaum, and Jeffery R. Westbrook. A functional approach to external graph algorithms. In Gianfranco Bilardi, Giuseppe F. Italiano, Andrea Pietracaprina, and Geppino Pucci, editors, *Proceedings of the 6th Annual European Symposium on Algorithms (ESA'98)*, volume 1461 of *Lecture Notes in Computer Science*, pages 332–343, London, UK, August 1998. Springer-Verlag.

[2] Reka Albert, Hawoong Jeong, and Albert-Laszlo Barabasi. The diameter of the world wide web. *Nature*, 401:130–131, 1999.

[3] David Aldous. *Random Walks on Finite Groups and Rapidly Mixing Markov Chains.* Springer-Verlag, Berlin, 1983.

[4] Lars Backstrom, Dan Huttenlocher, Jon Kleinberg, and Xiangyang Lan. Group formation in large social networks: Membership, growth, and evolution. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'06)*, pages 44–54, New York, NY, USA, 2006. ACM Press.

[5] Ricardo A. Baeza-Yates and Berthier A. Ribeiro-Neto. *Modern Information Retrieval.* ACM Press/Addison-Wesley, 1999.

References (2)

[6] Pierre Baldi, Paolo Frasconi, and Padhraic Smyth. *Modeling the Internet and the Web: Probabilistic Methods and Algorithms.* Wiley, 2003.

[7] Albert-Laszlo Barabasi and Reka Albert. Emergence of scaling in random networks. *Science*, 286(15):509–512, 1999.

[8] J. A. Barnes. Class and committees in a norwegian island parish. *Human Relations*, 7:39–58, 1954.

[9] Andras A. Benczur, Karoly Csalogany, Tamas Sarlos, and Mate Uher. Spamrank: Fully automatic link spam detection. In *Proceedings of the 1st International Workshop on Adversarial InformationRetrieval on the Web (AIRWeb'05)*, 2005.

[10] Michael K. Bergman. The deep web: Surfacing hidden value. http://www.brightplanet.com/resources/details/deepweb.html.

References (3)

[11] Krishna Bharat and Monika R. Henzinger. Improved algorithms for topic distillation in a hyperlinked environment. In *Proceedings of the 21st ACM International Conference on Researchand Development in Information Retrieval (SIGIR'98)*, pages 104–111, Melbourne, AU, 1998.

[12] Zhiqiang Bi, Christos Faloutsos, and Flip Korn. The "dgx" gistribution for mining massive, skewed data. In *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'01)*, pages 17–26, New York, NY, USA, 2001. ACM Press.

[13] Monica Bianchini, Marco Gori, and Franco Scarselli. Inside pagerank. *ACM Transactions on Internet Technology*, 5(1):92–128, 2005.

[14] Andrei Broder, Ravi Kumar, Farzin Maghoul, Prabhakar Raghavan, Sridhar Rajagopalan, Raymie Stata, Andrew Tomkins, and Janet Wiener. Graph structure in the web. In *Proceedings of the 9th International Conference on World Wide Web (WWW'00)*, pages 309–320. North-Holland Publishing Co., 2000.

[15] R. S. Burt and M. Minor. *Applied Network Analysis: A Methodological Introduction*. Sage, Beverly Hills, 1983.

[16] Carlos Castillo, Debora Donato, Luca Becchetti, Paolo Boldi, Massimo Santini, and Sebastiano Vigna. A reference collection for web spam. *SIGIR Forum*, 40(2):11–24, 2006.

References (4)

[17] D. Chakrabarti, Y. Zhan, and C. Faloutsos. R-mat: A recursive model for graph mining. In *Proceedings of the 2004 SIAM International Conference on Data Mining (SDM'04)*, Philadelphia, PA, 2004. SIAM.

[18] Soumen Chakrabarti. *Mining the Web: Discovering Knowledge from HyperText Data*. Science and Technology Books, 2002.

[19] Soumen Chakrabarti, Byron Dom, and Piotr Indyk. Enhanced hypertext categorization using hyperlinks. In *Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data (SIGMOD'98)*, pages 307–318, New York, NY, USA, 1998. ACM Press.

[20] Junghoo Cho, Sourashis Roy, and Robert E. Adams. Page quality: in search of an unbiased web ranking. In *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data (SIGMOD'05)*, pages 551–562, New York, NY, USA, 2005. ACM Press.

[21] J. Coleman. *Foundations of Social Theory*. Harvard University Press, Harvard, 1990.

[22] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. McGraw-Hill Higher Education, 2001.

References (5)

[23] Reinhard Diestel. *Graph Theory (3rd Edition)*, volume 173. Springer-Verlag, Heidelberg, 2005.

