Improving Web Spam Classification using Rank-time Features

Krysta Svore
Microsoft Research
AirWEB 2007

Joint work with Qiang Wu, Chris Burges, and Aaswath Raman

Our Problem

- Eliminate web spam from search results
- Perform classification at rank time
- Classify at the page level (URLs)
- Use page-level features
- Include some domain-based and link-based features

Our Problem

- Webspam is designed to fool search engines
- Webspam is designed both to get into the index and to fool the ranking algorithm
- Can we develop a classifier that webspam cannot fool?

Our Goal

- Catch webspam at rank time!
- Ranker is not trained to identify spam. It's solving a different problem.
- Detect webspam that even a ranker thinks is relevant!
- NOTE: We aim to solve the problem of ranking webspam, but this still leaves webspam in the index! Our approach is a last-resort hammer!

Dataset

- 31300 human-labeled (query,URL) pairs
  - Queries were frequency subsampled from Microsoft Live search engine
  - 10% labeled spam
- (query,URL) labeled as spam, non-spam, unknown
- Gathered in July 2006

Support Vector Machines (SVMs)

- Classify each (query,URL) as spam or non-spam
- Use linear SVM
  - Finds separating hyperplane with maximal margin in high-dimensional feature space
  - Choose linear kernel function

$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$$

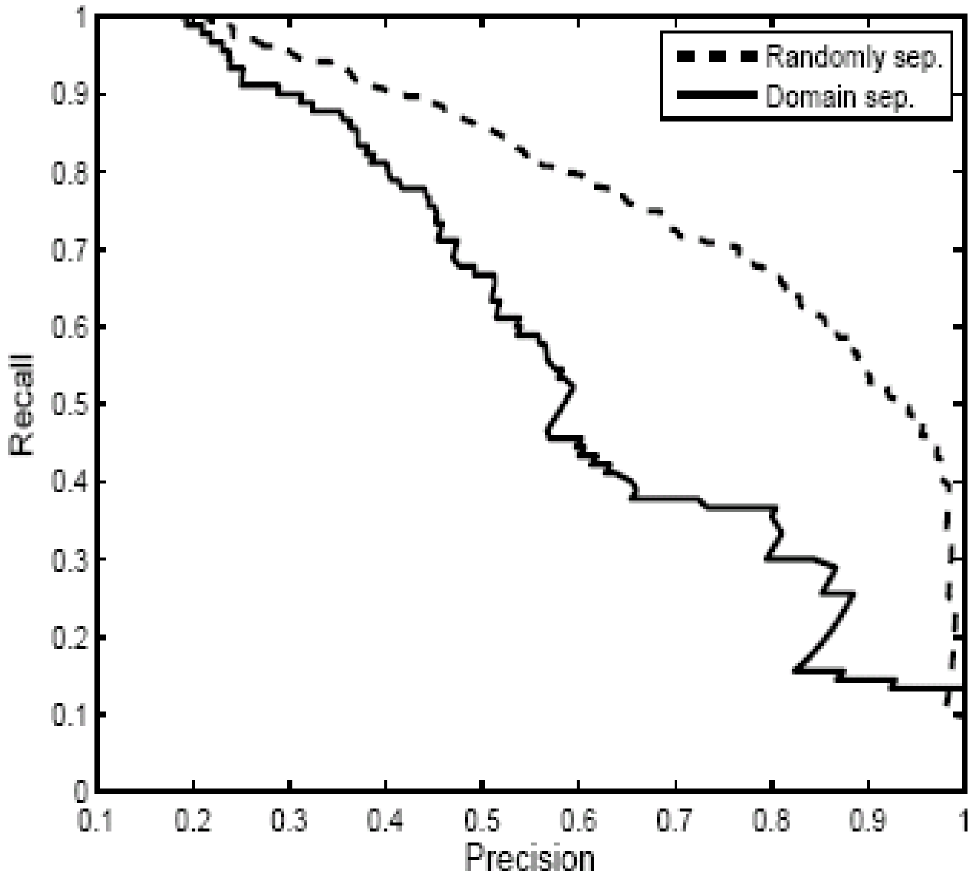
Evaluation

- TP: True positive (spam)
- FP: False positive
- TN: True negative (non spam)
- FN: False negative
- Precision: $\frac{TP}{TP+FP}$
- Recall: $\frac{TP}{TP+FN}$

