



## APPROVAL SHEET

Title of Dissertation: Detecting Spam Blogs: An Adaptive Online Approach

Name of Candidate: Pranam Kolari  
Doctor of Philosophy, 2007

Dissertation and Abstract Approved: \_\_\_\_\_  
Dr. Timothy W. Finin  
Professor  
Department of Computer Science and  
Electrical Engineering

Date Approved: \_\_\_\_\_

## CURRICULUM VITAE

Name: Pranam Kolari.

Degree and date to be conferred: Doctor of Philosophy, 2007.

Date of Birth: September 28, 1979.

Place of Birth: Mangalore, India.

Collegiate institutions attended:

- UVCE, Bangalore University  
Bachelor of Engineering, Computer Science, 2001.
- University of Maryland, Baltimore County,  
Master of Science, Computer Science, 2004.
- University of Maryland, Baltimore County,  
Doctor of Philosophy, Computer Science, 2007.

Major: Computer Science.

Professional publications:

1. Pranam Kolari et al, "Spam Blogs and their Detection: Protecting the Quality of the Blogosphere", (In Submission)
2. Pranam Kolari, Tim Finin, Kelly Lyons, Yelena Yesha, Yaacov Yesha, Anupam Joshi, Stephen Perelgut, Jen Hawkins, "Internal Corporate Blogs: Empowering Social Networks within Large Organizations", CACM 2007, submitted
3. Tim Finin, Anupam Joshi, Pranam Kolari, Akshay Java, Anubhav Kale, Amit Karandikar, The Information ecology of social media and online communities, AI Magazine 28(3), Fall 2007

4. Pranam Kolari, Anupam Joshi, "Web Mining - Research and Practice", IEEE Computing in Science and Engineering, Web Engineering, July/August 2004.
5. Li Ding, Pranam Kolari, Zhongli Ding, Sasikanth Avancha, "Using Ontologies on the Semantic Web: A Survey", Ontologies: A Handbook of Principles, Concepts and Applications in Information Systems, Springer, December 2006
6. Pranam Kolari, Tim Finin, Yelena Yesha, Kelly Lyons, Stephen Perelgut, Jen Hawkins, "On the Structure, Properties and Utility of Internal Corporate Blogs", Proceedings of the International Conference on Weblogs and Social Media, (ICWSM 2007), Nominated for Best Paper Award
7. Pranam Kolari, Akshay Java, Tim Finin, Anupam Joshi, "Towards Spam Detection at Blog Ping Servers", Proceedings of the International Conference on Weblogs and Social Media, (ICWSM 2007)
8. Akshay Java, Tim Finin, Pranam Kolari, Anupam Joshi, Tim Oates, "FTM: Feeds That Matter, A Study of Bloglines Subscriptions", Proceedings of the International Conference on Weblogs and Social Media, (ICWSM 2007)
9. Anubhav Kale, Amit Karandikar, Pranam Kolari, Akshay Java, Tim Finin, Anupam Joshi, "Modeling Trust and Influence in the Blogosphere Using Link Polarity", Proceedings of the International Conference on Weblogs and Social Media, (ICWSM 2007)
10. Pranam Kolari, Akshay Java, Tim Finin, James Mayfield, Anupam Joshi, Justin Martineau, "Blog Track Open Task: Spam Blog Classification", TREC 2006 Blog Track, NIST, Gaithersburg
11. Akshay Java, Pranam Kolari, Tim Finin, James Mayfield, Anupam Joshi, Justin Martineau, "BlogVox: Separating Blog Wheat from Blog Chaff", Proceedings of the Workshop on Analytics for Noisy Unstructured Text Data, 20th International Joint Conference on Artificial Intelligence (IJCAI-2007)
12. Pranam Kolari, Akshay Java, Tim Finin, Tim Oates, Anupam Joshi, "Detecting Spam Blogs: A Machine Learning Approach", Proceedings of the 21st National Conference on Artificial Intelligence, (AAAI 2006), Boston, Acceptance Rate 26%
13. Pranam Kolari, Akshay Java, Tim Finin, "Characterizing the Splogosphere", Proceedings of the 3rd Annual Workshop on Weblogging Ecosystem: Aggregation, Analysis and Dynamics, (WWW 2006)

14. Pranam Kolari, Tim Finin, Anupam Joshi, "SVMs for the Blogosphere: Blog Identification and Splog Detection", Proceedings of the AAAI Spring Symposium 2006, Stanford
15. B. Aleman-Meza, Amit Sheth, Meenakshi Nagarajan, Cartik Ramakrishnan, I. Arpinar, Li Ding, Pranam Kolari, Tim Finin, Anupam Joshi, "Semantic Analytics on Social Networks: Experiences addressing the problem of Conflict of Interest ", Proceedings of WWW 2006, Acceptance Rate 11%
16. Li Ding, Rong Pan, Tim Finin, Anupam Joshi, Yun Peng, Pranam Kolari, "Finding and Ranking Knowledge on the Semantic Web", Proceedings of the International Semantic Web Conference, (ISWC 2005), Galway, Acceptance Rate 25%
17. Li Ding, Pranam Kolari, Tim Finin, Anupam Joshi, Yelena Yesha, "On Homeland Security and the Semantic Web, A Provenance and Trust Aware Inference Framework", AAAI Symposium on Homeland Security, (AAAI 2005)
18. Tim Finin, Li Ding, Rong Pan, Anupam Joshi, Pranam Kolari, Akshay Java, Y. Peng, "Swoogle: Searching for Knowledge on the Semantic Web", AAAI 2005, Intelligent Systems Demo
19. Li Ding, Pranam Kolari "On Modeling and Evaluating Trust Network Inference", Seventh International Workshop on Trust in Agent Societies, (AAMAS 2004), New York
20. Pranam Kolari, Tim Finin, Yelena Yesha, Kelly Lyons, Stephen Perelgut, Jen Hawkins, "Policy Management of Enterprise Systems: A Requirement Study", Proceedings of the IEEE Workshop on Policy for Distributed Systems and Networks (POLICY 2006)
21. Pranam Kolari, Li Ding, Lalana Kagal, Shashidhara Ganjugunte, Anupam Joshi, Tim Finin, "Enhancing Web Privacy Protection through Declarative Policies", Proceedings of the IEEE Workshop on Policy for Distributed Systems and Networks (POLICY 2005), Acceptance Rate 20%
22. Pranam Kolari, Tim Finin, "Memeta: A Framework for Multi-relational Analytics on the Blogosphere ", Proceedings of the 21st National Conference on Artificial Intelligence, (AAAI 2006), Boston, Student Abstract
23. Pranam Kolari, Tim Finin, Yelena Yesha, Kelly Lyons, Stephen Perelgut , Jen Hawkins, "Conversations on the Blogosphere: An Enterprise Perspective", CASCON 2006 Exhibit Program, Toronto, Canada

24. Pranam Kolari, Li Ding, "BPEL for Web Services: Rounding Off the Essentials", SDA-Asia Computing Magazine, June/July 2005

Professional positions held:

- Researcher (October 2007 - Present).  
Search and Advertising, Yahoo! Applied Research  
Santa Clara, CA, USA
- Graduate Research Assistant (January 2004 - September 2007).  
Department of Computer Science and Electrical Engineering,  
University of Maryland, Baltimore County, Baltimore, Maryland, USA.
- Graduate Teaching Assistant (August 2002 - December 2003).  
Department of Computer Science and Electrical Engineering,  
University of Maryland, Baltimore County, Baltimore, Maryland, USA.
- Research Intern (May 2004 - August 2004).  
IBM T. J. Watson Research, Yorktown Heights, NY, USA.
- Visiting Ph.D. Fellow (August 2005 - October 2005).  
IBM Toronto Software Labs, Markham, ON, Canada.
- Visiting Ph.D. Fellow (May 2006 - July 2006).  
IBM Toronto Software Labs, Markham, ON, Canada.
- Software Engineer (October 2001 - July 2002).  
IBM Global Services Ltd., Bangalore, Karnataka, India.

Invited Talks, Tutorials, Press Mentions, Honors:

- Talk: On Leveraging Social Media  
Microsoft Live Labs  
Seattle, WA, USA
- Tutorial: Spam in Blogs and Social Media  
International Conference on Weblogs and Social Media (ICWSM), 2007  
Boulder, CO, USA
- Talk: The Business of Blogging  
CASCON 2007  
Toronto, ON, Canada
- Talk: Corporate Blogs, The Good, Bad and the Wonderful  
European Marketing Symposium, 2007  
IBM Europe
- Press Mentions: On Spam Blogs  
Wired, Red Herring, Baltimore Sun, Information Week
- Honors: US Blogging Scholarship
- Program Committee:  
Social Computing: How the Social Web (R)evolution is Changing the Business Landscape, CASCON  
2006, **Organizer**  
International Conference on Weblogs and Social Media, ICWSM 2008  
Data Engineering for Blogs, Social Media, and Web 2.0, Workshop, ICDE 2008, **Organizer**  
AIRWeb: Adversarial Information Retrieval on the Web, Workshop, WWW 2008



## ABSTRACT

Title of Dissertation: Detecting Spam Blogs: An Adaptive Online Approach

Pranam Kolari, Doctor of Philosophy, 2007

Dissertation directed by: Dr. Timothy W. Finin  
Professor  
Department of Computer Science and  
Electrical Engineering

Weblogs, or blogs are an important new way to publish information, engage in discussions, and form communities on the Internet. Blogs are a global phenomenon, and with numbers well over 100 million they form the core of the emerging paradigm of Social Media. While the utility of blogs is unquestionable, a serious problem now afflicts them, that of spam. Spam blogs, or splogs are blogs with auto-generated or plagiarized content with the sole purpose of hosting profitable contextual ads and/or inflating importance of linked-to sites. Though estimates vary, splogs account for more than 50% of blog content, and present a serious threat to their continued utility.

Splogs impact search engines that index the entire Web or just the blogosphere by increasing computational overhead and reducing user satisfaction. Hence, search engines try to minimize the influence of spam, both prior to indexing and after indexing, by eliminating splogs, comment spam, social media spam, or generic web spam. In this work we further the state of the art of splog detection prior to indexing.

First, we have identified and developed techniques that are effective for splog detection in a supervised machine learning setting. While some of these are novel, a few others confirm the utility of techniques that have worked well for e-mail and Web spam detection in a new domain i.e. the blogosphere. Specifically, our techniques identify spam blogs using URL, home-page, and syndication feeds. To enable the utility of our techniques prior to indexing, the emphasis of our effort is fast online detection.

Second, to effectively utilize identified techniques in a real-world context, we have developed a novel system that filters out spam in a stream of update pings from blogs. Our approach is based on using filters

serially in increasing cost of detection that better supports balancing cost and effectiveness. We have used such a system to support multiple blog related projects, both internally and externally.

Next, motivated by these experiences, and input from real-world deployments of our techniques for over a year, we have developed an approach for updating classifiers in an adversarial setting. We show how an ensemble of classifiers can co-evolve and adapt when used on a stream of unlabeled instances susceptible to concept drift. We discuss how our system is amenable to such evolution by discussing approaches that can feed into it.

Lastly, over the course of this work we have characterized the specific nature of spam blogs along various dimensions, formalized the problem and created general awareness of the issue. We are the first to formalize and address the problem of spam in blogs and identify the general problem of spam in Social Media. We discuss how lessons learned can guide follow-up work on spam in social media, an important new problem on the Web.

DETECTING SPAM BLOGS  
AN ADAPTIVE ONLINE APPROACH

by  
Pranam Kolari

Dissertation submitted to the Faculty of the Graduate School  
of the University of Maryland in partial fulfillment  
of the requirements for the degree of  
Doctor of Philosophy  
2007



*For my parents.*

## ACKNOWLEDGEMENTS

My time spent in the Ph.D. program has been a very important part of my career. During this period, I have had advise from many great individuals, who were a constant source of encouragement, energy and expertise. I am deeply indebted to them.

First of all, the decision to pursue graduate studies was very important, one which I will always be proud of. I thank my dad and my brother who encouraged me to make this decision.

My advisor, Prof. Tim Finin, his technical advice has always been very important. His practicality and ability of identifying trends is inspirational. His many general policies and ideals are gems, and great life-skills. As a student working with him, not even once did I ever think “Is a Ph.D. right for me?”, and I thank Prof. Finin for this.

My committee members, Prof. Anupam Joshi always persisted in questioning me about progress. Prof. Joshi got me interested in the eBiquity group early on during my graduate studies, and I am very thankful. Thanks to Prof. Yelena Yesha for her support during our interactions with IBM, and for her advise throughout my graduate studies. Prof. Tim Oates has been a great guide on Machine Learning aspects of this work, and it has been a pleasure working with him. Thanks to Dr. James Mayfield for his many practical inputs, and to Dr. Nicolas Nicolov for his encouragement during our meetings at conferences.

Many thanks go to IBM, for their support and encouragement during my graduate studies, and for hosting me during my visits to Toronto. Thanks especially to Dr. Kelly Lyons, Jen Hawkins and Stephen Perelgut from IBM Center for Advanced Studies.

My friends at UMBC, I have had many positive things to draw from them, and I should mention Li Ding who I will always respect. Thanks everyone, for being part of this experience.

Thanks to my mom, who was always pushing me, and sometimes annoying me with the “When are you graduating?” question. Thanks also to Mayu and Savi for being there.

# TABLE OF CONTENTS

<b>I</b>	<b>INTRODUCTION</b>	<b>1</b>
I.A	The Blogosphere . . . . .	2
I.B	Spam Blogs . . . . .	3
I.C	The Splog Detection Problem . . . . .	5
I.D	Thesis Statement . . . . .	8
I.E	Contributions . . . . .	8
I.F	Thesis Outline . . . . .	9
<b>II</b>	<b>BACKGROUND</b>	<b>11</b>
II.A	Web Spam Workflow . . . . .	11
II.A.1	Identify Profitable Contexts/Leads . . . . .	13
II.A.2	Create Doorway Pages . . . . .	14
II.A.3	Inflate Doorway Importance . . . . .	15
II.A.4	Infiltrate SERPs . . . . .	16
II.A.5	Redirect users from Doorway . . . . .	17
II.A.6	Monetize from Users . . . . .	18
II.B	Related Work . . . . .	19
II.B.1	Local Models . . . . .	19
II.B.2	Global Models . . . . .	20
II.B.3	Adaptive Techniques . . . . .	20
II.B.4	Characterizing Problem Contexts . . . . .	21
II.C	Datasets . . . . .	22

II.C.1	Dataset 2005 . . . . .	22
II.C.2	Dataset 2006 . . . . .	25
II.D	META-PING . . . . .	27
II.D.1	Blacklist/Whitelist Filtering . . . . .	28
II.D.2	REGEX filtering . . . . .	28
II.D.3	URL Model Based Filtering . . . . .	29
II.D.4	Blog Home-Page Based Filtering . . . . .	29
II.D.5	Feed Based Filtering . . . . .	29
II.E	Supervised Machine Learning . . . . .	29
<b>III</b>	<b>DETECTING SPAM BLOGS</b>	<b>31</b>
III.A	URL Based Classifiers . . . . .	32
III.B	Home-Page Based Classifiers . . . . .	33
III.B.1	Words . . . . .	33
III.B.2	Wordgrams . . . . .	34
III.B.3	Anchor Text . . . . .	35
III.B.4	Outlink . . . . .	35
III.B.5	Character Grams . . . . .	36
III.B.6	HTML Tags . . . . .	36
III.C	Derived features . . . . .	45
III.D	Feed Based Features . . . . .	46
III.E	Blog Identification . . . . .	47
III.F	Relational Features . . . . .	50
III.F.1	Labeling Nodes using Local Models . . . . .	50
III.F.2	Link Features . . . . .	50
<b>IV</b>	<b>ENABLING ADAPTIVITY</b>	<b>53</b>
IV.A	Related Work . . . . .	54
IV.B	Our Contribution . . . . .	54
IV.C	Adapting Potential . . . . .	55
IV.D	Our Approach . . . . .	57



IV.E	Properties of the Ensemble . . . . .	58
IV.F	Use in META-PING system . . . . .	63
<b>V</b>	<b>CASE STUDIES</b>	<b>64</b>
V.A	TREC Blog Track 2006 . . . . .	65
	V.A.1    Impact of Splogs . . . . .	65
	V.A.2    Splog Task Proposal . . . . .	68
V.B	Blogosphere vs. Splogosphere . . . . .	72
	V.B.1    Frequency of Words . . . . .	74
	V.B.2    Link Structure . . . . .	75
V.C	Splogs and Ping Servers . . . . .	77
V.D	Spings in 2007 . . . . .	83
V.E	ICWSM Blog Spam . . . . .	86
V.F	Splog Software from Hell . . . . .	89
V.G	Splog Bait Experiment . . . . .	92
<b>VI</b>	<b>CONCLUSIONS</b>	<b>96</b>
VI.A	Spam Blogs . . . . .	96
VI.B	Spam in Social Media . . . . .	98
VI.C	Future Work and Open Problems . . . . .	100
<b>A</b>	<b>Appendix</b>	<b>101</b>
A.A	Splog Taxonomy . . . . .	101
	A.A.1    Non-blog Pages . . . . .	102
	A.A.2    Keyword Stuffed Blogs . . . . .	103
	A.A.3    Excerpt Stitched Blogs . . . . .	104
	A.A.4    Fully Plagiarized Blogs . . . . .	105
	A.A.5    Post Weaved Blogs . . . . .	106
	A.A.6    Link Spam Blogs . . . . .	107
	A.A.7    Other Spam . . . . .	108

# LIST OF TABLES

II.1	Hosting statistics for blogs indexed by Technorati in 2005. . . . .	23
II.2	Hosting statistics for blogs pinging weblogs.com in 2005. . . . .	23
II.3	Manual Labeling of blogs sampled from Technorati in 2005. . . . .	24
III.1	URL n-gram Classifier Evaluation. . . . .	38
III.2	Words Classifier Evaluation. . . . .	39
III.3	Wordgram Classifier Evaluation. . . . .	40
III.4	Anchor. . . . .	41
III.5	Outlink. . . . .	42
III.6	Chargram. . . . .	43
III.7	Tag. . . . .	44
III.8	Effectiveness of Specialized Features. . . . .	45
III.9	Blog Identification Baselines. . . . .	47
III.10	Feature types and their feature vector sizes used in experiments. . . . .	48
III.11	Results for the BHOME dataset using Binary Features. . . . .	49
III.12	Results for the BSUB dataset using Binary Features. . . . .	49
III.13	Interpretation of Probability Thresholds. . . . .	51
IV.1	URL n-gram Adaptive Potential. . . . .	55
IV.2	Anchor Adaptive Potential. . . . .	55
IV.3	Chargram Adaptive Potential. . . . .	56
IV.4	Outlink Adaptive Potential. . . . .	56
IV.5	Tag Adaptive Potential. . . . .	56

IV.6	Words Adaptive Potential. . . . .	56
IV.7	Wordgram Adaptive Potential. . . . .	56
IV.8	Q-statistics of Classifiers. . . . .	63
V.1	Proposed assessment scores for spam blog classification. . . . .	69

# LIST OF FIGURES

I.1	Internet Spam Taxonomy. . . . .	2
I.2	The Blog Indexing Infrastructure. . . . .	3
I.3	A Typical Splog. . . . .	4
I.4	Compromised Search Results Page. . . . .	5
I.5	Spam Detection Requirements across various platforms. . . . .	6
II.1	Web Spam Creation Workflow. . . . .	12
II.2	Workflow: Profitable Context. . . . .	13
II.3	Workflow: Doorway. . . . .	14
II.4	Workflow: Artificially Inflate Importance. . . . .	15
II.5	Workflow: SERP Infiltration. . . . .	16
II.6	Workflow: Cryptic Javascript Redirect. . . . .	17
II.7	Workflow: Monetize from Users. . . . .	18
II.8	Sampling for SPLOG-2006. . . . .	26
II.9	META-PING System Architecture. . . . .	28
III.1	URL n-gram Classifier Evaluation. . . . .	38
III.2	Words Classifier Evaluation. . . . .	39
III.3	Wordgram Classifier Evaluation. . . . .	40
III.4	Anchor. . . . .	41
III.5	Outlink. . . . .	42
III.6	Chargram. . . . .	43
III.7	Tag . . . . .	44

III.8	Distribution of number of items in feeds. . . . .	46
III.9	Learning Curve with Feeds. . . . .	47
III.10	Link features with binary encoding. . . . .	52
III.11	Link features with frequency encoding. . . . .	52
IV.1	Ensemble Feedback to Words. . . . .	59
IV.2	Ensemble Feedback to Wordgram. . . . .	59
IV.3	Ensemble Feedback to URLText. . . . .	60
IV.4	Ensemble Feedback to HTML Tags. . . . .	60
IV.5	Ensemble Feedback to Outlink. . . . .	61
IV.6	Ensemble Feedback to Anchor. . . . .	61
IV.7	Ensemble Feedback to Charactergram. . . . .	62
V.1	The number of splogs in the top 100 results for 50 TREC queries. . . . .	66
V.2	The number of splogs in the top 100 results of the TREC collection for 28 highly spammed query terms. . . . .	67
V.3	Blog host distribution in the BlogPulse dataset. . . . .	72
V.4	Probability Distribution of Blogs in BlogPulse. . . . .	73
V.5	Probability Distribution of Splogs in BlogPulse. . . . .	73
V.6	Distribution of top discriminating word-features in blogs and splogs. . . . .	74
V.7	In-degree distribution of authentic blogs subscribe to a power-law. . . . .	75
V.8	Out-degree distribution of authentic blogs subscribe to a power-law. . . . .	75
V.9	Host distribution of pings received by an Update Ping Server. . . . .	77
V.10	Ping Time Series of Italian Blogs on a single day. . . . .	78
V.11	Ping Time Series of Italian Blogs over five days. . . . .	78
V.12	Ping Time Series of Blogs on a single day. . . . .	79
V.13	Ping Time Series of Blogs over five days. . . . .	79
V.14	Ping Time Series of Splogs on a single day. . . . .	80
V.15	Ping Time Series of Splogs over a five day period. . . . .	80
V.16	Ping Time Series of .info blogs over a five day period. . . . .	81
V.17	Distribution of URLs that ping the Update Ping Server. . . . .	81

V.18	56% of all blogs pinging weblogs.com are splogs in 2007. . . . .	83
V.19	High PPC (Pay Per Click) contexts are the primary motivation to spam. . . . .	85
V.20	ICWSM Experiment. . . . .	86
V.21	ICWSM Experiment. . . . .	87
V.22	Splog Software. . . . .	90
V.23	Splog Software: RSS Magician. . . . .	90
V.24	RSS Magician Interface. . . . .	91
V.25	Splog Bait Experiment. . . . .	93
V.26	Splog Bait Search Result. . . . .	94
V.27	Splog Bait Example Splog. . . . .	95
VI.1	Splogs continue to be a problem in 2007. . . . .	97
VI.2	Spam on the Internet. . . . .	99
A.1	Non-blog page. . . . .	102
A.2	Keyword Stuffed Blog. . . . .	103
A.3	Excerpt Stitched Blog. . . . .	104
A.4	Fully Plagiarized Blog. . . . .	105
A.5	Post Weaved Blog. . . . .	106
A.6	Link Spam Blog. . . . .	107

## Chapter I

# INTRODUCTION

Spam, is beginning to be broadly considered the dark side of “marketing” (“the process or act of making products appeal to a certain demographic, or to a consumer”)<sup>1</sup>. However, the question of when marketing turns into spam continues to be highly subjective, domain specific, and debatable. “Unsolicited communication” is now considered a key differentiation. Early forms of spam were seen through junk mail, and later through telemarketing. The trend of decoupling and outsourcing of marketing tasks to third parties (less accountability and auditability of marketing mechanisms), and popularity of applications enabled by the Internet (low communication costs) have been feeding into the recent rise of Spam.