[24] Michalis Faloutsos, Petros Faloutsos, and Christos Faloutsos. On power law relationships of the internet topology. In *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM'99)*, pages 251–262, New York, NY, USA, 1999. ACM Press.

[25] Dennis Fetterly, Mark Manasse, and Marc Najork. Spam, damn spam, and statistics: Using statistical analysis to locate spam web pages. In *Proceedings of the 7th International Workshop on the Web and Databases (WebDB'04)*, pages 1–6, New York, NY, USA, 2004. ACM Press.

[26] Dennis Fetterly, Mark Manasse, and Marc Najork. Detecting phrase-level duplication on the world wide web. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'05)*, pages 170–177, New York, NY, USA, 2005. ACM Press.

[27] G. W. Flake, R. E. Tarjan, and K. Tsioutsiouliklis. Graph clustering and minimum cut trees. *Internet Mathematics*, 1:385–408, 2004.

[28] Gary Flake, Steve Lawrence, and C. Lee Giles. Efficient identification of web communities. In *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'00)*, pages 150–160, Boston, MA, August 20–23 2000. ACM.

[29] Gary William Flake, Steve Lawrence, C. Lee Giles, and Frans Coetzee. Self-organization of the web and identification of communities. *IEEE Computer*, 35(3):66–71, 2002.

References (6)

[30] L. C. Freeman, D. R. White, and A. K. Romney. *Research Methods in Social Network Analysis*. George Mason University Press, Fairfax, VA, 1989.

[31] David Gibson, Jon M. Kleinberg, and Prabhakar Raghavan. Inferring web communities from link topology. In *UK Conference on Hypertext*, pages 225–234, 1998.

[32] Michelle Girvan and M. Newman. Community structure in social and biological networks. *PROC.NATL.ACAD.SCI.USA*, 99:7821, 2002.

[33] Antonio Gulli and Alessio Signorini. The indexable web is more than 11.5 billion pages. http://www.cs.uiowa.edu/ asignori/web-size.

[34] Zoltán Gyöngyi, Pavel Berkhin, Hector Garcia-Molina, and Jan Pedersen. Link spam detection based on mass estimation. In *Proceedings of the 32nd International Conference on Very Large Databases (VLDB'06)*, pages 439–450. ACM, 2006.

[35] Zoltán Gyöngyi and Hector Garcia-Molina. Link spam alliances. In *Proceedings of the 31st International Conference on Very Large Databases (VLDB'05)*, pages 517–528. ACM, 2005.

References (7)

[36] Zoltán Gyöngyi and Hector Garcia-Molina. Web spam taxonomy. In *Proceedings of the 1st International Workshop on Adversarial Information Retrieval on the Web (AIR-Web'05)*, 2005.

[37] Zoltán Gyöngyi, Hector Garcia-Molina, and Jan Pedersen. Combating web spam with trustrank. In *Proceedings of the 30th International Conference on Very Large Databases (VLDB'04)*, pages 576–587. Morgan Kaufmann, 2004.

[38] Zoltán Gyöngyi, Hector Garcia-Molina, and Jan Pedersen. Web content categorization using link information. Technical report, Stanford University, 2006.

[39] Jiawei Han and Micheline Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, 2003.

[40] Robert A. Hanneman and Mark Riddle. *Introduction to Social Network Methods*. University of California, Riverside, 2005.

[41] T. Haveliwala. Topic-sensitive pagerank. In *Proceedings of the 11st International World Wide Web Conference (WWW'02)*, pages 784–796, Honolulu, Hawaii, 2002. ACM.

[42] T. Haveliwala, A. Gionis, D. Klein, and P. Indyk. Evaluating strategies for similarity search of the web. In *Proceedings of the 11st International World Wide Web Conference (WWW'02)*, pages 432–442, Honolulu, Hawaii, 2002. ACM.

References (8)

[43] M. Henzinger, R. Motwani, and C. Silverstein. Challenges in web search engines. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJ-CAI'03)*, pages 1573–1579, 2003.

[44] John Hopcroft, Omar Khan, Brian Kulis, and Bart Selman. Natural communities in large linked networks. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'03)*, pages 541–546, New York, NY, USA, 2003. ACM Press.

[45] G. Jeh and J. Widom. Scaling personalized web search. In *Proceedings of the 12nd International World Wide Web Conference (WWW'03)*, pages 271–279, Budapest, Hungary, 2003. ACM.

[46] Richard M. Karp. Reducibility among combinatorial problems. *Complexity of Computer Computations*, 1972.

[47] David Kempe, Jon Kleinberg, and Eva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'03)*, pages 137–146, New York, NY, USA, 2003. ACM Press.

[48] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. In *Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithm (SODA'98)*, pages 668–677. ACM, 1998.