Domain Separation

- Crucial to separate by domain
- Spammers buy large blocks of domains
  - Entire domains could be spam
- Feature is hash of domain
  - Classifier simply learns hash to spam label
- Misleading performance on test set
  - Generalizes poorly to unseen domains

Example of Domain Separation



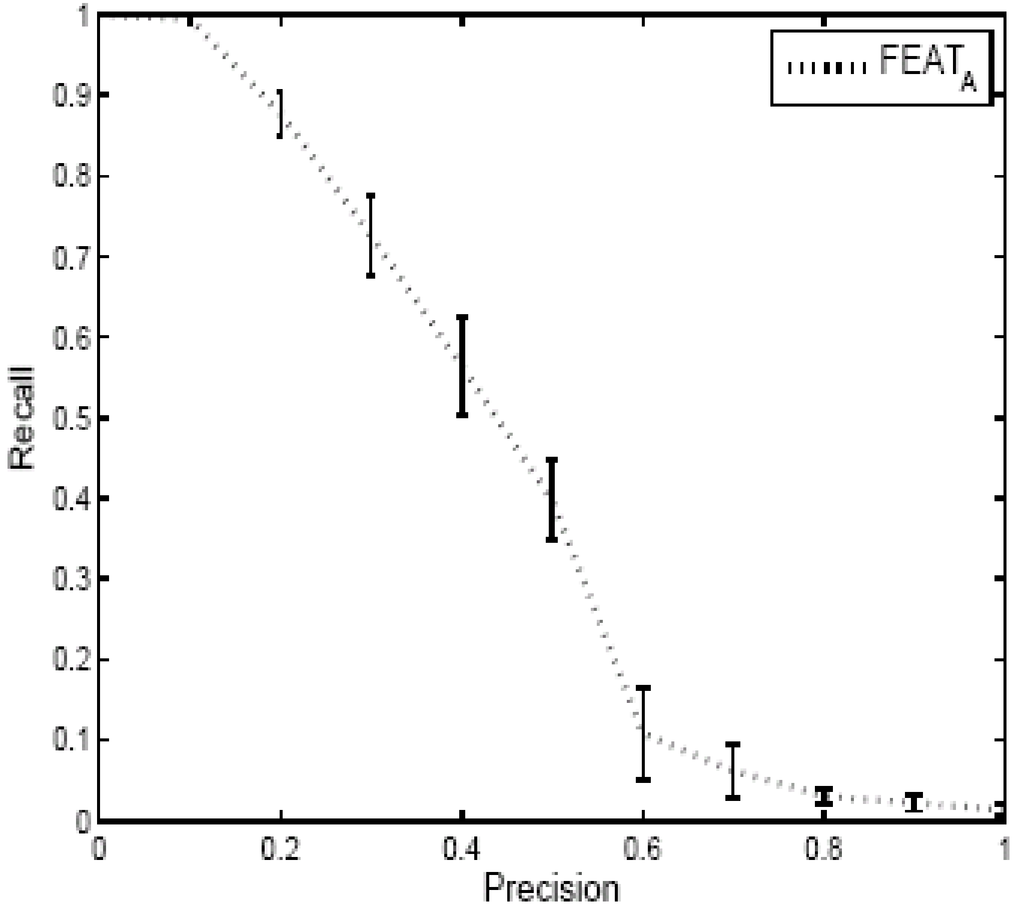
Motivation for Rank-time Features

- Spam appears in search results
- Spammers must “fool” index and rank algorithms
- Distribution of features is hard to match
- Train ranker on spam labels
- Spam pages will be outliers in distribution

Rank-independent Features

Number of spammy in-links
Top level domain of the site
Quality of phrases in the document
Density of keywords (spammy terms)

Rank-independent Results



Rank-time Features

- 360 rank-time features
- Separate into query-independent and query-dependent features
- Query-dependent features may reflect how spammers try to fool the ranker
- Page-level, domain-level, popularity, time