Spam on the Internet dates back over a decade, with its earliest known appearance as an email about the infamous MAKE.MONEY.FAST. campaign. This was also around the time when the term was first coined. The term “spam” is hence commonly (Webster) associated with “unsolicited usually commercial e-mail sent to a large number of addresses”. Spam has however co-evolved with Internet applications, and is now quite common on the World-Wide Web. As Social Media systems such as blogs, wikis and bookmark sharing sites have emerged, spammers have quickly developed techniques to infect them as well. The very characteristics underlying the Web, be it version 1.0, 2.0 or 3.0, also enable new varieties of spam.

Figure I.1 depicts the different forms of spam on the Internet, classified broadly into two categories. The first, represents direct targeting where communication between spammer and user is direct (as in e-mail). The second, and the form addressed in this work is indirect targeting where communication is promoted by compromising ranking algorithms used by search engines. The emphasis of this work is on the latter, and we address the problem of indirect spam through spam blogs.

---

<sup>1</sup><http://en.wikipedia.org/wiki/Marketing>

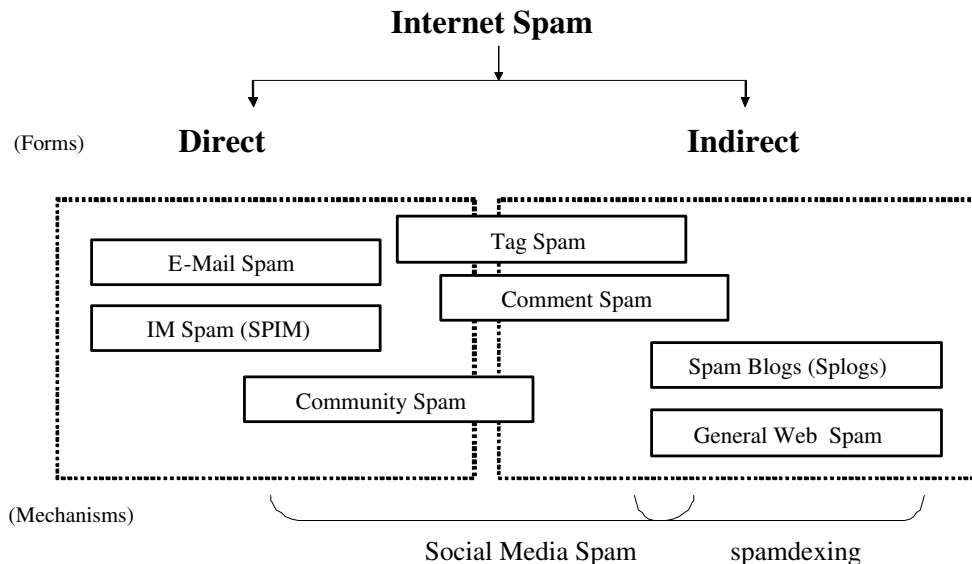


Figure I.1: Internet Spam Taxonomy.

In the rest of this chapter we first introduce blogs, and its collective whole, identified as the blogosphere. We then discuss the forms of spam prevalent in blogs, place it in the context of general spam, and introduce contributions made by this work.

## I.A The Blogosphere

Since its creation, the Internet has hosted diverse applications for content management, dissemination and exchange, ranging from the completely private, informal, dialogue oriented e-mail to the public, discourse oriented Web. A new form, constituted by weblogs, is now bridging the gap between the two. Weblogs, or blogs are web sites (pages) consisting of dated entries (posts) typically listed in reverse chronological order on a single page. The phenomenon of blogs and the social pulse they radiate is so influential that the subset they constitute is identified as the **Blogosphere**<sup>2</sup>.

Figure I.2 depicts the various systems and the underlying infrastructure that renders the Blogosphere. Blog publishing tools (step 1) enable bloggers to post content on their individual blogs. Unlike the Web, the time sensitive nature of blog content motivates Ping Servers and update streams (step 2,3). Ping Servers accept notifications (step 2) from newly updated blogs and route this on to downstream systems that harvest blogs, from now on referred to as blog harvesters. Blog hosting services also use independent mechanisms to

<sup>2</sup><http://en.wikipedia.org/wiki/Blogosphere>



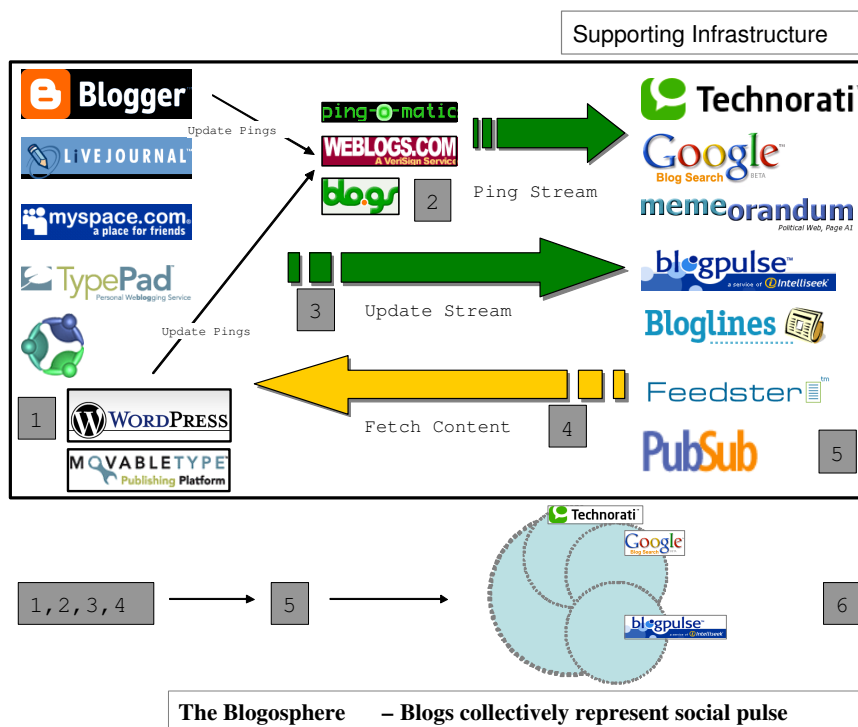


Figure I.2: The Blog Indexing Infrastructure.

notify (step 3) blog harvesters of newly updated blogs. Both ping servers and update streams enable a new form of push mechanism, unlike the pull model used by earlier harvesting systems (web crawlers).

Blog harvesters use information about the “live” blogosphere to identify newly updated blogs, and to fetch (step 4) and index (step 5) such new blog posts. Indexed results are made available to users in step 6. However, based on how effectively the blogosphere is mapped by blog harvesters, different overlapping versions of the blogosphere are rendered at any point of time. Clearly blog harvesters form key components of both blog and web search engines, and enable interesting new search and analysis based applications. Some common services enabled include blog post indexing and citation search, sentiment analysis for product marketing and meme tracking for trend and buzz monitoring.

## I.B Spam Blogs

The quality of a blog harvester is tied to how effectively and efficiently the blogosphere is instantiated. An instance of such a blogosphere is typically judged by: (i) reach and representation, (ii) freshness i.e., the ability to incorporate most recent blog posts, and (iii) robustness to spam. While (i) is quite well understood



Figure I.3: A Typical Splog.

in the context of the Web, and (ii) is enabled by ping servers, (iii) is addressed by contributions made in this thesis.

Blog harvesters are inundated by spam blogs, or splogs. The urgency in culling out splogs has become all the more evident in the last year. The problem's significance is frequently discussed and reported on by blog search and analysis engines [79, 16], popular bloggers [74], and through a formal analysis by us [48]. This analysis makes some disturbing conclusions on spam faced by ping servers. Approximately 75% of such pings are received from splogs.

Splogs are generated with two often overlapping motives. The first is the creation of fake blogs, containing gibberish or hijacked content from other blogs and news sources with the sole purpose of hosting profitable context based advertisements or functioning as doorways. The second, and better understood form, is to create false blogs, that realize a link farm [84] intended to unjustifiably increase the ranking of affiliated sites. Figure I.3 shows a post from a splog, obtained by querying the index of a popular blog search engine. As shown, it (i) displays ads in high paying contexts, (ii) features content plagiarized from other blogs, and (iii) contains hyperlinks that create link farms. Scores of such pages now pollute the blogosphere, with new ones springing up every moment. Eventually, splogs compromise search results of web and blog search engines. One such case is shown in figure I.4.

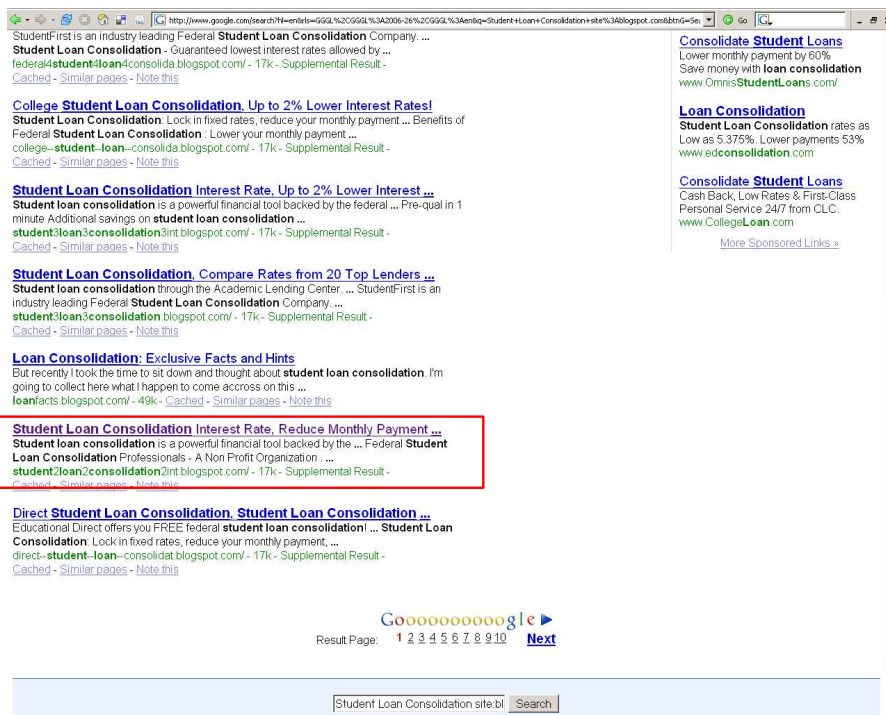


Figure I.4: Compromised Search Results Page.

## I.C The Splog Detection Problem

We next introduce the splog detection problem from the perspective of a blog harvester, a problem that motivates the core contributions made in this thesis.

In the classical web graph model  $G(V, E)$ , the set of nodes  $V$  represent web pages, and the set of edges  $E$  stand for hyper-links between these pages. In contrast, blog harvesters treat the Web using a slightly more intricate and tuned model,  $G(V, E)$ , where  $V = B \cup W$ . The membership of nodes in this web-graph is in either of  $B$  or  $W$ , where  $B$  is the set of all pages (permalinks) from blogs, and  $W$  is the set representing the rest of the Web. Splog detection is a classification problem within the blogosphere subset,  $B$ . Typically, the result of such a classification leads to disjoint subsets  $B_A, B_S, B_U$  where  $B_A$  represents all authentic content,  $B_S$  represents content from splogs and  $B_U$  represents those blog pages for which a judgment of authenticity or spam has not yet been made.

The splog detection problem for any node  $v \in B$ , can be expressed as:

$$P(v \in B_S | O(v)); P(v \in B_S | L(v))$$

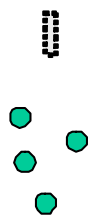
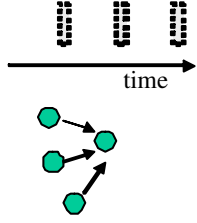
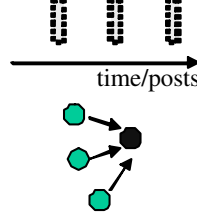
	E-MAIL	WEB	BLOGS
NATURE			
WHO USES IT?	<ul style="list-style-type: none"> <li>• Users</li> <li>• E-mail Service Provider</li> </ul>	<ul style="list-style-type: none"> <li>• Search Engines</li> <li>• Page Hosting Services (e.g. Tripod)</li> </ul>	<ul style="list-style-type: none"> <li>• Web Search Engines</li> <li>• Blog Search Engines</li> <li>• Blog Hosting Services</li> <li>• (Ping Servers)</li> </ul>
CONSTRAINTS	<ul style="list-style-type: none"> <li>• Fast Detection</li> <li>• Low Overhead</li> <li>• Online</li> </ul>	<ul style="list-style-type: none"> <li>• Batch Detection</li> <li>• Mostly Offline</li> </ul>	<ul style="list-style-type: none"> <li>• Fast Detection</li> <li>• Low Overhead</li> </ul>
ATTACKS	Image Spam, Character Replacement	Scripts, Doorways	Scripts, Doorways, Temporal Deception

Figure I.5: Spam Detection Requirements across various platforms.

$$P(v \in B_S | L(v), O(v))$$

where  $v \in B$ ,  $O(v)$  represents local features, and  $L(v)$  represents the link (relational) features, and  $P$  represents the conditional probability of  $v \in B_S$ .

On cursory investigation, this might still appear to be a classical web classification problem, and specifically a variation on the web spam problem [34]. However, the methodologies used by blog harvesters and the nature of the blogosphere make this an interesting special case of web classification. We summarize these intricacies in figure I.5 and discuss them in detail below:

**(i) Nature of the Problem.** First, e-mail spam detection is generally applied to e-mails individually, unlike splog and web spam detection, which also gives emphasis to a relational (link-based) component. Second, splogs are associated with a life-time as new posts and hyperlinks (in and out) are created, requiring that splog detectors be used on a blog multiple times during its lifetime, unlike e-mail spam detection.

**(ii) Nature of the Filter.** Depending on whether the blog harvester is owned by a blog or web search engine, they have access to different subsets of the Web i.e., harvesters at blog search engines employ preferential crawling and indexing towards the set  $B$ . Based on the harvester deployment scenario, a feasible solution should appropriately balance the emphasis on local models (properties local to a blog) and relational models using pages within  $B$ .

**(iii) Use.** E-mail spam filters are most commonly deployed at individual inboxes on personal computers. This imposes restrictions on machine learning techniques that support filters. Hence, Naïve Bayes technique with linear training costs is preferred for e-mail spam detection. Filters underlying spam blogs and web spam typically have no such restrictions, and commonly use the more robust, but computationally expensive (training) Support Vector Machines.

**(iv) Constraints.** In addition to reach, blog harvesters are judged by how quickly they can index and analyze new content. Hence, splogs must be eliminated quickly. This emphasis on speed differentiates splog detection from classical web spam detection that is usually applied hours or days after content creation. Filters using local models are hence preferred.

**(v) Attacks.** Although the underlying mechanisms that serve spam is similar across all three domains, adversarial attacks differ. For instance, the common attacks of image spam and character replacement (e.g. v1agra) prevalent in e-mail is less effective in blogs, where the presence of profitable contextual keywords is a requirement for indexing by search engines.

Overall, the emphasis of splog detection is on fast detection with an emphasis on exploiting local spam signals. Although this emphasis is higher when a blog harvester is associated with a blog search engine, it is also beginning to be a constraint in blog harvesters associated with web search engines (see [27]).

Based on these intricacies of the domain of splog detection, we next present the thesis statement and discuss contributions made in this work.

## **I.D Thesis Statement**

**Developing an effective, efficient and adaptive system to detect spam blogs is enabled through**

- (i) a continuous, principled study of the characteristics of the problem,**
- (ii) a well motivated feature discovery effort,**
- (iii) a cost-sensitive, real-time filtering implementation, and**
- (iv) an ensemble driven classifier co-evolution.**

## **I.E Contributions**

Though blogs as a publishing platform date back to a decade, splogs started appearing only in the early part of 2004, and presented a serious threat to blog harvesters only in 2005. Early discussion of the issue was limited to a few prominent bloggers. Many questions however remained answered, namely, what are their characteristics, how are they created, how many are created, followed by questions on how the problem differs from spam seen in e-mail and the more general Web. Our first contribution stems from answering these questions, through empirical studies and characterization using principled approaches. Through this analysis, we have also motivated constraints associated with the problem, i.e., blogs requiring fast online detection.

Our second contribution is through identifying effective features for splog detection. The problem is grounded in traditional text classification, specifically webpage classification. However, it is not clear as to which features that have worked well in other domains are applicable, and which new features are effective in this specialized domain. We have evaluated the effectiveness of features like words, word-grams and character-grams, and discovered new features based out of anchor-text, out-links and HTML tags, and validated their effectiveness. We have introduced the notion of feed based classification, and presented how classification performance evolves with blog lifecycle. We have evaluated the utility of relational features in this domain, and presented arguments to support our findings. Finally, we have also addressed the related problem of blog identification, i.e., separating out blogs from the rest of the Web, which is also a key functionality requirement of blog harvesters.

Our next contribution is through understanding deployment requirements of splog filters, and by implementing a first of a kind system that supports real-time, cost-effective filtering. We quantify classifier cost by page fetches, a simple yet effective metric, and use classifiers in an increasing cost pipeline, with the use of higher cost classifiers based on the output and confidence of low cost classifiers preceding them. Using this

principle, we have implemented a “META-PING” system, a name that signifies the system’s use between a ping server and a blog harvester in an online setting. We have validated its effectiveness through multiple deployments of its variations in real world settings, including at industrial partners (IceRocket, LMCO) and academic institutions (Harvard, UMBC). A full-version of this system deployed at UMBC has run over extended periods on a need-to basis, and supported blog harvesting and case studies on the growing problem of splogs both in 2006 and 2007.

Our final contribution is motivated by attributes shared by this adversarial classification problem with those of concept drift and co-training, combined with our experiences from real-world deployments. Concept drift has typically been addressed in the context of a stream of labeled instances, and co-training is used when the base learners are weak, assumptions that we have relaxed in this domain. We show how classifiers can co-evolve when supported by an ensemble of base classifiers. We have evaluated the use of this ensemble to retrain individual classifiers on a stream of unlabeled instances, and validated how such an approach is effective in splog detection. By unweaving the properties of the ensemble and the domain, we discuss other domains where this approach could be potentially effective. We also discuss how such adaptive classifiers can be incorporated into our developed META-PING system.

Finally, over the course of the last two years we have been the first to draw attention to the growing problem of Spam in Social Media.

## **I.F Thesis Outline**

The rest of this thesis is organized as follows.

In chapter II, we first provide an anatomy of the web spam problem, by enumerating a workflow typically used by spammers. We discuss related work on web spam detection and scope our work. We then discuss the META-PING implementation, a system we developed to filter out splogs, experiences from which has motivated most the work. To end the chapter, we detail the datasets used in the rest of the thesis, and introduce the machine learning context.

In chapter III, we discuss our contributions on feature identification for the splog detection problem. The emphasis of this chapter is on features that enable fast online splog detection. We propose novel features that are effective in this specialized domain, and provide an intuition supporting their effectiveness. We discuss the blog identification problem, a related issue, and an additional competency required of blog harvesters.

We introduce the problem of feed based classification, and show how certain features can be effective early in a blog life-cycle. We finally discuss feature types that support relational models, and discuss why such techniques are not that effective for the problem scoped out in this thesis.

In chapter IV, we motivate the problem of adaptive classifiers in an adversarial setting. We identify the nuances of the problem in the specialized context by tightening assumptions made in the concept drift and co-training contexts. We evaluate an approach that uses an ensemble of classifiers and show how it can be effective to adapt classifiers exposed to a stream of unlabeled instances. We discuss properties of the ensemble, and discuss how this technique can be used in the META-PING system.

In chapter V, we discuss our efforts on characterizing the problem. We first discuss the impact of splogs during the TREC Blog Track of 2006. We then discuss the peculiarities of splogs by comparing them with authentic blogs over three datasets. We also discuss a few experiments aimed at understanding the use of splogs for web spamming, the availability of “splog software”, and the case of content plagiarism in the blogosphere. Our efforts on characterizing the problem motivates identifying new features for splog detection, and we believe will serve well for future work by us, and by other researchers.

Finally, in chapter VI we conclude this thesis by laying out future work and introducing the broader problem of Social Media spam.



## Chapter II

# BACKGROUND

In this chapter we first detail the problem of web spam by discussing a workflow commonly used by spammers. We next discuss related work and scope out the contributions made by this thesis. To introduce the motivation behind this work, we next detail the META-PING system that we have implemented. We then detail labeled datasets extensively referred to in the rest of this thesis, and finally provide an introduction to machine learning techniques used in this work.