References (9)

[49] Jon M. Kleinberg, Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, and Andrew S. Tomkins. The web as a graph: Measurements, models and methods. *Lecture Notes in Computer Science*, 1627:1–17, 1999.

[50] Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, and Andrew Tomkins. Trawling the web for emerging cyber-communities. In *Proceeding of the 8th International Conference on World Wide Web (WWW'99)*, pages 1481–1493, New York, NY, USA, 1999. Elsevier North-Holland, Inc.

[51] A. Langville and C. Meyer. Deeper inside pagerank. *Internet Mathematics*, 1(3):335–380, 2004.

[52] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graphs over time: Densification laws, shrinking diameters and possible explanations. In *Proceeding of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining (KDD'05)*, pages 177–187, New York, NY, USA, 2005. ACM Press.

[53] P. R. Monge and N. S. Contractor. *Emergence of Communication Networks*. Sage, Thousand Oaks, CA, 2006. New Handbook of Organizational Communication.

[54] Rajeev Motwani and Ying Xu. Evolution of page popularity under random web graph models. In *Proceedings of the 25th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS'06)*, pages 134–142, New York, NY, USA, 2006. ACM Press.

References (10)

[55] Isheeta Nargis, David A. Pike, and Neil McKay. Neighborhoods in the web graph. In *Proceedings of the 4th Workshop on Algorithms and Models for the Web Graph (WAW'06)*, 2006.

[56] M. E. J. Newman. Detecting community structure in networks. *The European Physical Journal B - Condensed Matter*, 38(2):321–330, March 2004.

[57] Alexandros Ntoulas, Junghoo Cho, and Christopher Olston. What's new on the web?: the evolution of the web from a search engine perspective. In *Proceedings of the 13th International Conference on World Wide Web (WWW'04)*, pages 1–12, New York, NY, USA, 2004. ACM Press.

[58] Alexandros Ntoulas, Marc Najork, Mark Manasse, and Dennis Fetterly. Detecting spam web pages through content analysis. In *Proceedings of the 15th International World Wide Web Conference (WWW'06)*, pages 83–92, New York, NY, USA, 2006. ACM Press.

[59] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford University, 1998.

[60] John Scott. *Social Network Analysis Handbook*. Sage Publications Inc., 2000.

References (11)

[61] Dou Shen, Jian-Tao Sun, Qiang Yang, and Zheng Chen. A comparison of implicit and explicit links for web page classification. In *Proceedings of the 15th International Conference on World Wide Web (WWW'06)*, pages 643–650, New York, NY, USA, 2006. ACM Press.

[62] A. C. Thompson. *Minkowski Geometry*. Cambridge University Press, New York, 1996.

[63] Stanley Wasserman and Katherine Faust. *Social Network Analysis*. Cambridge University Press, 1994.

[64] B. Wellman and S. D. Berkowitz. *Social Structures: A Network Approach*. Cambridge University Press, Cambridge, 1988.

[65] Barry Wellman. For a social network analysis of computer networks: a sociological perspective on collaborative work and virtual community. In *Proceedings of the 1996 ACM SIGCPR/SIGMIS Conference on Computer Personnel Research (SIGCPR'96)*, pages 1–11, New York, NY, USA, 1996. ACM Press.

References (12)

[66] B. Wu and B. D. Davison. Identifying link farm spam pages. In *Proceedings of the 14th International World Wide Web Conference (WWW'05)*, pages 820–829, New York, NY, USA, 2005. ACM Press.

[67] Baoning Wu, Vinay Goel, and Brian D. Davison. Topical trustrank: Using topicality to combat web spam. In *Proceedings of the 15th International World Wide Web Conference (WWW'06)*, pages 63–72, New York, NY, USA, 2006. ACM Press.

[68] Ricardo Baeza Yates, Paolo Boldi, and Carlos Castillo. Generalizing pagerank: Damping functions for link-based ranking algorithms. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'06)*, pages 308–315, New York, NY, USA, 2006. ACM Press.

[69] H. Zhang, A. Goel, R. Govindan, K. Mason, and B. V. Roy. Making eigenvector-based reputation systems robust to collusion. In *Proceedings of the 3rd Workshop on Algorithms and Models for the Web Graph (WAW'04)*, volume 3243 of *Lecture Notes in Computer Science*, pages 92–104. Springer, October 2004.

References (13)

[70] B. Zhou and J. Pei. "Sketching Landscapes of Page Farms". In Proceedings of the 2007 SIAM International Conference on Data Mining (SDM'07), Minneapolis, MN, USA, April 26-28, 2007.

[71] B. Zhou, J. Pei, and Z. Tang. "A Spamicity Approach to Web Spam Detection". In Proceedings of the 2008 SIAM International Conference on Data Mining (SDM'08), Atlanta, GA, April 24-26, 2008.