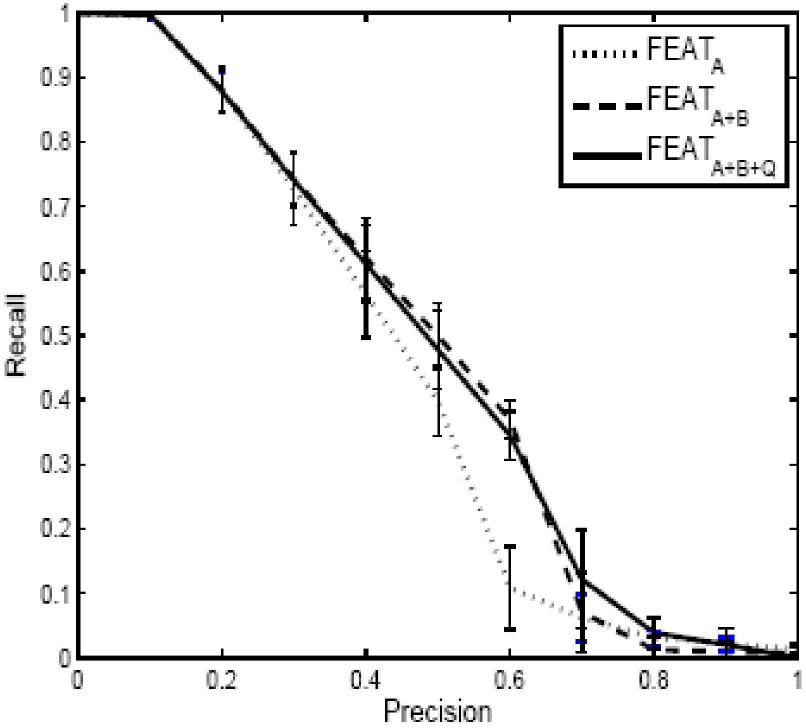
Query-independent Features

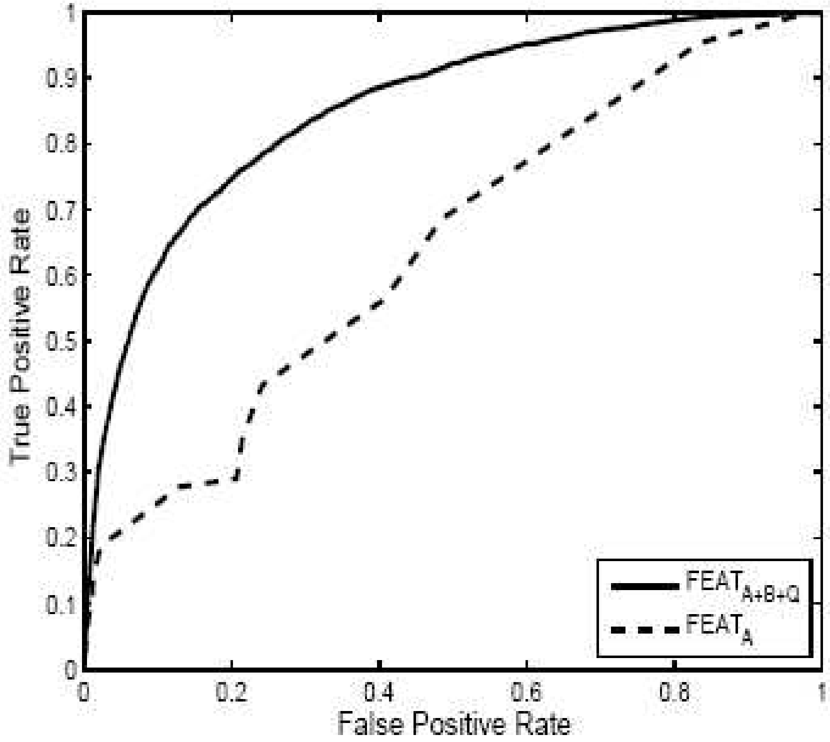
Page-level
Static rank
Most frequent term
Number unique terms
Total number of terms
Number of words in path
Number of words in title
Domain-level
Domain rank
Average number of words
Top-level domain
Popularity
Domain hits
Domain users
URL hits
URL users
Time
Date crawled
Last change date
Time since crawled

Query-dependent Features

Number query terms in title
Freq. counts of query term in doc.
Freq. counts of query term over all docs.
Number docs. containing query term
n-grams over query terms/ doc.

Rank-time Results





Conclusions

- Necessary to evaluate on domain-separated data to determine worst-case performance
- Data separation is a general problem
- Rank-time features improve classification performance by as much as 25% in recall at a set precision

Future Work

- Combine rank-time features with other approaches such as link-level classification
- Consider additional query-dependent features to further improve performance
- Evaluate method using other machine learning techniques

Acknowledgements

- Andrey Zaytsev
- Jacob Richman
- Matt Richardson
- Andy Laucius