### II.A Web Spam Workflow

Web spammers use various techniques and are constantly adapting them. A comprehensive taxonomy is provided by Gyöngyi et al. [34]. Here, we discuss how spammers typically go about creating web spam by enumerating and discussing a commonly used workflow. We believe such a description will better enable focusing research efforts that thwart them. A somewhat similar analysis is presented by Wang et al [80] in their spam double funnel model.

Figure II.1 depicts key entities that are part of this workflow, and loosely consists of:

1. Identify Profitable Contexts/Leads
2. Create Doorway Pages
3. Inflate Doorway Importance
4. Infiltrate SERPs

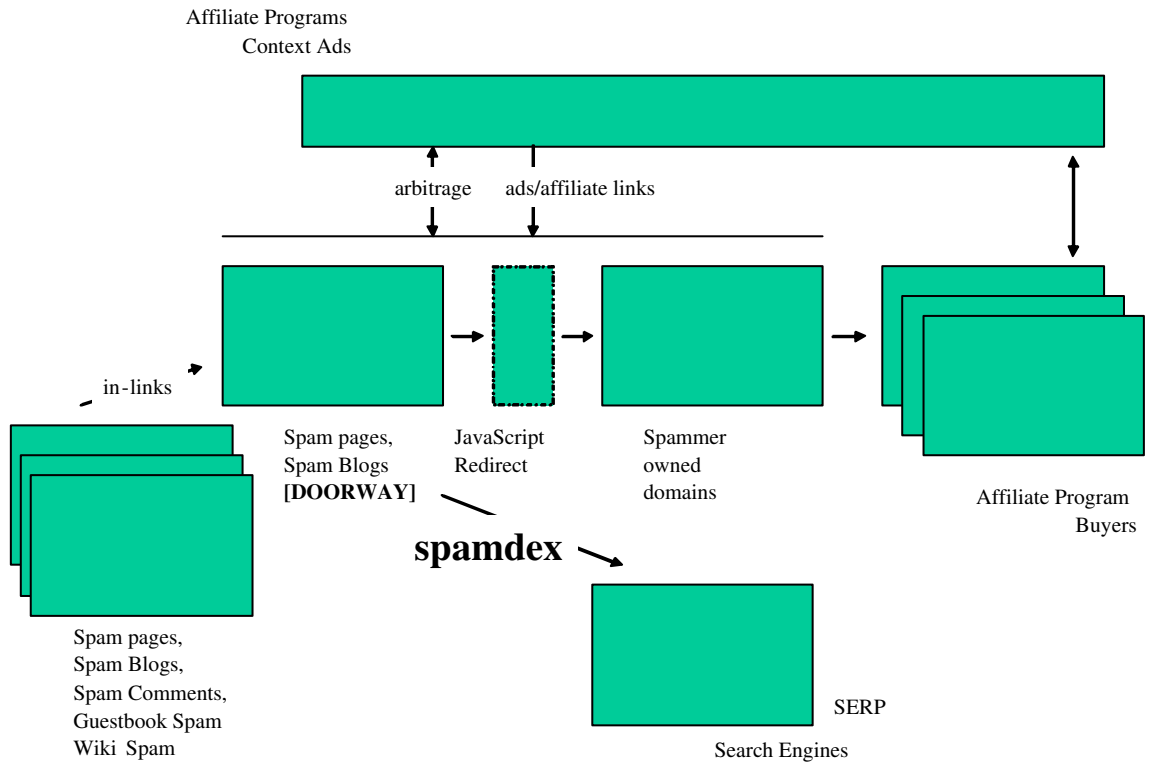


Figure II.1: Web Spam Creation Workflow.

5. Redirect from Doorway

6. Monetize from Visitors

It is quite uncommon for spammers to own websites that sell merchandise or gather user information online directly. They generally monetize through leads (referrals) that they provide to such websites from self-managed (“mislead”) pages. These leads are brokered either through contextual advertisement providers or affiliate programs. Spammers are experts in both identifying profitable leads and promoting pages they control to users through search engines (could also be through other channels like e-mail). We next discuss these steps as they relate to web spam in more detail.

The screenshot shows a Google search for "make money keywords". The search results page displays "Results 1 - 10 of about 40,700,000 for make money keywords. (0.28 seconds)". A red box highlights the number "40,700,000". The search results include several links related to making money online, such as "High Paying Keywords on AdSense to Make Money", "How to make money on your news content website", and "Google AdSense - an excellent way to make money free online". On the right side, there are sponsored links with titles like "Make Huge Money Now", "I was scammed 37 times", "Free Offer - Make Money", "Make Money - Free", "\$1000 a day from home", "Make \$10 in 5 minutes", and "Need part-time job?". A large text overlay in the center of the page reads "Identify Profitable Contexts".

Figure II.2: Workflow: Profitable Context.

## II.A.1 Identify Profitable Contexts/Leads

As the first step, spammers identify profitable leads brokered by contextual ad providers or affiliate programs. Contextual advertisements are text ads appearing on a page, matched to the content of the page based on textual similarity. Affiliate links are very similar to contextual ads, but provide better control over what ads (links) gets published i.e., with less of an emphasis on contextual similarity. Interestingly, a search on “affiliate programs” turns up more than 30 million results, with a number of directories<sup>1</sup> that list affiliate programs. A search for “make money keywords” turns up more than forty million results on popular search engines (see Figure II.2). Spammers also discuss their techniques on mailing lists and forums, most of which are controlled by them and generally open to a select few.

Each profitable lead in these identified contexts can draw from tens to hundreds of US dollars. Most of them are in the health, insurance, finance and travel contexts.

<sup>1</sup><http://www.affiliatescout.com/>

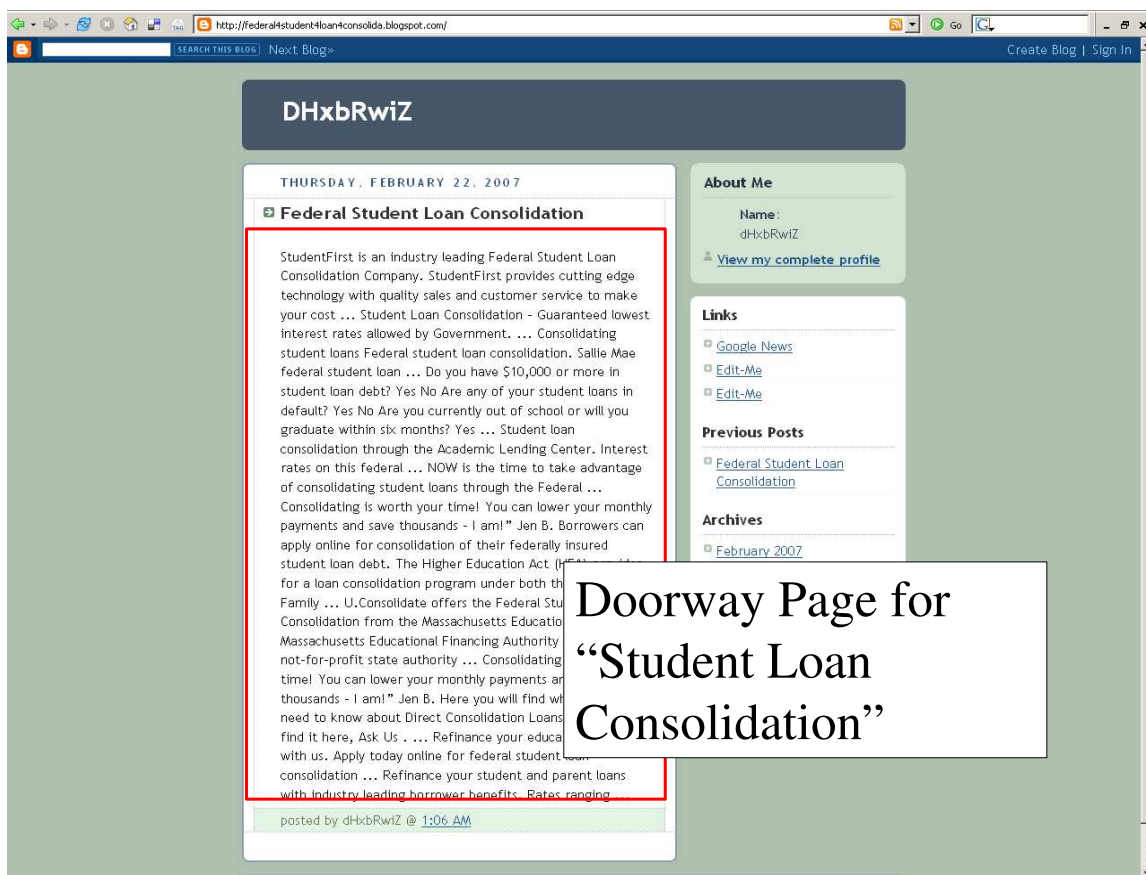


Figure II.3: Workflow: Doorway.

## II.A.2 Create Doorway Pages

Having identified profitable leads, the spammer has to next host them on self-managed websites, of what we refer to as “mislead pages”. Early forms of web spam involved getting these pages indexed directly by search engines. However as search engines got better at detecting spam, a technique of obfuscation involving doorway pages (which are indexed) is quite commonly used. Doorways redirect the user to mislead pages, and generally, there exists a  $n:1$  mapping between doorways and mislead pages. Mislead pages are hence no longer required to be indexed by search engines.

Doorways are chosen to be on well-established hosts that enjoy a high level of trust with search engines, most recently blogs (see figure II.3). Clearly, having such doorways makes the task of separating spam difficult for search engines. In addition, content for doorways and mislead pages are either plagiarized or auto-generated.

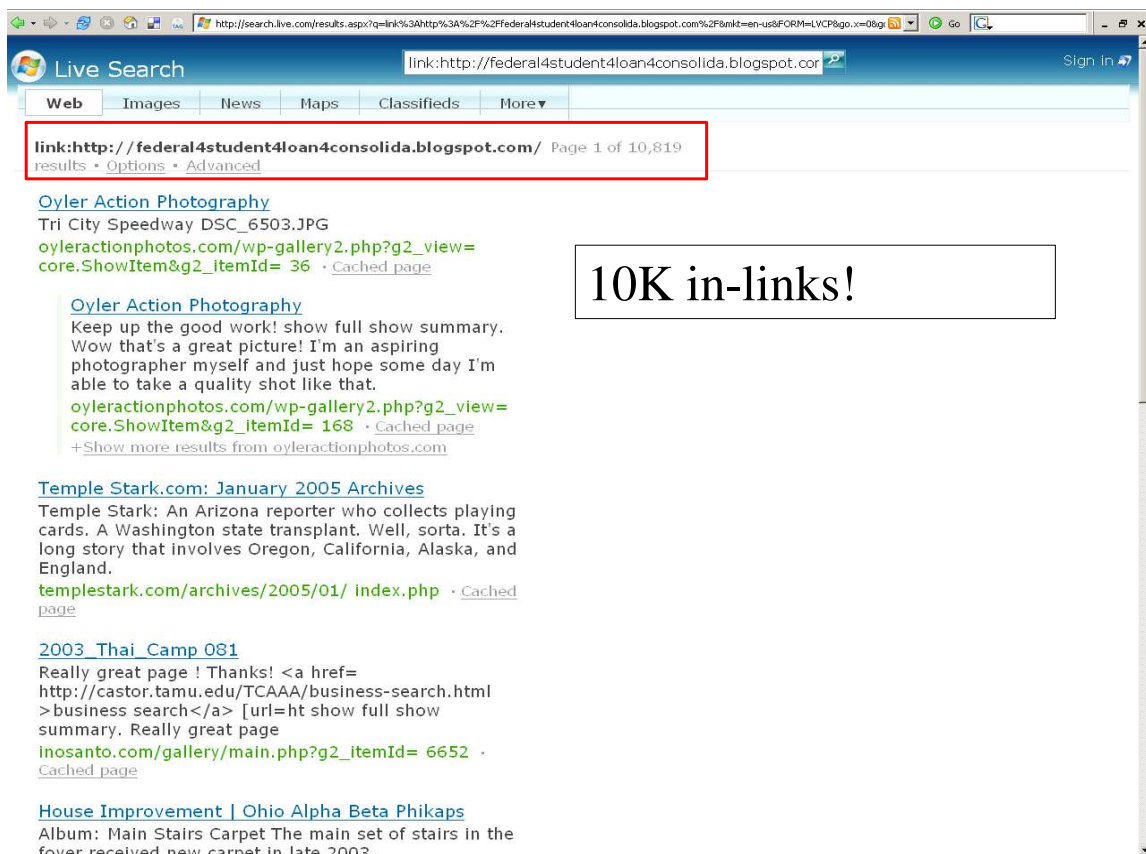


Figure II.4: Workflow: Artificially Inflate Importance.

### II.A.3 Inflate Doorway Importance

Search engines employ fairly robust algorithms for ranking search results, mainly based on relevance and importance of web-pages. Relevance is based on local properties of the page including content, meta-tags, title and other stylistic characteristics of the page. Importance is based on global properties of the page measured by the number and quality of incoming links to the page, formally described by PageRank.

Spammers inflate both relevance and importance of doorways. Inflating importance requires that they spam other writeable pages on the Web, most common of which are guestbooks, wikis and blog comments. An alternative approach is to generate farms of blogs hosted on trusted domains that point to doorways. A new technique is to incorporate hidden links on pages served by compromised web servers. Figure II.4 shows a doorway that has accumulated more than ten thousand in-links.

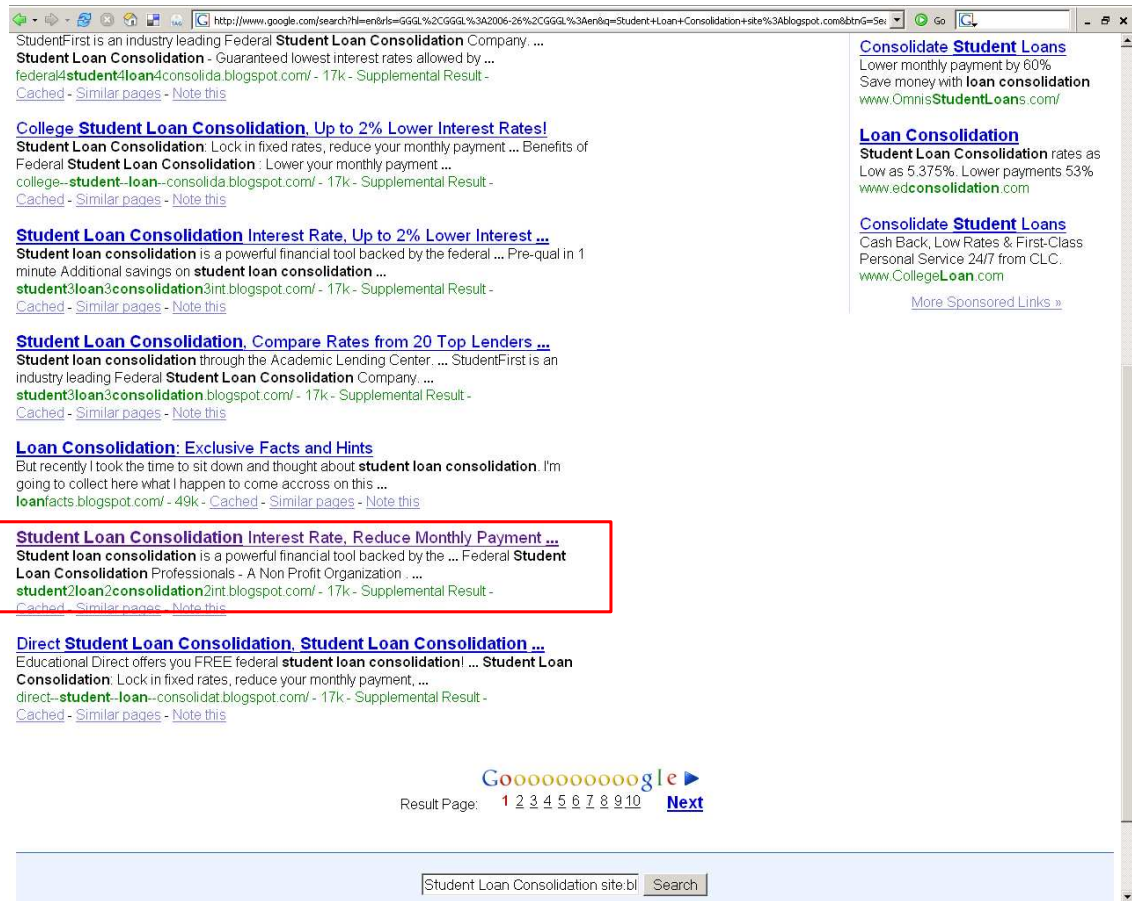


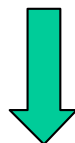
Figure II.5: Workflow: SERP Infiltration.

## II.A.4 Infiltrate SERPs

SERP is a Search Engine Results Page and the position of the doorway (see Figure II.5) on this page is a measure of the effectiveness of spamming. Spammers are known to continually experiment with multiple techniques to improve SERP positioning, until the doorways are eventually blacklisted by search engines or pulled down by hosts where they are hosted. While creating in-links for doorways on other trusted sources serves the dual purpose of infiltrating a search engine and boosting importance, a quicker technique commonly used is to infiltrate through ping servers, which are listened to by almost all major search engines.

Spammers sometimes use an alternate approach. The technique is now popularly known as AdSense arbitrage, owing to its resemblance to currency arbitrage, and involves attracting users through low cost leads (that the spammer pays for) and redirecting them towards more profitable leads. The relationship of this to web spam is however debated and questionable.

```
eval(unescape("%64%6f%63%75%6d%65%6e%74%2e%77%72%69%74%65%28%22%3c%73%63%72%69%70%74%20%73%72%63%3d%27%68%74%74%22%2b%22%70%3a%2f%2f%62%69%67%22%2b%22%68%71%2e%69%6e%66%6f%2f%73%74%61%74%73%32%2e%70%68%70%3f%69%22%2b%22%64%3d%31%32%26%67%72%6f%75%70%3d%32%27%3e%3c%2f%73%63%72%69%70%74%22%2b%22%3e%22%29%3b"))
```



document.write

```
("<script  
src='htt"+"p://big"+"hq.info/stats2.php?i"+"d  
=12&group=2'>");
```

Figure II.6: Workflow: Cryptic Javascript Redirect.

### II.A.5 Redirect users from Doorway

Users are directed to doorways from search results. To monetize from users spammers are required to redirect<sup>2</sup> them to mislead pages hosting ads and affiliate links. Many techniques have been used to achieve this. The earliest was the use of the “HTML META refresh”, that redirects the browser to mislead pages. Search engines noticed the increase in such illegitimate use and subsequently blacklisted pages using the META refresh.

To counter this, many other techniques followed, including cloaking, which continues to be widely used even today. This was followed by javascript based redirects. In response to search crawler’s ability to process lightweight scripts, spammers are now employing cryptic javascript (See figure II.6) which is now very commonly used, until future enhancements of search crawlers (spiders) that can better identify such spamming.

<sup>2</sup>It is also not uncommon to see doorways serve a dual purpose of acting as mislead pages.

The screenshot shows a web browser window with the URL <https://www.studentloanheadquarters.com/3891/display.asp>. The page header features the logo "Student Loan HEADQUARTERS" and a navigation menu with "Apply Now!", "How It Works", "FAQ", and "Contact Us". The main content area is titled "Consolidate your existing loans for as low as 4.5%" and includes a "Loan Information" section with several questions and dropdown menus. A callout box with a black border and white background is overlaid on the form, containing the text "LEADS" to affiliates highly profitable". Below the callout, the form continues with a "Social Security Number" field and a "Your Contact Information" section with a "First Name" field.

**Student Loan HEADQUARTERS**  
Need Money for School? Need to Lower your monthly payments?  
Student Loan HeadQuarters is the best place to help you get or consolidate your existing loans.

Apply Now! How It Works FAQ Contact Us

**Consolidate your existing loans for as low as 4.5%**

**Loan Information** Powered by SecureRights

What type of loan are you looking for?

How much student loan debt do you have?

Must have more than 1 loan

Have you ever consolidated your student loans?  Yes  No  
Must not have consolidated previously to qualify

Are any of your student loans in default?  Yes  No  
Must not be in default to qualify

Will you graduate in the next 6 months?  Yes  No  
Must have graduated or will graduate in 6 months to qualify

**How It Works**

- Complete online form
- Get a new loan
- Consolidate your existing loans
- Make one low monthly payment

**Consolidation**

- Reduce your payments
- One lender, one payment
- No credit checks, No fees!
- Apply in minutes
- Confidential and secure

**Help Us Verify Your Loan Information**

Social Security Number

Your Social Security Number is required by our lender and only be used to verify your loans with the US Department of Education National Student Loan Data System (NSLDS).

**Your Contact Information**

First Name

**New Student Loan**

Get your money fast  
Dont pay until after

**“LEADS” to affiliates highly profitable**

Figure II.7: Workflow: Monetize from Users.

## II.A.6 Monetize from Users

Spammers seldom monetize directly from users. They monetize through the leads they provide to websites that monetize from users. One such website is shown in figure II.7 on which a users finally ends up, as a result of redirection from a doorway to a mislead page, followed by a referral from the mislead page through contextual ads or misleads. While most contextual ad payouts are CPC (Cost Per Click), most affiliate payouts are CPA (Cost Per Action) based on a completed form, or purchase of merchandise (ring-tones, books, drugs, etc.)

The distinction between contextual advertisements and affiliate links are tending to become blurry by the day. Eventually, this distinction might cease to exist.



## II.B Related Work

Having discussed the traits of web spam through a workflow, we briefly survey existing work that has addressed the problem from different perspectives. We also argue the necessity of characterizing the problem that can guide future work.

The problem of spam on the Internet has largely been viewed from the e-mail and web spam perspective. While the focus of e-mail spam detection has been on using local models, the focus of web spam has so far been on using global (relational) models.

### II.B.1 Local Models

A *local feature* is one that is completely determined by the contents of a single web page, i.e., it does not require following links or consulting other data sources. A local model is one built only using local features. Local models can provide a quick assessment of the authenticity.

Since spam was initially seen in e-mails, research on spam detection was exclusively limited to spam in e-mails. Early efforts in e-mail spam detection used a rule-based approach [14]. As the effectiveness of rule-based approaches reduced, other techniques followed, mainly those based on using word and domain specific features [75][8], with models built using SVM [23], Naive Bayes, or simpler TFIDF [37] techniques. Some other techniques employed multiple models [36] in parallel.

More attention is now being given to the problem of web spam, though still not at the same level as e-mail. Web spam detection includes the use of local content-based features [22] and identifying statistical outliers [25]. Other combinational models [68] for web spam detection are also beginning to be explored. Since blog comment spam has been a problem for a while, some existing work [60], as well as commercial tools (e.g., Akismet,), address the problem. The issue of plagiarism, and automatically generated content is receiving some attention of late, and techniques based on text compression [18] are becoming effective.

While most of these existing techniques could be useful for detection of spam blogs, two issues are still to be addressed (i) how effective are these in the blogosphere, and (ii) what new features that exploit the nuances of the splogs can be effective.

## II.B.2 Global Models

A global model is one that uses some non-local features, i.e., features requiring information beyond the content of Web page under test.

Though HITS [43] and the offline PageRank [71] are viewed as techniques for ranking web-pages, one of the reasons for their popularity is their ability to separate out spam from authentic content. Both these techniques use implicit social ranking that exist on the Web, based on hyperlinks, and were quite effective when spammers used only content based spamming techniques. However as spamming the Web's link graph became common-place, new ranking techniques were required. Recent work on graph based algorithms are addressing this issue, the most popular being TrustRank [35, 33, 32]. Extensions to TrustRank, including topical trust rank [85], and other approaches that analyze link-farms [6][84] have also been proposed. To combat plagiarism and page stitching, detecting phrase level duplication on a global scale [26] have also been previously explored.

From the machine learning community, new techniques [64][62][73] that exploit multi-relational information sources [17] are being developed. These techniques are primarily being applied to web-page topic classification, and not directly to web spam detection. Popular approaches like link-based classification [57] could be effective for spam classification [44] as well. Approaches for iterative classification [63], and their dynamics [31] are also being studied, so are techniques based on learning graph structures through grammars [69], all of which could be effective.

In looking at global models, we want to capture the intuition that authentic blogs are very unlikely to link to splogs and that splogs frequently do link to other splogs, and view the entire problem from within the blogosphere. We present our findings in this thesis.

## II.B.3 Adaptive Techniques

E-mail spam classifiers were on one of the earliest systems which incorporated adaptive strategies [24][59][70][82]. The model underlying the spam filter learns from new e-mails that are manually tagged, continuously updating the model. Such updates are either local i.e., restricted to an individual e-mail inbox, or global i.e., based on aggregating tagged content from multiple (shared) inboxes.

Adaptive techniques are also being considered in adversarial classification settings. Adversarial classification [19] [56] is one where the classification task is treated as a game between the classifier and the adversary, where the adversary obfuscates discriminating features so as to by-pass the classifier. A related research direction is

that of concept drift, where the underlying distribution defining a concept changes either naturally (seasonal), or through adversarial techniques.

We approach the problem of adaptive classifiers from the following context of, how an ensemble of base classifiers enables the individual classifiers to be adapted on a stream of unlabeled instances. Note that this is different from co-training, which is typically applied in the context of weak base classifiers and concept drift, which is used on a stream of labeled instances. Arguably, we present the first results of using this technique in a real-world setting.

#### **II.B.4 Characterizing Problem Contexts**

Wordnet<sup>3</sup> defines “characterize” as “to describe or portray the character or the qualities or peculiarities”. Characterizing various aspects of systems, technology adoptions or applications have long been found useful in understanding and progressing the state of the art, more so on the Web. Some of the earliest efforts on Web characterization was initiated from the view of user navigational studies [11] on websites, that feeds into improving navigation and content design. As loads on the web infrastructure increased, characterizing web-page access patterns [2, 3, 55, 5, 4], both at web-servers and web-clients were found useful in improving the scalability and efficiency of the underlying infrastructure. To cater to requirements of web search engines, web document change and content evolution on the Web [53, 67, 13] have been studied, so as to understand its implication on page-revisit strategies. Studies on the structure of web [21, 10] have been effective in designing strategies for developing new, and improved page ranking algorithms.

The Web is a highly complex social network [65], and has been characterized from different structural perspectives. Characterizing this structure [10], and to some extent the content has been useful for web search engines [9], mainly in designing efficient crawlers and developing new ranking algorithms. With a similar motivation, but rather towards improving the state of the art of blog harvesters, we approach the characterization problem for the blogosphere.

Our characterization involves understanding the differences between blogs, and spam blogs or splogs. We study various aspects with the primary aim of designing effective algorithms to detect and eliminate spam. This includes local content and link-structure, all of which can be useful.

---

<sup>3</sup><http://wordnet.princeton.edu/>

## II.C Datasets

We have been collecting blogs for use in many different social media projects at UMBC. In this section we describe a subset of the corpus relevant to this work, give some basic statistics and discuss our labeled dataset.

Typically, there are three primary sources for collecting blogs: (i) processing streams of pings from blog ping servers (or blog hosts) that specify URLs of blogs with new posts; (ii) crawling directories which allow robot indexing of their site content; and (iii) using developer API's provided by popular blog search engines. Option (i) is the most common way of collecting blogs, evident from how blog search engines claim to index blogs. We followed techniques (i) and (iii) to collect new blog URLs. We briefly describe how we used these mechanisms in the creation of labeled datasets in 2005 and 2006.

### II.C.1 Dataset 2005

The goal of creating labeled datasets in 2005 was three-fold. We were developing a blog harvesting system that could harvest and index blogs. With this motivation we created three labeled subsets that would enable us to develop classifiers to identify blog homepages (BHOME), blog associated pages i.e., posts (BSUB) and splogs (SPLOG).

The Technorati<sup>4</sup> blog search engine provides an API<sup>5</sup> to access its data on blogs. We used this API (query limit of 500) over a period of four months (May-August 2005) to randomly sample Technorati's blog index by submitting queries for the freshest posts containing words picked randomly from an English dictionary. While we expected to collect only blogs in English, it was surprising that our collection also contained many non-English blogs. We believe this is due to some commonly occurring words in multiple languages. Based on this technique, we collected the URLs of approximately 500,000 unique blog home pages. Since we queried for the freshest posts our subset also included many splogs which are known to be very "live". A statistic on the blogs we collected and the top 10 hosting services (based on URL pattern match) where they are published is listed in table II.1.

Blog update ping services (and some blog search engines) accept pings from updated blogs and pass them on to blog search engines or other interested parties. This information is often republished as the most recently updated blogs as an XML document (e.g., "changes.xml"). We monitored these published files by fetching them at regular intervals from one of the of the popular ping mediators (<http://weblogs.com/>). Based

---

<sup>4</sup><http://technorati.com/>

<sup>5</sup><http://www.technorati.com/developers/>

	Popularity
blogspot	44%
msn	23%
livejournal	8%
aol	1%
splinder	1%
20six	1%
typepad	1%
blog	1%
fc2	1%
hatena	1%

Table II.1: Hosting statistics for blogs indexed by Technorati in 2005.

	Percentage
blogspot	39%
msn	23%
e-nfo	2%
travel-and-cruise	1%
lle	1%
find247-365	1%
typepad	<1%
blog	<1%
fc2	<1%
hatena	<1%

Table II.2: Hosting statistics for blogs pinging weblogs.com in 2005.

on this technique we collected around five million unique blog home pages over a period of five months (April-August 2005). Table II.2 gives a statistic on blog homepages we collected and the top ten domains where they (pings) come from.

Our motivation behind this analysis was to confirm that the data collected through Technorati is indeed a good sampling of the blogosphere. Results generally matched in the relative order of blog hosting popularity but not in their exact position. However, it is evident from table II.2 that update ping data is noisy. Hence we did not use ping data directly in our experiments, but rather indirectly during the creation of the negative training examples.

Due to the noisy nature of data collected from ping update services we used the Technorati index as our primary blog dataset. We dropped the top 30 hosts (e.g., blogspot, msn-spaces, etc.) from the dataset and uniformly sampled for around 3000 blog home pages. We then eliminated pages which were non-existent, whose size was less than 5KB or were written in Chinese, Korean or Japanese. We finally obtained 2600 blog homepages which we identified for use in the creation of our sample sets. Through a long manual process

category	percent
Legitimate	75%
Splog	25%
English	85%
Non-English	15%

Table II.3: Manual Labeling of blogs sampled from Technorati in 2005.

we tagged these blog homepages as one of legitimate or splog<sup>6</sup> and one of English or non-English. The distribution of these blogs is listed in table II.3.

It is clear from the table that even a popular search engine like Technorati has a high percentage of splogs. In addition some of the pages identified as blogs by Technorati were actually forums (with syndication feeds). Also note that though our Technorati queries used English words, results still had a good number of non-English blogs. We then randomly sampled local links from these blogs which are non-blog home pages and post pages (blog subpages). We sampled to get a sample of around 2,100 subpages. We did not manually verify these samples for correctness.

We then went about creating negative examples for two of our subsets (BHOME, BSUB). In training classifiers, generating negative samples has traditionally been both time consuming and erroneous. While a simple option is to sample a web search engine randomly, we employed a slightly more intricate approach. We extracted all external links from our positive dataset. This set consisted of links to other blogs (within the blogosphere) and the rest of the Web. The number of links we extracted was in the order of half a million. A completely manual process to identify non-blog pages among these links would have been next to impossible.

Hence, from these extracted outlinks we eliminated those URLs that we knew were blogs. We compared host names of extracted URLs against the host names of URLs we had collected through Technorati and update ping services. This was extremely useful since we had collected around five million unique blog homepages and consequently a high number of unique blog hosts (both blog hosting services and self hosted). For example, since we now know that pages with the domain “blogspot”, “instapundit” etc., are blogs, we eliminated all such URLs from the negative set.

After the process of elimination of URLs in the negative set we were left with around 10,000 URLs out of which we uniformly sampled a smaller number. We then manually validated our process to generate a total of around 2600 negative samples. Our data creation process also made sure that negative samples had a sizeable number of non-English pages commensurate with the numbers in positive samples.

---

<sup>6</sup>We made sure we incorporate some of the common splog techniques listed by a blog search engine - <http://blog.blogpulse.com/archives/000424.html>

Towards generating positive samples for SPLOG we picked those blogs that we had manually identified as splogs, around 700 in number. This does imply that our splog dataset does not include blogs from the top hosting services, the ones which were eliminated them from our base dataset. We made this choice since it eased the process of manual labeling. Further investigation suggested that the nature of splogs were similar in top blog hosting services, and hence our choice should not significantly affect classification accuracy. We randomly sampled for 700 authentic blogs from the remaining 1900 blog home pages to generative negative samples for SPLOG.

With all of the above techniques we created three data sets of samples:

1. **BHOME:** (blog home page, negative) consisting of (2600 +, 2600 -) samples for a total of 5200 labeled samples.
2. **BSUB:** (blog all pages, negative) of (2100 +, 2100 -) samples.
3. **SPLOG:** (blog home-page spam, blog home-page authentic) of (700 +, 700 -) samples.

From here on we refer to these dataset as BHOME, BSUB and SPLOG-2005.

## II.C.2 Dataset 2006

To support our blog related projects and to better understand the evolution of classifier performance we created a new dataset in 2006, by sampling at an update ping server (<http://weblogs.com>). The sampling process is depicted in figure II.8, and the labeled dataset created will from now on be identified as SPLOG-2006.

SPLOG-2006 was created from pings aggregated in October 2006, a total of 75 million unique pings. As shown in figure II.8, the samples were arrived at as follows. To generate the negative samples (authentic blogs) we separated out all blogs that were part of bloglines subscription list, collected as part of Feeds That Matter [38]. Amongst these blogs, we sampled for around 750 authentic blogs to obtain the negative samples. To obtain positive samples, we first eliminated blogs from myspace which overwhelmed most of this subset. We then eliminated non-blogs in this collection using our classifiers that identify blogs [47]. We then sub-sampled for around 2000 blogs. To simulate the working of a typical blog harvester, we limited this sub-sampling to blogs that featured full syndication. Manually labeling all these URLs resulted in 750 splogs.

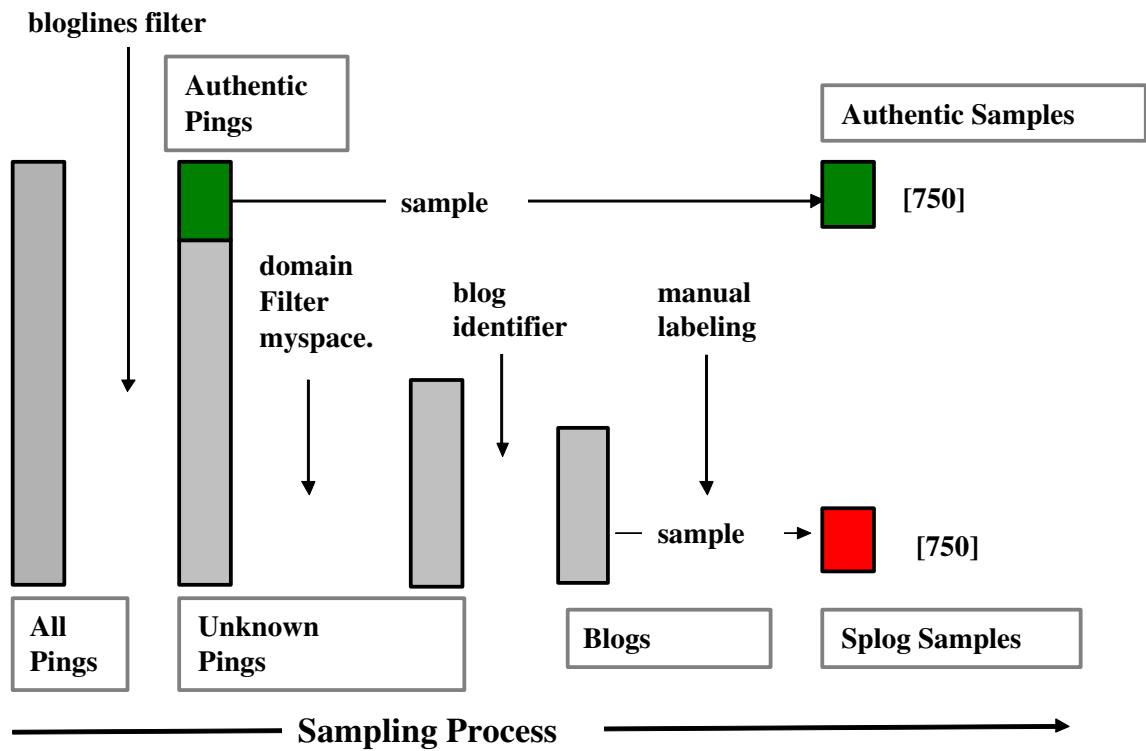


Figure II.8: Sampling for SPLOG-2006.



We finally reached at 750 positive samples (splogs) and 750 negative samples. Unlike SPLOG-2005, this dataset also featured feeds associated with all samples. Note the SPLOG-2006 is potentially more noisy than SPLOG-2005, given that SPLOG-2005 has gone through a preliminary filtering at a blog search engine (Technorati).

## II.D META-PING

An underlying system that motivated most of the work presented in this thesis is a META-PING system we developed to filter out splogs from a stream of update pings, or from URLs that are served to it as input. In addition to serving as a mechanism to test newly developed techniques, it also enabled us to draw from experiences that guided continuing work in this thesis. We also distributed and deployed this system at multiple locations (a blog search engine, within UMBC, two academic partners) and drew from these experiences. We discuss the details of the META-PING system in this section.

The meta-ping service feeds off update stream from ping servers and appends additional metadata to each ping. One such metadata is a real-valued score of *authenticity* between zero and one, where zero identifies a splog and one an authentic blog. A black-box view of the meta-ping server is shown below.

INPUT:

```
name='`UMBC Ebiquity Blogger`'  
url='`http://ebiquity.umbc.edu/blogger`'  
when='`February 21, 2007 04:33 PM`'
```

OUTPUT:

```
name='`UMBC Ebiquity Blogger`'  
url='`http://ebiquity.umbc.edu/blogger`'  
when='`February 21, 2007 04:33 PM`'  
authenticity='`1`'
```

Given a ping identified by its name, url and time-stamp<sup>7</sup>, information on *authenticity* is computed. Additional metadata including language, feed location etc. is made available on a need-to basis. The overall system is shown in Figure II.9. The system operates on a stream of pings from weblogs.com<sup>8</sup> and filters are serially

<sup>7</sup>See <http://weblogs.com/api.html>

<sup>8</sup>150K pings per hour on an average as of February 2007

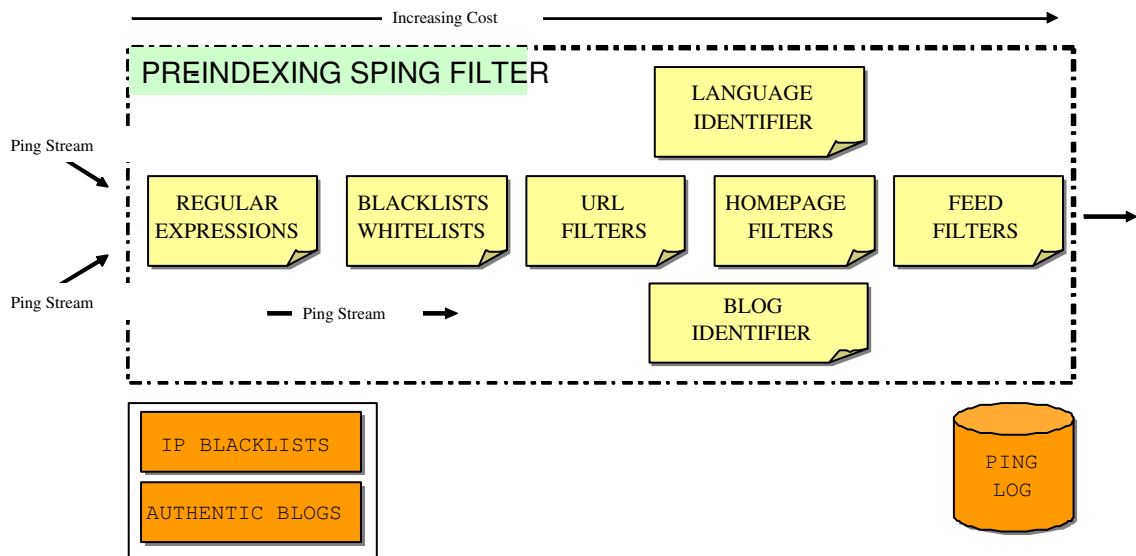


Figure II.9: META-PING System Architecture.

used to decide whether to stop or let through a ping. As of this implementation, each of these filters work independent of each other.

We briefly discuss individual filters that make up the system.

### II.D.1 Blacklist/Whitelist Filtering

Using a catalog of spam domains, blacklisted IP addresses and known authentic blogs forms the core component of this filter. The approach of using blacklisted IP addresses [15] has been found effective previously. However, since many splogs are hosted on blog hosting services, IP blacklists are cautiously used. Pings from all unknown sources (authentic or spam) are evaluated in subsequent steps.

### II.D.2 REGEX filtering

Regular Expression based filters have been used in e-mail spam filters due to its simplicity. We employ a set of regular expressions, based on the URL and correlations between URL and name of a ping. These expressions are carefully tested (using our existing whitelists) to avoid false positives, and are rather conservatively used.

### II.D.3 URL Model Based Filtering

This filter is the last in the series characterized by no web fetches. Search engines attribute additional relevance to pages whose URL string closely matches a user query, prompting spam URLs to feature query specific keywords. This rich context of the URL string is utilized in this step. We use character n-gram based segmentation<sup>9</sup>, with models trained using SVMs.

### II.D.4 Blog Home-Page Based Filtering

The fourth step in filtering is based on analyzing content of the blog-homepage. Though this step involves a web-fetch per URL, it offers many advantages. (i) It enables identifying the ping source as a blog or non-blog using language independent techniques. Ping servers are easy inlets to indices of search engines, commonly exploited by spammers to notify the existence of non-blog pages. (ii) It enables identifying the blog's authoring language, which in turn facilitates the use of either language independent or language dependent models. (iii) It enables the use of existing splog detection models based on text and outgoing URL tokens from the page.

### II.D.5 Feed Based Filtering

In addition to developing models over text and hyperlinks in a blog, HTML tags used across posts can be used to identify signatures of readily available splog software [30]. This requires the use of post content, available only in syndication feeds, and involves two web-fetches per URL. In addition, this form of filtering enables identifying life-time of a blog, measured by the number of posts hosted on the blog.

Using a stateless operation across pings and a decentralized implementation this system is highly scalable and configurable. Models at step three, four and five are trained to predict probabilities and thresholds to eliminate splogs at each of these steps is chosen on a per-project/deployment basis.

## II.E Supervised Machine Learning

Most machine learning algorithms use a vector representation of documents (web pages) for classification. Such vectors are commonly termed as feature vectors and represented by a set of features as  $\{ f_1, f_2 \dots f_m \}$ . Every document has a specific signature for these features represented by a weight as  $\{ w(f_1), w(f_2) \dots w(f_m) \}$

---

<sup>9</sup>Results from this work yet to be reported.

}. Traditionally web page classification has required using different types of features (feature types), selecting the best (feature selection) and encoding their values in different ways (feature representation).

One of the most common feature types is the bag-of-words where words appearing on a page are used as features, and is quite effective for topic classification. In addition, other features like content of specific tags on a page (anchor text, title) and customized features (most popular for email spam detection) are also used either independently or in conjunction with bag-of-words. While these traditional features are quite effective for topic classification, their applicability to categorization problems in the blogosphere has not yet been completely analyzed.

Based on a vector representation of documents, Support Vector Machines (SVM) [8] are widely used for text (and hyper-text) classification problems. They have been shown to be very effective in many domains including topic classification, sentiment classification [72] and email spam classification [23]. Based on this universality of SVMs for categorization and its relative efficiency and effectiveness [87] we use SVMs for all our classification tasks.

## Chapter III

# DETECTING SPAM BLOGS

In this chapter, we discuss the utility of features for splog detection. As with any form of webpage classification different features could be effective. Our aim is not to evaluate a catalog of all possibilities, but to draw features based on their effectiveness in webpage classification, and to discover new features that could be effective in this domain.

We separate feature identification efforts into:

1. **Local Features**, that are completely based out of information local to a blog i.e., does not require following links or consulting other data sources.
2. **Derived Features**, that are derived from local features.
3. **Relational Features**, that are computed from the relationship between webpages, and involve following links across pages.

We view local features based on the cost associated with the classifiers they support, measured by webpage fetches. We hence categorize classifiers as URL based classifiers (no page fetch), blog home-age based classifiers (one fetch), and feed based classifiers (two fetches). We first report experimental results based on local features.

All results are based on evaluation against two independently sampled datasets, SPLOG-2005 and SPLOG-2006, using ten-fold cross validation. It should not be surprising that SPLOG-2006 in general provides better classifier performance given that it was sampled at a ping server. SPLOG-2005 having been sampled through a blog search engine features splogs that have already passed one set of filters. Results are also based on using

the top fifty thousand features selected using frequency (other feature selection techniques did not improve results significantly), linear kernel and binary feature encoding. We report precision/recall and area under the curve for SVMs using default parameters, and a linear kernel. We also report results from Naïve Bayes Classifier for completeness. All experiments use the Weka [83] and libsvm [12] toolkits. Further, we report the top discriminating features based on weights associated by the linear SVMs, a technique known to be quite effective [61].

### III.A URL Based Classifiers

A classifier using only the URL (without fetching the associated webpage) is sufficiently effective significantly relieves computational costs. We would like to capture the following intuition. Search engines associate higher ranking to pages with a URL that is contextually similar to the search query. Most spam pages hence tend to have contextually rich tokens in the URL. For instance, a spam page on “loan consolidation” generally features this text in the domain, sub-domain or path of the URL.

The natural approach to generating features from a URL is by tokenizing the URL on “:”, “/”, “-”, “\_”, “?” and “=”. But this technique taken alone suffers from certain limitations. Most spam URLs generally combine keywords, for instance “http://loanconsolidation.biz” or “http://1loanconsolidation.biz”, which (i) obfuscates their technique from less aggressive spam filters, and (ii) expands the number of domain name permutations, while continuing to render them effective for search result ranking. To address this, we use a combination of 3-grams, 4-grams and 5-grams (See [20]) obtained from the natural tokenization. “x-grams” are arrived at by using “x” adjacent characters in a token (word).

Precision, recall and F-1 measures are shown in table III.1 and figure III.1. Their values for SPLOG-2005 is significantly lower from SPLOG-2006, given that blog search engines use many heuristics to reject spam blogs based on URLs. Though not documented formally, it is well understood that the appearance of hyphens and the length of the URL are two common spam signals. Such URLs are absent in SPLOG-2005, but present in SPLOG-2006, and are easier to detect using machine learning based models, evident from performance values for SPLOG-2006. Interesting discriminating features are observed. While tokens “edu” and “web1” feature prominently among authentic blogs, tokens “hot”, “info”, “loans” feature among splogs. Note that “loans” is a keyword in a profitable advertising context, “edu” domains are less susceptible to hosting splogs, whereas the inexpensive “info” domains host a high number of splogs. This last characteristic has prompted

a few blog search engines to completely blacklist the “info” domain.

Given the significantly low cost, the utility of webpage URL for classification has received some attention [41] in the past. The use of URL tokenization for splog detection was initially proposed by Salvetti et. al [77, 66] through a novel URL segmentation (tokenization) approach that provides improved performance over natural tokenization.

## **III.B Home-Page Based Classifiers**

In this section we discuss features extracted out of blog home-pages. Blog home-page is typically associated with one page fetch, and provides several advantages that include, (i) hosting a summary of the blog through recently created posts, (ii) capturing blog-rolls and link-rolls which are highly personalized in authentic blogs, and (iii) capturing historical information like archives which hint to the lifetime of the blog. The utility of these will be evident when observing discriminating features in the rest of this section. However, a potential disadvantage of using only a blog home-page is that knowledge about the blog life-cycle is not available during the classification process, which leads to erroneous classifications of newly created blogs and splogs. In the rest of this section we discuss features at the blog home-page level that could be effective.

### **III.B.1 Words**

The most common, popular, and effective feature type used for text classification tasks is the bag-of-words, where words occurring on page are used as features. The content of a page is quite naturally the single most important attribute that is typical of the topic or genre of text. The same holds for splogs. Since splogs are generally used as doorways, and are required to be indexed by search engines, they feature text that increases its relevance for queries in profitable contexts. By using a bag-of-words feature type we want to capture this intuition.

We do not use stemming and stop word elimination to better capture the nuances associated with blog post authoring, which is highly personal. Moreover, SVMs are known to be robust even in the absence of such preprocessing. Results from using bag-of-words are shown in table III.2 and figure III.2. Clearly, this feature type continues to be effective for splog detection. Upon analyzing discriminating features, it turns out the classifier was built around features which the human eye would have typically overlooked. Blogs often contain content that expresses personal opinions, so words like “I”, “We”, “my”, “what” appear

commonly on authentic blog posts. To this effect, the bag-of-words model is built on an interesting “blog content genre” as a whole. In general, such a content genre is not seen on the Web, which partly explains why spam detection using local textual content is not all that effective there. Further, other discriminating features are also interesting. Authentic blogs feature posts from multiple months, whereas splogs feature posts only from the month when the sampling was performed, namely “oct” indicating the short-lived nature of splogs. In addition, the presence of “copyright” among splogs confirms that most of splog content is generally plagiarized from other web sources, and the keywords “sponsors” and “ads” confirm the economic incentive behind splogs.

Most of the work on web spam detection has emphasized on link-based detection techniques (relational techniques). The earliest work [22] the exclusively evaluated the bag-of-words approach reports AUC values close to 0.9, slightly lower than their values for splog detection. Given that the emphasis of splog detection is fast online detection, classifiers using bag-of-words can provide an extremely effective solution.

### **III.B.2 Wordgrams**

In an adversarial classification context, a common obfuscation technique is the “word salad”, which gained in popularity as an attack against Bayesian e-mail spam classifiers. The technique involves incorporating random words that are indicative of authentic content, which in the context of spam blogs is the use of discriminating features referred to earlier. While quite effective against e-mail spam filters, this technique is less effective against web spam filters where “word salad” reduces the TFIDF score for profitable contextual keywords. Nonetheless, it does find limited use, and motivates the evaluation of this feature type.

Though the most effective solution against the “word salad” is sentence level natural language processing, this is computationally expensive. Hence, the use of shallow natural language processing through wordgrams, or more generally word-n-grams, where “n” represents the number of adjacent words used as features have been found effective.

We used word-2-grams as feature type, with results shown in table III.3 and figure III.3. Note that this feature is not as effective as bag-of-words, but useful nonetheless. Interesting discriminative features were observed. For instance, text like “comments-off” (comments are usually turned off in splogs), “new-york” (a high paying advertising term), “in-uncategorized” (spammers do not bother to specify categories for blog posts) are features common to splogs, whereas text like “2-comments”, “1-comment”, “i-have”, “to-my” were some features common to authentic blogs. Similar kind of features rank highly in the word-3-gram model.



Note that wordgrams based technique is not useful in the case of “paragraph salad”, where an entire paragraph of authentic content is incorporate. Further, unlike bag-of-words the feature space of word-n-grams increases exponentially with n. The use of word-n-grams has also been explored in the context of web spam detection [68] with good results.

### **III.B.3 Anchor Text**

Anchor text is the text that appears in an HTML link (i.e., between the `<a . . . >` and `</a>` tags.) In addition to the well-known PageRank metric that associates topic independent ranks to pages based on hyperlinks, a supplementary technique is quite commonly used to find topic-specific importance. This is enabled through analyzing anchor text of in-linking pages. To inflate topic sensitive ranking of spam pages, spammers typically create farms of pages with anchor text rich with keywords in the compromised profitable context. This hints to the possibility of using “only” anchor text on a page as features. Note that this is different from using anchor text on in-links, an effective technique in topic classification.

We hence evaluate the bag-of-anchors feature, where anchor text on a page, with multiple word anchors split into individual words, is used as feature. Results are depicted in figure III.4 and table III.4. Note that while this technique is not as effective as bag-of-words (which subsumes this feature type), it still can provide significant merit to filters that use anchor text alone. Interesting discriminating features were observed. Authentic blogs feature text that point to customized pages, linking to flickr and other sites, in addition to text associated with archive content that denote the life-time of a blog. The lack of profitable contextual keywords among splogs leads to one possible explanation. Splogs are more likely to be used to create doorway pages, than as members of a link-farm.

As far as we know, the use of anchor text on the page being classified as a feature has not been explored earlier.

### **III.B.4 Outlink**

Out-links are all hyperlinks hosted on a page, we generalize it to include links both internal and external to a page. This feature choice is motivated by the fact that many splogs point to the “.info” domain, whereas many authentic blogs point to well known websites like “flickr”, “technorati” and “feedster” and strongly associated with blogs. This feature choice can also be thought of as a logical extension of the URL based classifier referred to in the earlier section.

We term these features as bag-of-urls, which for simplicity was arrived at by tokenizing URLs using “.”, “/”, “-”, “\_”, “?” and “=”. Results are shown in figure III.5 and tab III.5. Clearly, the listed top features supports our intuition, more so in the SPLOG-2006 dataset. While the domain suffix “org” is a clear signal of authenticity, suffixes of “info” and “com” appear to be signaling splogs. Interesting to note is the appearance to “google” and “technorati” as top features among splogs, confirming our speculation from our case studies [28].

Discriminating features also appear to indicate that an n-charactergram approach might lead to a better classifier, a feature (and segmentation approaches) that could potentially be explored in future work. Our use of bag-of-urls is the first such use for any classification task, and clearly is quite effective for splog detection.

### **III.B.5 Character Grams**

Character grams [20] are known to be an effective feature for text classification, more so when the task spans multiple languages. Since our evaluation of feature types do not use the preprocessing step of stemming, an evaluation using this technique is particularly important.

Results are shown in figure III.6 and table III.6. The performance of this feature type is slightly lower than that of bag-of-words taken alone. Closer examination of the discriminating features clearly shows the effect of redundancy in features, an issue common to ngram features, suggesting the use of more advanced feature selection technique.

### **III.B.6 HTML Tags**

Throughout our exploration of the feature space, we were continuously engaged in characterizing the problem of spam blogs, by understanding the mechanisms used in creating them, one of which is the availability of splog creation software [30]. Though we look at this aspect in more detail in a subsequent chapter, we provide an intuition as to why this led to the exploration of a new feature space, based on the HTML stylistic properties. Splog creation tools use a simple workflow, that begins with plagiarizing text from other sources, the key here is that the term “text” lacks stylistic information. Splog software can hence easily be stereotyped based on the HTML tags it sprinkles around and within this text, a case not common to the varied use of such tags in authentic blogs. We capture this intuition by using this feature type.

Results from using HTML tags is shown in figure III.7 and table III.7, which turns out to be a very effective feature. Analyzing top features leads to interesting insights. “blockquote”, “img”, “embed” are

common to authentic blogs, whereas “b” that stands for bold text is quite common to splogs. Other tags common to splogs capture the stereotypes of various splog creation tools.

Note that this feature has a significantly low overlap with other features discussed so far, and is largely context independent. Though the effectiveness of this technique is clear, we believe variances of this technique can provide significant benefit in the long term. We have not come across any prior work that looks at page classification exclusively using HTML tags.

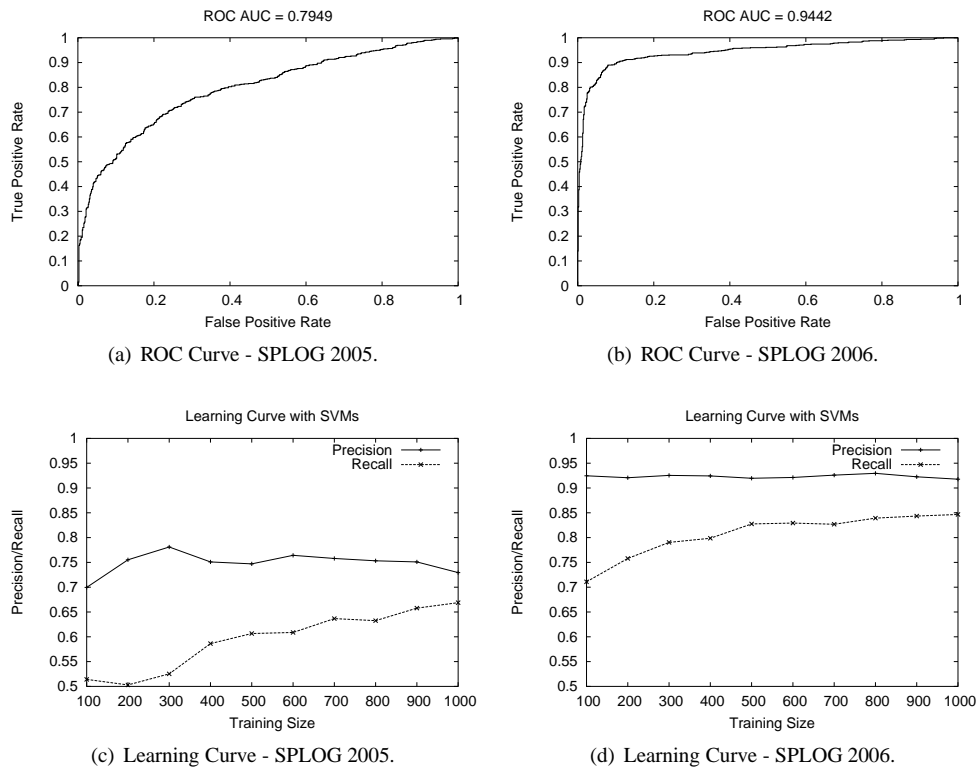


Figure III.1: URL n-gram Classifier Evaluation.

(a) Performance - SPLOG 2005.

	P	R	F1
SVM	0.70	0.70	0.70
NB	0.70	0.67	0.69

(b) Performance - SPLOG 2006.

	P	R	F1
SVM	0.92	0.88	0.90
NB	0.82	0.90	0.85

(c) Top Features - SPLOG 2005.

Authentic
fif, sig, yww, lee enk, mod, hop, dae ose, edu, mode, bab baby, aby, ile, blu evie, file, evi, hat
Spam
nhg, vkq, hot, mat chao, ree, urs, herb cha, she, shev, hev ool, karl, rlz, des info, ate, inf, ies

(d) Top Features - SPLOG 2006.

Authentic
law, xre, org, cha hds, ibn, bnl, ibnl bnliv, bnli, ibnli, clau poo, rea, log, lau aus, rma, webl, weblo
Spam
htm, imv, nfo, info inf, car, eac, each abe, ach, blogs, job sta, grac, grace, ogs logs, star, ace, loan

Table III.1: URL n-gram Classifier Evaluation.

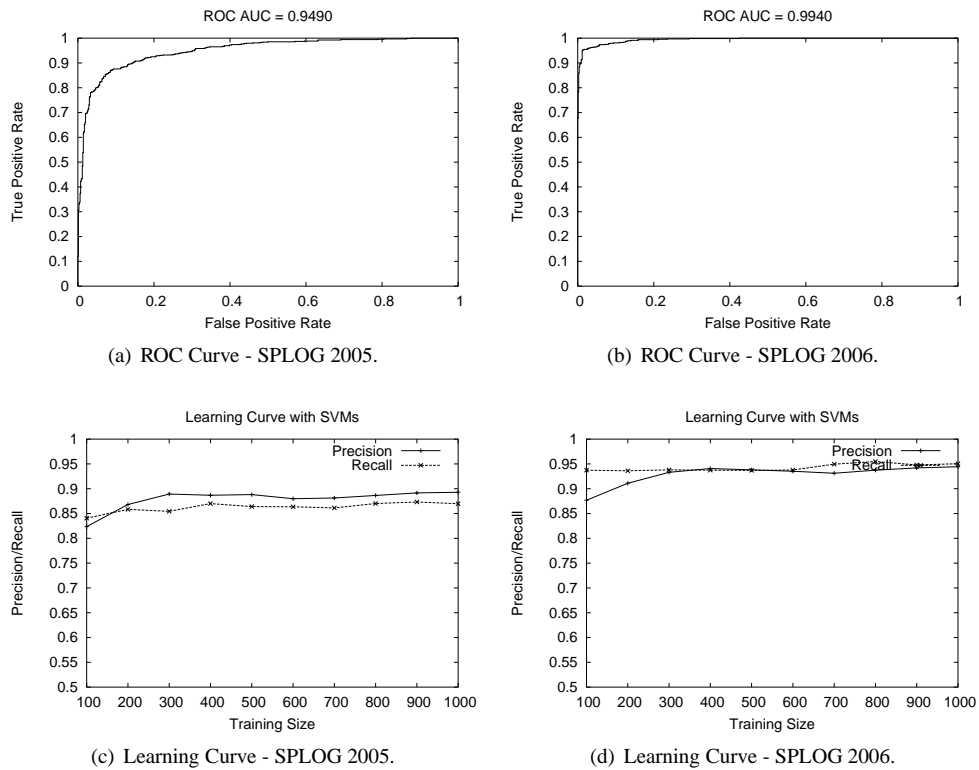


Figure III.2: Words Classifier Evaluation.

(a) Performance - SPLOG 2005.

	<b>P</b>	<b>R</b>	<b>F1</b>
SVM	0.90	0.88	0.89
NB	0.82	0.85	0.83

(b) Performance - SPLOG 2006.

	<b>P</b>	<b>R</b>	<b>F1</b>
SVM	0.95	0.95	0.95
NB	0.92	0.92	0.92

(c) Top Features - SPLOG 2005.

<b>Authentic</b>
my, we, s, new, this, please, what, org, log, linux gallery, words, paper, jane, political, web, september, reseller, flickr, open
<b>Spam</b>
news, your, on, find, previous, info, uncategorized, best, information, top, posted, com, laquo, related, business, articles, august, may action, looking

(d) Top Features - SPLOG 2006.

<b>Authentic</b>
location, april, february, june march, pm, november, creative friday, july, may, thursday link, article, fun, wednesday privacy, down, today, read
<b>Spam</b>
by, oct, posts, technorati tag, edit, at, sitemap as, to, start, info friendly, and, free, news buy, sponsors, copyright, ads

Table III.2: Words Classifier Evaluation.

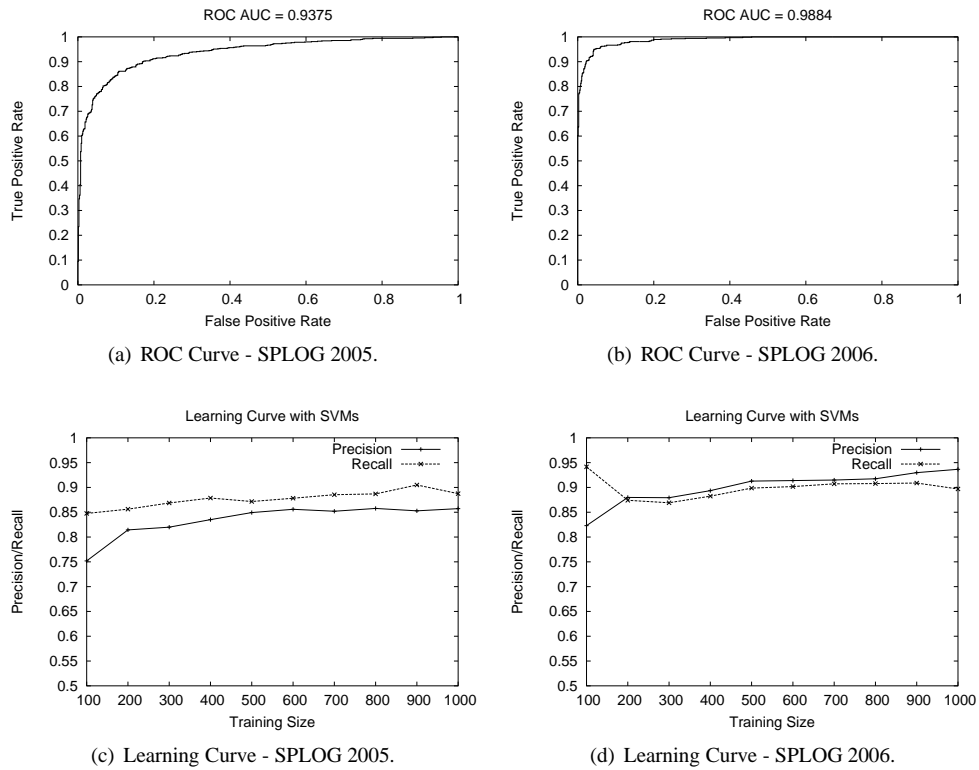


Figure III.3: Wordgram Classifier Evaluation.

(a) Performance - SPLOG 2005.

	<b>P</b>	<b>R</b>	<b>F1</b>
SVM	0.86	0.86	0.86
NB	0.80	0.83	0.82

(b) Performance - SPLOG 2006.

	<b>P</b>	<b>R</b>	<b>F1</b>
SVM	0.93	0.92	0.92
NB	0.90	0.92	0.91

(c) Top Features - SPLOG 2005.

<b>Authentic</b>
nnbsp-nbsp, personal-web, s-personal please-read, read-my, are-my a-new, nnbsp-blog, blog-nbsp, about-me, here-are, search-this september-august, i-have, s-blog
<b>Spam</b>
to-us, at-am, uncategorized-no comments-off, linking-to, com-archives site-index, self-publishing, writer-s archives-august, the-internet, in-den new-york, the-best, many-people

(d) Top Features - SPLOG 2006.

<b>Authentic</b>
pm-nbsp, me-do, profile-links this-post, comments-links, am-nbsp to-this, previous-posts, nnbsp-about nbsp-friday, the-new, nnbsp-thursday links-to, post-nbsp, march-april
<b>Spam</b>
technorati-tag, recent-posts, comments-nbsp tuesday-october, am-comments, friendly-blogs tue-oct, my-favorites, mon-oct original-post, blog-tag, sun-oct sponsors-ads, thu-oct, ads-recent

Table III.3: Wordgram Classifier Evaluation.

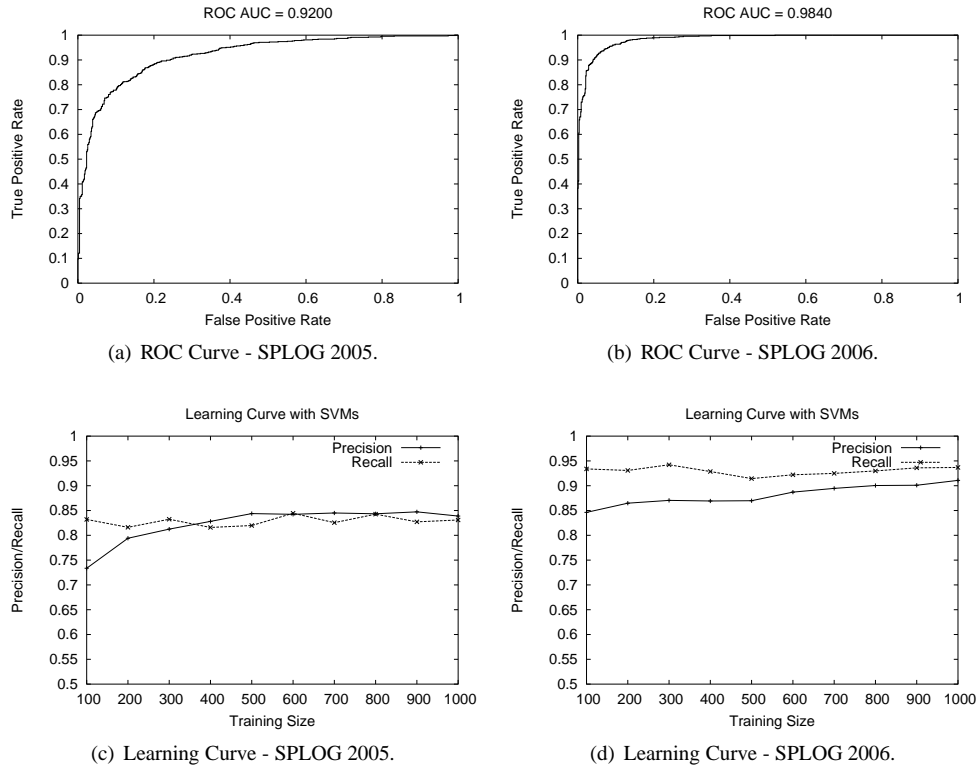


Figure III.4: Anchor.

(a) Performance - SPLOG 2005.

	<b>P</b>	<b>R</b>	<b>F1</b>
SVM	0.84	0.85	0.85
NB	0.83	0.82	0.82

(b) Performance - SPLOG 2006.

	<b>P</b>	<b>R</b>	<b>F1</b>
SVM	0.92	0.94	0.93
NB	0.88	0.56	0.68

(c) Top Features - SPLOG 2005.

<b>Authentic</b>
greymatter, rant, monk, chapitre terrorism, comment, jane, the postcount, permalink, archives, disclaimer flickr, trackback, journals, about s, space, report, random
<b>Spam</b>
read, chapter, revisionaryjen, generation ii, laquo, lost, more biz, to, top, jaguar soulessencehealing, now, used, directory august, free, town, an

(d) Top Features - SPLOG 2006.

<b>Authentic</b>
december, site, about, flickr july, links, august, this september, november, memories, link here, february, march, projects archives, photos, email, article
<b>Spam</b>
prop, start, comments, nbsp by, edit, google, for sitemap, and, oceanriver, freedom search, hawaii, university, xhtml news, to, mmorpqsource, superforum

Table III.4: Anchor.

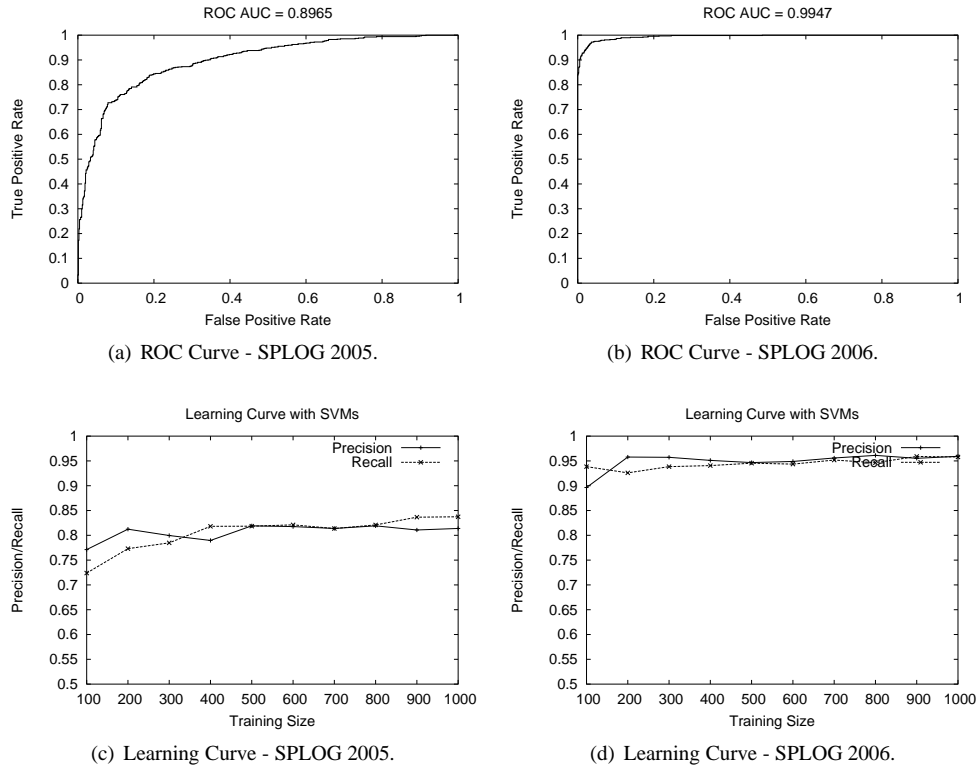


Figure III.5: Outlook.

(a) Performance - SPLOG 2006.

	<b>P</b>	<b>R</b>	<b>F1</b>
SVM	0.81	0.83	0.82
NB	0.79	0.81	0.80

(b) Performance - SPLOG 2006.

	<b>P</b>	<b>R</b>	<b>F1</b>
SVM	0.95	0.96	0.96
NB	0.95	0.57	0.71

(c) Top Features - SPLOG 2005.

<b>Authentic</b>
darklevel, lcrwtvl, malumphy alerts, marriagecounselingnews, asteriski globaldreamers, blogthings, chelf thao, house, fishspeaker craftybutterfly, words, myhorribletrash
<b>Spam</b>
riversidecaliforniaonline, pairingwine, myluckyskills focusresourcesinc, jakstar, boychat soulessencehealing, mine, qxt frnews, percar, gerboni u, bwh, directory

(d) Top Features - SPLOG 2006.

<b>Authentic</b>
rudayday, weblog, archives October, august, id sundaymornings, email, jpg mailto, september, photos brin, org, of
<b>Spam</b>
info, com, prop cessna, technorati, solution page, proactiv, mybeautyadviceblog tag, www, profile comment, google, post

Table III.5: Outlook.



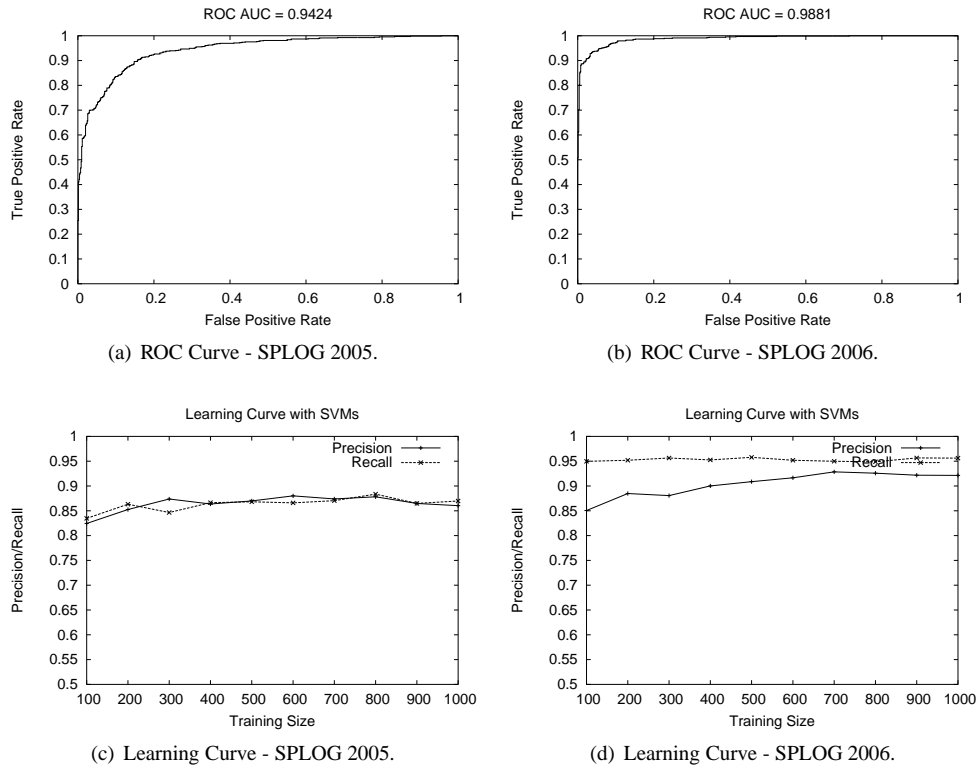


Figure III.6: Chagram.

(a) Performance - SPLOG 2005.

	P	R	F1
SVM	0.86	0.87	0.87
NB	0.78	0.83	0.80

(b) Performance - SPLOG 2006.

	P	R	F1
SVM	0.93	0.93	0.93
NB	0.86	0.87	0.86

(c) Top Features - SPLOG 2005.

<b>Authentic</b>
lle, blo, gal, see ami, thin, add, pleas plea, woul, son, lou inu, gall, flic, gue jan, galle, wha, erenc
<b>Spam</b>
new, ver, rti, bes oste, poste, aqu, ail prev, inf, ran, hei icl, man, pro, fin tra, itie, rov, che

(d) Top Features - SPLOG 2006.

<b>Authentic</b>
oca, ocati, ocat, loca apr, locat, apri, loc catio, marc, jun, cati bru, vem, riv, jul feb, lic, ebr, vemb
<b>Spam</b>
pos, ost, post, blo lin, new, tio, pro rec, edi, com, ssn essn, rat, ess, chnor honor, hnora, norat, ent

Table III.6: Chagram.

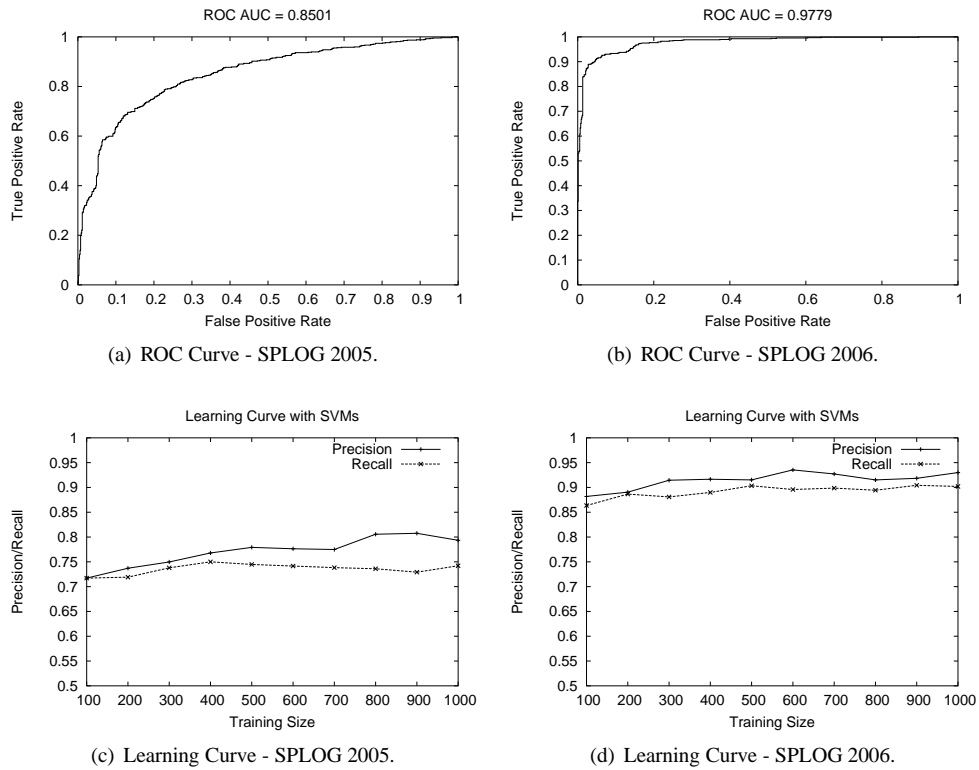


Figure III.7: Tag

(a) Performance - SPLOG 2005.

	<b>P</b>	<b>R</b>	<b>F1</b>
SVM	0.79	0.74	0.77
NB	0.75	0.72	0.74

(b) Performance - SPLOG 2006.

	<b>P</b>	<b>R</b>	<b>F1</b>
SVM	0.94	0.91	0.92
NB	0.94	0.85	0.90

(c) Top Features - SPLOG 2005.

<b>Authentic</b>
dt, marquee, table, pre wbr, embed, img, s noembed, warning, ahem, link background, no, del, blockquote basefont, description, i, ins
<b>Spam</b>
entrytitle, script, tt, bgsound dl, nyt, byline, li tr, td, nobr, content hr, mainorarchivepage, state, meta tblog, font, activated, status

(d) Top Features - SPLOG 2006.

<b>Authentic</b>
blockquote, sup, html, mainorarchivepage dt, del, span, img tag, th, option, select noscript, em, strike, ol big, o, noembed, embed
<b>Spam</b>
link, h, acronym, d marquee, thead, tfoot, fieldset dl, b, doctype, street center, abbr, title, a head, meta, description, nobr

Table III.7: Tag.

### III.C Derived features

In the final set of evaluation using local models, we explored the utility of language models and other heuristics as shown in Table III.8. Unlike binary features used in previous experiments, all these features were encoded using numeric floating point values between 0 and 1, except for feature number 11 whose value could potentially be greater than 1. Values for the first five features were based on extracted named entities using the ling-pipe<sup>1</sup> toolkit, and the ratios were computed using the count of all words on the page. Splogs usually promote products or websites (usually named entities) by repeating such named-entities many times within a page. This is a standard exploit on TFIDF indexing employed by search engines.

We also experimented with other heuristic based features. The repeatability of text, URLs and anchors on splogs prompted us to use compression ratio as a feature. The intuition being that such pages have low compression ratio. Compression ratio was computed as the ratio of the size of deflated to inflated size for each of the feature types. To capture the notion of splogs containing a higher amount of anchor text and URLs as compared to the size of the page, we computed the ratio of their character size to the character size of the blog home-page as a whole.

In addition we also used the ratio of distinct URLs (and Anchors) on a page divided by all URLs (and anchors) to check for repeating URL's and anchor text, which is quite common in splogs. To evaluate the hypothesis that hyperlinks on splogs feature many URLs with hyphens, we employed the ratio of number of hyphens to number of URLs as a feature.

<i>No.</i>	Feature Description
1	Location Entity Ratio
2	Person Entity Ratio
3	Organization Entity Ratio
4	Female Pronoun Entity Ratio
5	Male Pronoun Entity Ratio
6	Text Compression Ratio
7	URL Compression Ratio
8	Anchor Compression Ratio
9	All URLs character size Ratio
10	All Anchors character size Ratio
11	Hyphens compared with number of URLs
12	Unique URLs by All URLs
13	Unique Anchors by All Anchors

Table III.8: Effectiveness of Specialized Features.

<sup>1</sup><http://www.alias-i.com/lingpipe/>

Contrary to expectations, results from using these features together were significantly less effective than the standard features. The best performance was observed when using linear kernels with a value of 0.75 for AUC.

### III.D Feed Based Features

Blogs have been considered as killer applications for RSS (RDF Site Summary) and Atom syndication standards, that specify an XML based vocabulary to represent information like title, date, summary etc. for time sensitive web content (e.g. a blog post). This metadata enables aggregation of content, from multiple blogs, thus freeing the consumer from the rather time consuming process of having to visit multiple websites.

From the perspective of splog detection RSS feeds presents multiple opportunities. The structured nature of feed gives a sense of life-cycle of the blog, in terms of both lifetime of the blog and the number of posts made.

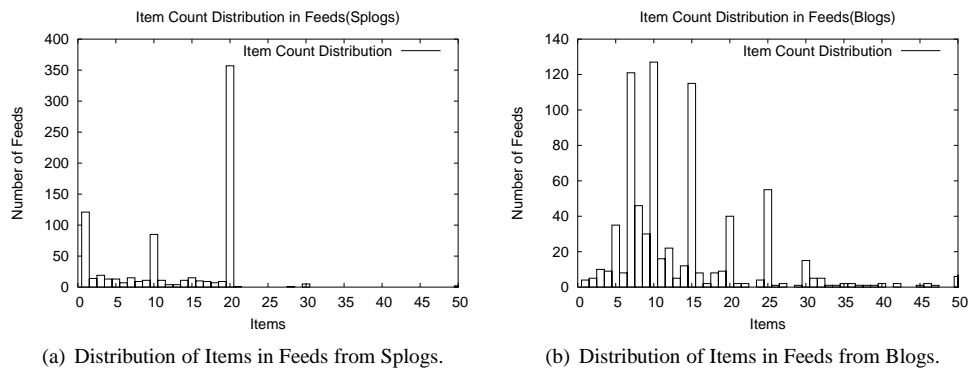


Figure III.8: Distribution of number of items in feeds.

Using feeds from the SPLOG-2006 we can see how a classifier using HTML tags can capture the stereotypes of splog creation software much earlier than a classifier using post content. Figure III.9 shows this result. The x axis represents the number of posts used in training the classifier, and the y axis represents precision and recall numbers. Interestingly, from the chart HTML tags are a much more effective feature type early during the lifecycle of a blog. Through this part of our work, we also introduce the notion of feed based classification, a problem whose importance will grow in the very near future.

There has been some related efforts on splog detection that capture correlations across posts [54], that can be leveraged in feed based classification, though the notion of blog lifecycle and its relationship to the effectiveness of features is not addressed.

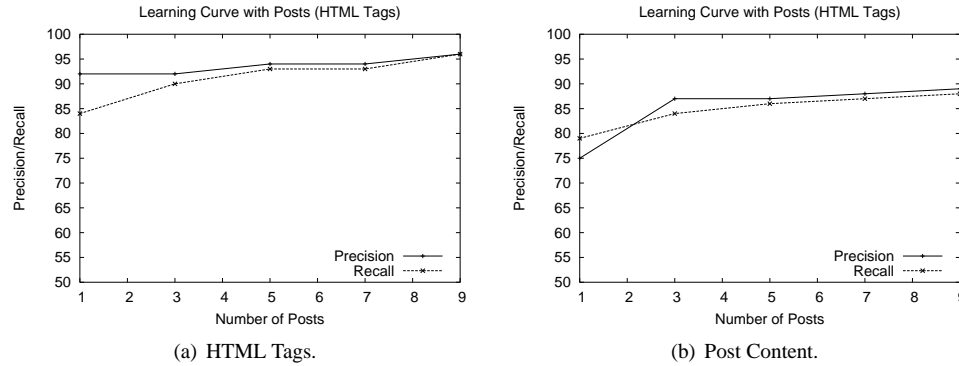


Figure III.9: Learning Curve with Feeds.

	P	R	F1
META	1	0.75	0.85
RSS/Atom	0.96	0.90	<b>0.93</b>
blog	0.88	0.79	0.83
comment	0.83	0.87	0.85
trackback	0.99	0.18	0.30
2005	0.56	0.97	0.71

Table III.9: Blog Identification Baselines.

### III.E Blog Identification

A second competency requirement from blog harvesters is the ability to separate out blogs from the rest of the Web. This requirement is seen both for focused blog crawling, as well as for filtering out non-blogs seen at ping servers. We discuss our feature identification efforts in this section.

We first make a case for using SVMs for blog identification by comparing with simple heuristics. These features include the HTML meta tag with name “generator”, the HTML link tag for RSS or Atom feed or the occurrence of one of the substrings “blog, 2005, comment, weblog” on a web page. The accuracy by using these heuristics individually are summarized in table III.9. To arrive at these results we used the **BHOME** sample set.

These numbers suggest that the existence of HTML link tag about RSS/Atom feed on a page is a good heuristic for blog identification. However two points should be noted here. First, the wide adoption of RSS and Atom feeds was initiated on the blogosphere, and is now seeing adoption elsewhere. This implies that the precision rates observed here will see a significant drop as feed links are published elsewhere on the Web. Second, we believe that our data collection process is slightly biased to collecting blogs which have RSS or Atom feeds. In fact, many blogs (more than what our recall suggests here) do not publish these syndication

Feature	#BHOME	#BSUB	#SPLOG
words	19000	19000	19000
urls	10000	10000	7000
anchors	10000	10000	8000
meta	2000	2000	3500
words+urls	29000	29000	26000
meta+link	2500	2500	4000
urls+anchors	20000	20000	15000
urls+anchors+meta	22000	22000	18500
4grams	25000	25000	25000

Table III.10: Feature types and their feature vector sizes used in experiments.

feeds. A recent statistic<sup>2</sup> by an influential blog search engine suggests that RSS and Atom enablement, atleast among the influential bloggers, is not all that high. Even with these factors we believe that results from blog identification using SVMs (without using HTML link element) should atleast match the one seen from RSS/Atom heuristic if not better them.

For all of our experiments we used the SVM toolkit [40] using linear kernels with the margin parameter “c” set to the default value. The feature types we experimented with and the number of top features (selected using mutual information) used for different sets are listed in table III.10. Each of these features was used in a bag-of-{tokens} where “token” stands for different feature types. We used binary feature values. Most of the feature names listed in the table is self-explanatory. “meta” was based on the HTML meta tag (name=“generator”), page title and page URL. The “link” feature type used contents of HTML link element on a page (with rel=’alternate’). The last feature type “4grams” used 4gram sequences on a combination of urls, anchors and meta information on a single page. Note also that we did not use “link” in all but one of the features to show that classifiers are accurate even without this information.

In the first set of experiments we used feature count (number of times the feature appears in all documents) as the feature selection technique for **BHOME**. The results were either similar or slightly less accurate than those we obtained using Mutual Information. Consequently, we report on only experiments using Mutual Information based feature selection.

Results for blog identification are shown in table III.11 with an f-1 measure of close to 98%.

Based on the success of the choice of features for identifying blog home pages, we next ran the same experiments on **BSUB** using binary (results in table III.12) features.

It was clear from these results that the same features which fared well for the blog home page identification

<sup>2</sup><http://www.sifry.com/alerts/archives/000310.html>

Feature	P	R	F1
words	.976	.941	.958
urls	.982	.962	.972
anchors	.975	.926	.950
meta	.981	.774	.865
<b>words+urls</b>	.985	.966	<b>.975</b>
meta+link	.973	.939	.956
urls+anchors	.985	.961	.973
<b>urls+anchors+meta</b>	.986	.964	<b>.975</b>
4grams	.982	.964	.973

Table III.11: Results for the BHOME dataset using Binary Features.

Feature	P	R	F1
words	.976	.930	.952
urls	.966	.904	.934
anchors	.962	.897	.923
meta	.981	.919	.945
<b>words+urls</b>	.979	.932	<b>.955</b>
meta+link	.919	.942	.930
urls+anchors	.977	.919	.947
<b>urls+anchors+meta</b>	.989	.940	<b>.964</b>
4grams	.976	.930	.952

Table III.12: Results for the BSUB dataset using Binary Features.

problem performed well in identifying all blog pages.

### III.F Relational Features

A global model is one that uses some non-local features, i.e., features requiring data beyond the content of Web page under test. Most blog related global features capture relations among web resources. In particular, we have investigated the use of link analysis to see if splogs can be identified once they place themselves on the web hyper-link graph. We want to capture the intuition that authentic blogs are very unlikely to link to splogs and that splogs frequently do link to other splogs. We tackle this problem by using splogs that could be identified using local attribute models. These splogs now become part of link distributions over which link-based models are constructed.

#### III.F.1 Labeling Nodes using Local Models

In the spirit of simulating the index of a typical blog search engine we employed the following technique. We started with the seed set (SPLOG-2005), and all of its in-link and out-link pages. We then used two fairly robust classifiers (with accuracy 95%, and part of our other projects) on blog and news-media detection to identify members of the set  $B$ ,  $N$  and  $W$  for the in-link and out-link pages.

Next, using this  $B$  set created from in-link and out-link pages, we experimented using different cut-off thresholds on a logistic regression based splog detection model built using local features. Using these cut-offs we labeled members of the set  $B$ . For any node  $x$  identified as a blog (and not part of the seed set), these thresholds  $th_1$  and  $th_2$  are used as:

$$x \in B_S, \text{ if } P(x \in B_S | O(x)) > \tau_1$$

$$x \in B_A, \text{ if } P(x \in B_A | O(x)) > \tau_2$$

$$x \in W, \text{ otherwise}$$

The interpretation of these thresholds is listed in Table III.13. The first and last values completely ignore the use of a local feature model to feed into the link-model.

#### III.F.2 Link Features

The link features of our choice are similar to those used in other link-based classification tasks [57]. Referring to our graph model, for all nodes in  $X = \{x_1, x_2, \dots, x_N\}$ , if  $e_{i \rightarrow j} \in E$  is a hyper-link from node  $x_i$  to node



Threshold	Comment
$\tau_1 = 0, \tau_2 = 1$	All Blogs are Splogs
$\tau_1 = 0.25, \tau_2 = 0.25$	Aggressive Splog Threshold
$\tau_1 = 0.5, \tau_2 = 0.5$	Intermediate Splog Threshold
$\tau_1 = 0.75, \tau_2 = 0.75$	Conservative Splog Threshold
$\tau_1 = 1, \tau_2 = 0$	All Blogs are Authentic

Table III.13: Interpretation of Probability Thresholds.

$x_j$ , then:

$L_I(x_i)$  - the set of all incoming links of node  $x_i$ ,  $\{x_j | e_{j \rightarrow i} \in E\}$

$L_O(x_i)$  - the set of all outgoing links of node  $x_i$ ,  $\{x_j | e_{i \rightarrow j} \in E\}$

$L_C(x_i)$  - the set of objects co-cited with node  $x_i$ ,  $\{x_j | x_j \neq x_i, \text{ and there exists another object } x_k, \text{ where } x_k \neq x_j, x_k \neq x_i \text{ and } x_k \text{ links to both } x_i \text{ and } x_j\}$

The nodes in each of these link distribution sets were assigned to their respective web graph sub-sets. Our features were finally based on using these assignments and computing set membership cardinality - as  $(|B_U|, |B_S|, |B_A|, |N|, |W|)$  for each of the link-distributions. This created a total of fifteen features for use by SVMs. We experimented with both binary and count based feature representation. Results are shown in Figure III.10 and Figure III.11, and probability threshold  $th_1$  is represented on the x axis. These charts show how local models that can be used to pre-classify out-links and in-links of the seed set is effective, and renders an otherwise inconsequential link-features only model useful. Augmenting these link features of the seed set with their bag-of-words did not improve accuracy beyond bag-of-words taken alone. This suggests that given the current nature of splogs, local textual content is arguably the most important discriminating feature.

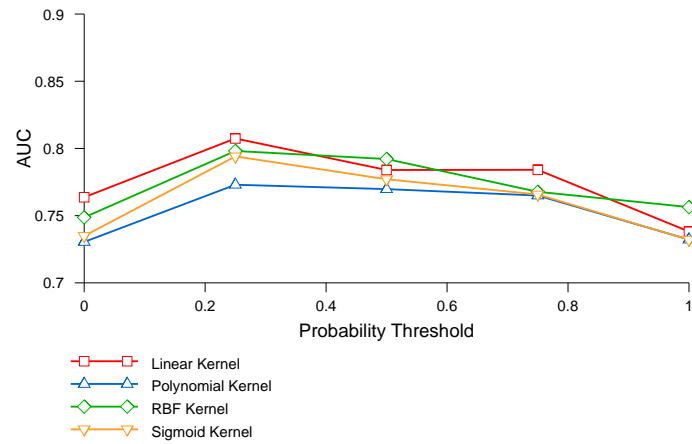


Figure III.10: Link features with binary encoding.

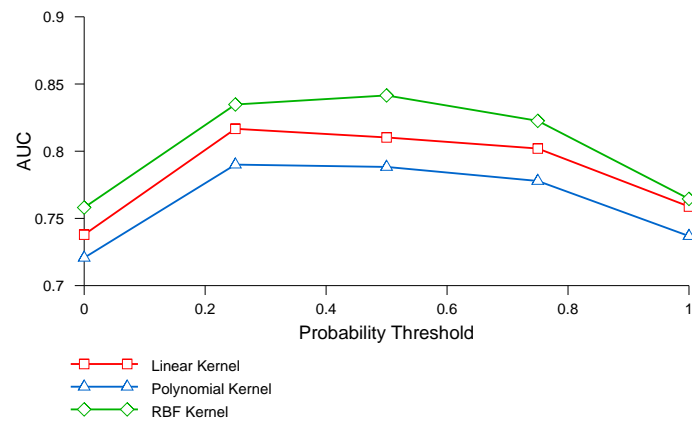


Figure III.11: Link features with frequency encoding.

## Chapter IV

# ENABLING ADAPTIVITY

The problem addressed in this chapter can be summarized as follows:

**Given a catalog of detection techniques (their classifiers), and a stream of unlabeled instances susceptible to drift (potentially adversarial), how can these classifiers adapt collectively?**

The motivation behind addressing the problem is multi-fold.

First, detecting spam blogs is a classic case of adversarial classification i.e., where classification is essentially a game between the classifier and an adversary, where based on partial or complete knowledge of the classifier, the adversary constantly adapts to obfuscate spamming techniques. Such obfuscation can be viewed as a change in distribution in the underlying feature space, and is beginning to become quite common in spam blogs. Spammers are increasingly plagiarizing content from authentic blogs, and sandwiching terms in profitable context between them.

Second, detecting spam blogs on a stream of update pings has to, more generally, deal with concept drift, where the underlying distribution in the feature space can change naturally, which need not be necessarily adversarial. Note that this kind of change in distribution in the feature space is seen in both authentic blogs and splogs. Topics and content hosted on authentic blogs are quite susceptible to drift seasonally. Splogs could feature drift through seasonal popularity of certain keywords, or through the identification of new profitable contexts by spammers.

Third, there has not yet been sufficient exploration of semi-supervised learning in adversarial classification domains. Semi-supervised learning utilizes unlabeled samples to improve classification accuracy, and

relieves the large time and resource requirements associated with labeling examples. Further, the problem has not yet been looked at exclusively in the web spam detection space, a detection problem that typically uses a large catalog of classifiers.

## IV.A Related Work

The problem we address shares attributes with related efforts by the research community, namely, concept drift [81] [78], adversarial classification [19], and semi-supervised learning [88][7]. Drifting concepts is a result of multiple factors[42] [51]. Using the notation introduced earlier, and for simplicity using only object features, the problem is viewed as a change in concept description due to changes in the:

1. Prior probability,  $P(x \in B_S)$
2. Posterior probability,  $P(x \in B_S/O(x))$
3. Class-conditional probability distribution,  $P(O(x)/x \in B_S)$

The change in class-conditionals is clearly a case of adversarial classification, where as the change in posterior probability is a classic case of concept drift due to seasonality. Either of these changes in distribution requires that the classifiers be updated to reflect the drift.

## IV.B Our Contribution

Though existing work on concept drift, and semi-supervised learning (specifically, co-training) share attributes with the problem we address, we relax assumptions made in existing work, and better tune to the solution to the adversarial classification domain. We list them as:

1. Concept drift is generally applied in the context of a stream of labeled instances, unlike our work which exploits a stream of unlabeled instances.
2. Co-training is based on conditional independence of two feature spaces, using two classifiers. We explore it from a multi-feature space context, using a catalog of classifiers.
3. Co-training is used when the base classifiers are weak.
4. Concept drift techniques have largely been explored and evaluated when the underlying distribution changes are sudden and drastic.

Train,Test	P	R	F1
SPLOG-2005,SPLOG-2005	0.71	0.70	0.71
SPLOG-2006,SPLOG-2006	0.89	0.85	0.87
SPLOG-2005,SPLOG-2006	0.61	0.72	0.66

Table IV.1: URL n-gram Adaptive Potential..

Train,Test	P	R	F1
SPLOG-2005,SPLOG-2005	0.82	0.85	0.83
SPLOG-2006,SPLOG-2006	0.92	0.94	0.93
SPLOG-2005,SPLOG-2006	0.83	0.81	0.82

Table IV.2: Anchor Adaptive Potential.

5. Concept drift evaluation has so far largely used synthetic data.

We address the problem of adapting each of the classifiers that support the META-PING system, as they work on a stream of unlabeled instances. Our work is the first that addresses the problem in this splog detection context, and arguably the first in the adversarial classification domain.

## IV.C Adapting Potential

The potential of the classifiers to adapt is largely dependent on the change in the underlying distribution in the feature space. To examine this potential, we first discuss a motivating result.

The first set of tables show precision, recall and f-measure for 7 classifiers evaluated in the previous chapter, namely URL n-gram (table IV.1), words (table IV.6), word-n-grams (table IV.7), charactergrams (table IV.3), tags (table IV.5), out-links (table IV.4), anchor text (table IV.2). Classifiers were trained using SVMs, with the top ten thousand features selected (based on frequency) from SPLOG-2005. SVMs were used with default parameters and a linear kernel. A row item, “SPLOG-2005, SPLOG-2006” stands for a classifier trained on SPLOG-2005, and tested on the SPLOG-2006 dataset. The row “SPLOG-2006, SPLOG-2006” shows the upper limit that can be potentially reached, when labeled samples for SPLOG-2006 is available. The difference between “SPLOG-2005, SPLOG-2006” and “SPLOG-2006, SPLOG-2006” gives the potential for improvement of the classifiers using semi-supervised learning. Clearly, all base learners show a high potential for improvement.

Train,Test	P	R	F1
SPLOG-2005,SPLOG-2005	0.88	0.86	0.87
SPLOG-2006,SPLOG-2006	0.93	0.94	0.94
SPLOG-2005,SPLOG-2006	0.84	0.71	0.77

Table IV.3: Chagram Adaptive Potential.

Train,Test	P	R	F1
SPLOG-2005,SPLOG-2005	0.81	0.82	0.82
SPLOG-2006,SPLOG-2006	0.94	0.96	0.96
SPLOG-2005,SPLOG-2006	0.85	0.88	0.87

Table IV.4: Outlink Adaptive Potential.

Train,Test	P	R	F1
SPLOG-2005,SPLOG-2005	0.77	0.76	0.77
SPLOG-2006,SPLOG-2006	0.93	0.90	0.92
SPLOG-2005,SPLOG-2006	0.77	0.80	0.78

Table IV.5: Tag Adaptive Potential.

Train,Test	P	R	F1
SPLOG-2005,SPLOG-2005	0.89	0.87	0.88
SPLOG-2006,SPLOG-2006	0.96	0.96	0.96
SPLOG-2005,SPLOG-2006	0.88	0.83	0.85

Table IV.6: Words Adaptive Potential.

Train,Test	P	R	F1
SPLOG-2005,SPLOG-2005	0.85	0.85	0.85
SPLOG-2006,SPLOG-2006	0.89	0.93	0.91
SPLOG-2005,SPLOG-2006	0.87	0.86	0.86

Table IV.7: Wordgram Adaptive Potential.

## IV.D Our Approach

Our approach is motivated by the following intuition. Unlike topic classification, most practical solutions to (web) spam detection typically uses a catalog of classifiers, each of which differentiates itself through either the feature space or underlying classification technique. The use of multiple classifiers is quite common, and tends to make the overall filter more robust to obfuscation techniques in an adversarial situation. These classifiers are retrained when new, labeled samples, become available, an activity generally carried out in batches (day, month and so on).

The above scenario presents interesting opportunities. We need to explore adaptive techniques between the batch updates to the classifier, using unlabeled instances. We propose an approach that uses an ensemble of the base classifiers. To discuss our approach in more detail, we introduce the following notations:

Time batches, represented as  $\tau_1, \tau_2 \dots \tau_n$ , with sets of instances  $X_1, X_2 \dots X_n$

Individual instance,  $x_i^j$ , that represents instance  $j$  at  $\tau_i$ , with  $x_i^j \in X_i$

A catalogue of classifiers,  $\zeta_1, \zeta_2 \dots \zeta_m$ , using disjoint feature spaces,  $O_1, O_2 \dots O_m$

$P(x_i^j/O_k)$  to represent the output of a probabilistic classifier  $\zeta_k$ , for instance  $x_i^j$

$y_i^j$ , the true label for instances

$\xi_i^j$ , the predicted classification from the ensemble of  $\zeta_1, \zeta_2 \dots \zeta_m$

The approach is depicted as follows:

**Input:** A catalog of classifiers,  $\zeta_1, \zeta_2 \dots \zeta_m$ , trained using a labeled batch  $a$ , and exposed to an unlabeled batch  $b$ , re-train module that uses  $y_i^j$  or in its absence  $\xi_i^j$

**Output:** Updated classifiers,  $\zeta_1^u, \zeta_2^u \dots \zeta_m^u$

**foreach** unlabeled instance  $x_i^b$  **do**

$$p_i^b = \frac{\sum_k P(x_i^b/O_k)}{m};$$

$\xi_i^b = 1;$

**if**  $p_i^b \leq 0.5$  **then**

$\xi_i^b = 1;$

**end**

**end**

**foreach** classifier  $\zeta_k$  **do**

$\zeta_k^u = \text{re-train}(X_a \cup X_b);$

**end**

**Algorithm 1:** Ensemble driven classifier adaptivity.

Intuitively, the technique works as follows. A probabilistic committee based ensemble is created using all the classifiers in the catalogue. This ensemble labels new instances, and feeds such labeled instances back into the learning algorithm supporting the base classifiers, enabling retraining and hence adaptive base classifiers.

We evaluate this technique using two of our available datasets, SPLOG-2005 and SPLOG-2006, that represents two batches. The base classifiers are trained using SPLOG-2005, and a probabilistic committee based ensemble is created using the seven base classifiers. The set of plots that follow show how feedback from this ensemble is effective in adapting each of these base classifiers by providing labeled instances. In the plots “fbkensemble” represents retraining using feedback from the ensemble of classifiers, “fbkgold” represents gold feedback i.e., assumes that the ensemble is 100% accurate, and sets the top-line, “fbkself” represents feedback from self (self-training) and sets the baseline, and “fbkonly” represents the case where only instances derived through feedback from the ensemble is used i.e., the 2005 dataset is dropped during re-training. F-1 measure is plotted on the y-axis. The x-axis plots the number of labeled instances that are part of the feedback from the ensemble. The rest of the instances in SPLOG-2006 is used for testing. All values are based out of ten runs.

Clearly, “fbkensemble” plots across all classifiers show that using an ensemble based feedback to re-train base classifiers can be quite effective. Clearly, this effectiveness is largely tied to the properties of the classifiers that are part of the ensemble. We discuss these properties in the next section.

## IV.E Properties of the Ensemble

In this section we discuss the basic properties of the base classifiers that support the ensemble used in the previous section.

First, the overall precision and recall values of the ensemble for the positive class (splogs) is 92% and 93% respectively. Note that this is significantly higher than the effectiveness of any of the base classifiers taken alone on SPLOG-2006. This effectiveness is attributed to the diversity of the base classifiers, which can be evaluated using different metrics.

We measure diversity using the Q-statistics [86], which is defined between two classifiers as:



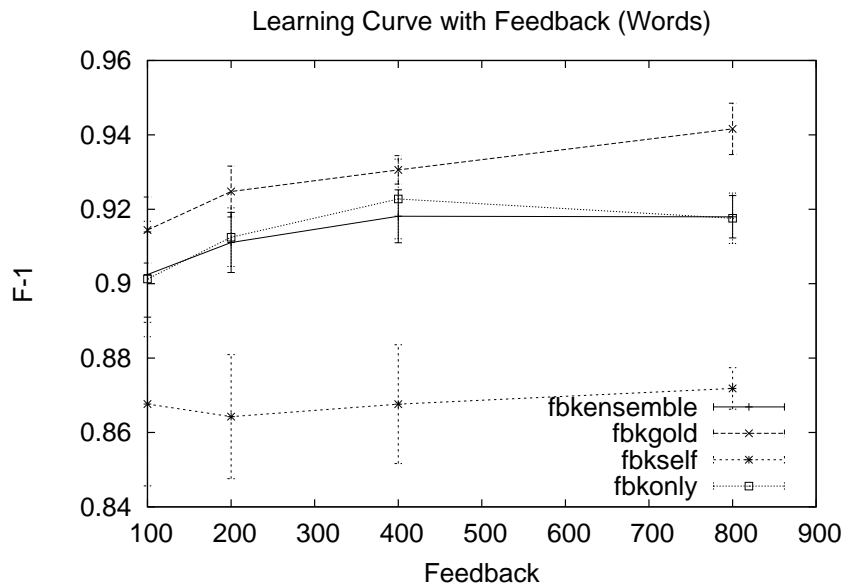


Figure IV.1: Ensemble Feedback to Words.

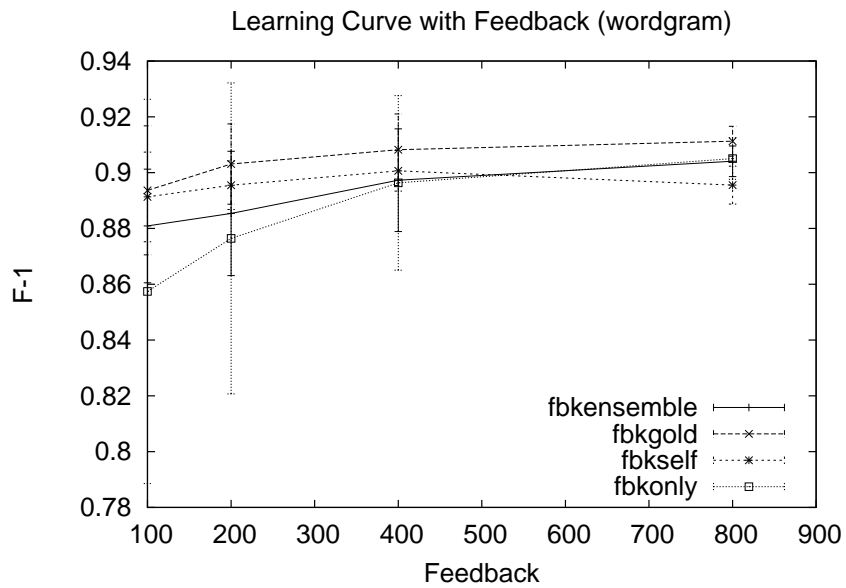


Figure IV.2: Ensemble Feedback to Wordgram.

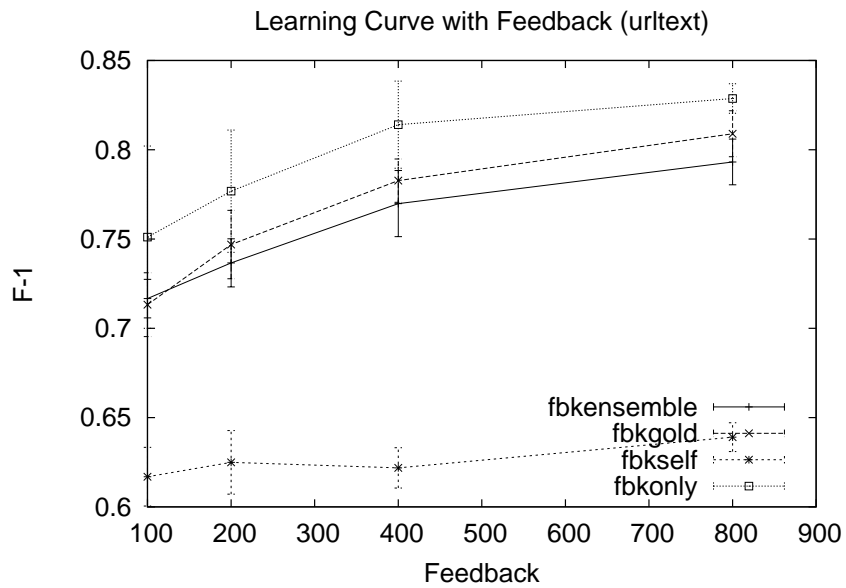


Figure IV.3: Ensemble Feedback to URLText.

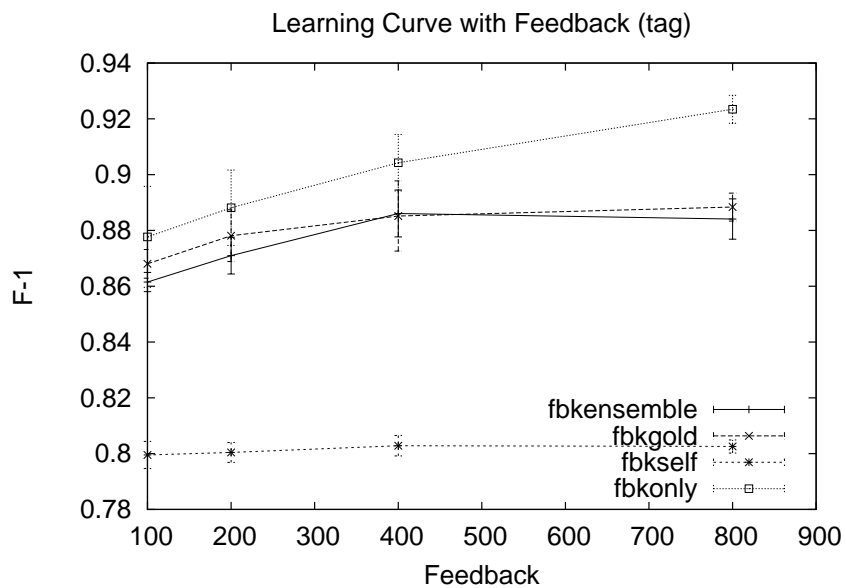


Figure IV.4: Ensemble Feedback to HTML Tags.

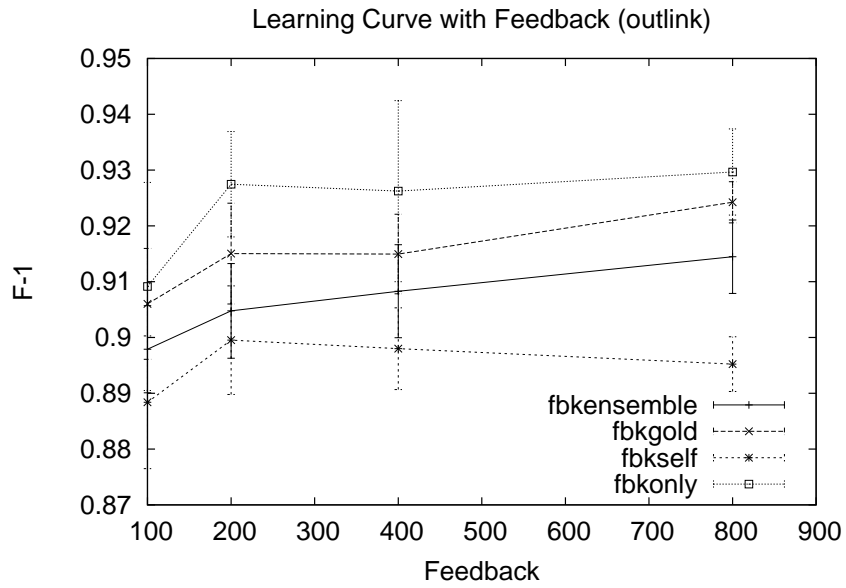


Figure IV.5: Ensemble Feedback to Outlink.

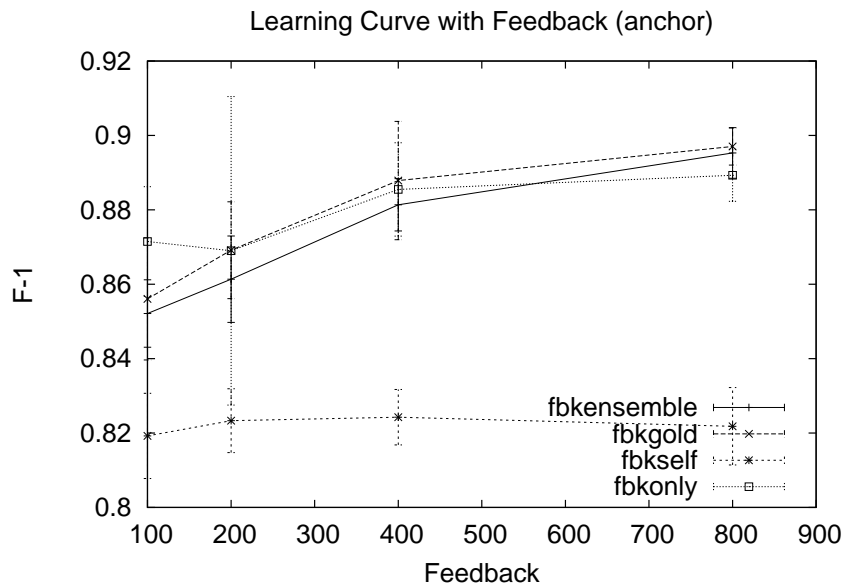


Figure IV.6: Ensemble Feedback to Anchor.

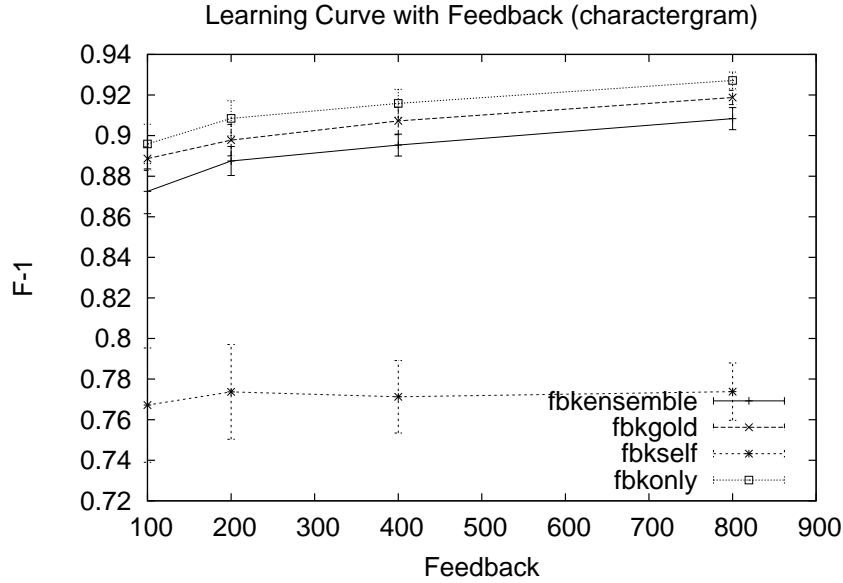


Figure IV.7: Ensemble Feedback to Charactergram.

$$Q = \frac{N^{11}N^{00} - N^{01}N^{10}}{N^{11}N^{00} + N^{01}N^{10}}$$

where,  $N^{11}$  and  $N^{00}$  are the number of times both classifiers are correct or incorrect respectively, and  $N^{10}$  and  $N^{01}$  are the number of times either one of them is correct and the other incorrect. The value of  $Q$  is bounded by  $[1, -1]$ , with one representing complete agreement, and minus one signifying complete diversity. Classifier with low pair-wise values of  $Q$  are considered useful members of an ensemble.

We evaluated the pair-wise  $Q$ -statistic for all seven classifiers making up the ensemble. Results are shown in table IV.8, with classifier pairs listed in increasing value of  $Q$  scores, and hence decreasing diversity.

$Q$ -statistics of the base classifiers leads to interesting observations. Clearly, the classifier using HTML Tags on a page is the best member of the ensemble with high diversity values across all other classifiers. Note that the feature space of this classifier has no overlap with the other classifiers, which are all based out of content on the blog. In addition, other novel features discovered by us, namely outlinks and anchors consistently feature high diversity values indicating their importance in an ensemble, and emphasizing the importance of novel feature discovery in adversarial contexts.

Though we have used an ensemble which uses the full catalog of classifiers, the subset selection problem [52][1], from such a catalog presents interesting research opportunities.

Classifier Pairs	qstat
charactergrams, tags	-0.23
words, tags	-0.19
anchors, tags	-0.12
tags, wordgrams	-0.08
anchors, url	0.03
url, words	0.04
charactergrams, url	0.08
outlinks, url	0.10
wordgrams, url	0.11
outlinks, tags	0.15
tags, url	0.24
charactergrams, outlinks	0.35
anchors, outlinks	0.45
outlinks, words	0.53
anchors, wordgrams	0.56
anchors, charactergrams	0.58
outlinks, wordgrams	0.62
charactergrams, wordgrams	0.67
wordgrams, words	0.77
charactergrams, words	0.86

Table IV.8: Q-statistics of Classifiers.

## IV.F Use in META-PING system

We believe that an approach outlined above can be quite effective in any domain with drifting concepts, either adversarial or seasonal. Though we have not deployed it in the META-PING system, we clearly see the utility of using this technique. The benefit of using classifier co-evolution is two fold.

First, it relieves labeling efforts required to maintain effectiveness of the META-PING system. Referring back to the time batches,  $\tau_1, \tau_2 \dots \tau_n$ , consider that labeled samples are available at time  $\tau_i$ , obtained by sampling splogs seen during time leading up to  $\tau_i$  and including  $\tau_i$ . Consider that no new update is available until  $\tau_j$ . All batches between  $\tau_i$  and  $\tau_j$ , can clearly exploit the ensemble based approach.

Second, classifiers are used in a pipeline-based approach in the META-PING system, placed in increasing cost of detection. An improvement in classification accuracy early in the pipeline could lead to significant reduction in computational costs.

## **Chapter V**

# **CASE STUDIES**

Being the first to have addressed the problem of spam blogs, and having developed tools that were capable of detecting splogs in a real-world setting, we were continuously involved in updating ourselves and the research and technical audience on the characteristics of splogs. During this process we participated in the TREC Blog Track open task on Splog detection, published studies on the severity and seriousness of splogs, and carried out hands-on experiments that enabled us to better understand the problem.

In this chapter we discuss some of our efforts in this direction.

## V.A TREC Blog Track 2006

TREC Blog Track 2006 asked participants to implement and evaluate a system for “opinion retrieval” from blog posts. Specifically, the task was defined as follows: build a system that will take a query string describing a topic, e.g., “March of the Penguins”, and return a ranked list of blog posts that express an opinion, positive or negative, about the topic. For evaluation, NIST provided a dataset of over three million blogs drawn from about 80 thousand blogs. Participants built and trained their systems to work on this dataset. Contestants do an automatic evaluation by downloading and running, without further modification to their systems, a set of fifty test queries. The results are evaluated by NIST in an annual competition.

We studied the impact of splogs [49] during our participation in TREC 2006 [39], and report them here. Our analysis is based on a splog detection technique that works on a blog home-page using word based features. We found that splogs significantly impact blog analytics by affecting the opinion retrieval task, and more generally query relevance.

### V.A.1 Impact of Splogs

In order to make the challenge realistic NIST explicitly included 17,969 feeds from splogs, contributing to 15.8% of the documents [58]. There were 83,307 distinct homepage URLs present in the collection, of which 81,014 could be processed. The collection contained a total of 3,214,727 permalinks from all these blogs. Our automated splog filter identified 13,542 splogs. This accounts for about 16% of the identified homepages. The total number of permalinks from these splogs is 543,086 or around 16% of the collection. While the actual list of splogs is not available for comparison, the current estimate appears close.

To keep the analysis generic, we evaluate the influence of splogs in the context of search engine retrieval. Given a search query, we would like to estimate the impact splogs have on search result precision. Figure V.1 shows the distribution of splogs across the 50 TREC queries. The number of splogs present varies across the queries since splogs are query (topic) dependent. For example, the topmost spammed query terms were ‘cholesterol’ and ‘hybrid cars’. Such queries attract a high paying advertisement market, which splogs exploit.

The description of the TREC collection [58] provides an analysis of posts from splogs that were added to the collection. Top informative terms include ‘insurance’, ‘weight’, ‘credit’ and such. Figure V.2 shows the distribution of splogs identified by our system across such spam terms. In stark contrast from Figure V.1 there is a very higher percentage of splogs in the top 100 results.

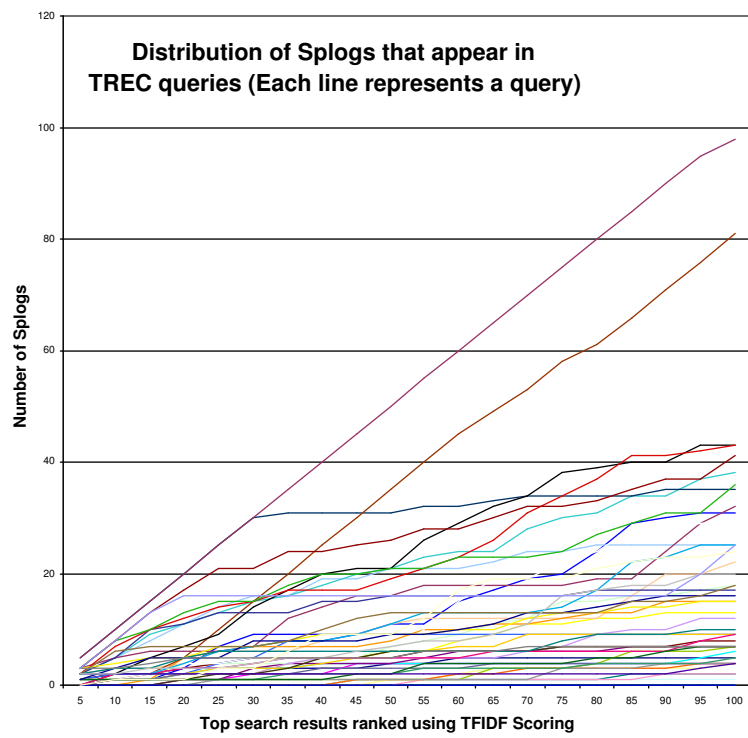


Figure V.1: The number of splogs in the top 100 results for 50 TREC queries.



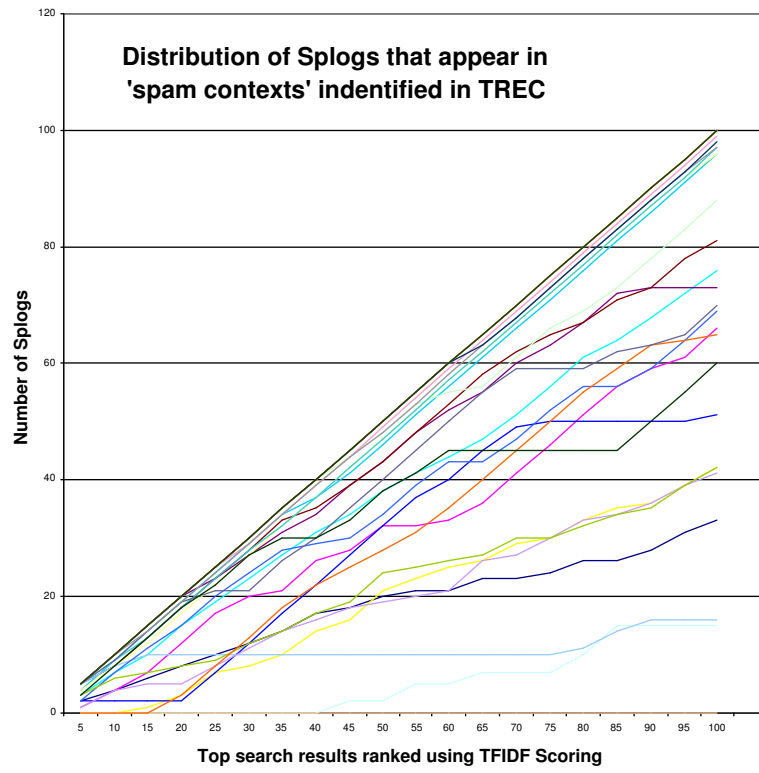


Figure V.2: The number of splogs in the top 100 results of the TREC collection for 28 highly spammed query terms.

## V.A.2 Splog Task Proposal

Based on our analysis we proposed a splog task proposal. Such open task proposals are used as input if new tasks are formally announced by NIST.

We approach splog detection at the blog home-page level, and then propagate to all posts (permalinks) hosted by the blog. Although this seems to work well in practice, is in use by a blog search engine and a partner academic institution, we propose a more structured approach for TREC.

Inspired by e-mail spam detection, we argue that permalinks (individual posts) be treated as atomic entities for assessment (and hence evaluation) in the splog detection task, irrespective of whether splogs are detected at the post or blog home-page level. Independent of IP blacklists, blacklisted e-mail addresses (analogous to blog home-pages) and proxies, e-mail spam detection techniques are evaluated on a per-mail basis. For a splog detector permalinks could be treated analogously to emails received at an address, providing intuition and structure to the task. This also aligns with how blog search engines tap into blog updates, making on the fly decisions about indexing new posts.

We propose a categorization scheme (possibly overlapping) for spam permalinks based on a web spam taxonomy scheme proposed by [34], and our experience dealing with spam blogs (additional details in the Appendix).

- **non-blog** pages attempt to associate themselves with the blogosphere to exploit increased search engine exposure. Non-blogs usually infiltrate the blogosphere through unjustified pings at update ping servers.
- **keyword-stuffing** targets tfidf [76] based relevance measures used by search engines. Spam posts repeat target keywords (query terms) multiple times on their pages.
- **post-stitching** is used to automatically generate content, by combining excerpts from plagiarized posts or news stories, in high paying advertising contexts.
- **post-plagiarism** is full content theft from blogs or news pages. The recent debate surrounding Bitacle<sup>1</sup> is one such example.
- **post-weaving** is used to conceal excessive links to affiliate sites by copying entire posts from other blogs and weaving in hyperlinks promoting affiliates.

---

<sup>1</sup><http://bitacle.org>

<i>Score</i>	Interpretation
-1	Not judged
0	Authentic blog post
1	non-blog
2	keyword-stuffing
3	post-stitching
4	post-plagiarism
5	post-weaving
6	link-spam
7	other-techniques

Table V.1: Proposed assessment scores for spam blog classification.

- **link-spam** is an attempt to artificially inflate page rank or get new pages indexed, using link dumps that contribute to a link-farm. Note that post-weaving can be considered a form of link-spam.
- **other-techniques** group techniques that appear in none of the classes above - common ones being page redirection and cloaking.

The motivation behind a categorization is that different detection models will have to be developed based on the category of spam to be detected. For instance, our own word based model works well for the keyword-stuffing and post-stitching category. We believe that such an explicit categorization will encourage the consideration of all aspects of spam blogs by task participants.

Our proposed assessment values and their interpretation is shown in table V.1. Assessment is done to permalinks independent of query term or context. “-1” score represents a non-judgment, and is similar in semantics to its use in Blog Track 2006 opinion task. “0” represents an authentic blog post, and the rest of the scores are used to identify a spam post and its category.

Though assessments are attached to permalinks, the assessor labels blog home-pages which can then be propagated down to all permalinks from the blog. Based on our experience labeling spam blogs, each assessment takes around 1-2 minutes, as measured from the time the page was accessed to the time the assessment was entered.

Assuming the existence of a TREC Collection similar to the collection of 2006 spanning  $n$  days, we propose that the dataset be divided into two subsets. The first subset, from here on referred to as  $D_{base}$  will span the first  $(n - x)$  days of the collection and the second, referred to as  $D_{test}$  will span the last  $x$  days of the collection. The exact value of  $x$  will be decided collectively and could be one, two or more days.

$D_{base}$  will be released at task announcement for participants to train and build their splog detection

models.  $D_{test}$  will be released subsequently along with a input (test) file to the spam blog detector. Unlike TREC Blog 2006 where systems were judged by 50 independent topics (queries), the proposed task will be judged based on 50 independent sets of permalinks (sampled from  $D_{test}$ ). The cardinality of each such set will be arrived at through further discussions. We believe this is a good model for what blog and feed search engines have to do i.e., make judgment on newly created posts based on knowledge gathered while indexing earlier posts, observed attributes of blogs vs. splogs, and models they built around them.

Spam blog detectors developed by participants will rank the set of permalinks based on an estimated “splogginess”. The overall evaluation of systems will be done just on this ranking, but the category data will allow participants to see where their systems were strong and weak, to informally compare across participants, and will serve as feedback for overall improvement of the quality of the blogosphere.

### **Dataset Creation**

The approaches followed in the creation of the TREC 2006 Blog Collection is detailed in [58]. In addition to permalinks, blog home-pages and feeds were also part of the collection, cached regularly polling a static list of blogs (and splogs) over a period of 77 days. One key component missing in this collection (and important) for a spam blog classification task is the dynamic aspect of newly created blogs and their posts; splogs are transient and short-lived.

To avoid replication of data collection efforts, an approach to create collections for multiple tasks together can be used. To overcome the problem<sup>2</sup> noted in [58] a ping server with better coverage<sup>3</sup>, or multiple ping servers together can be first employed to tap into updates in the blogosphere. As a next step two collections could be created from the updates - (i) posts from all blog updates, and (ii) posts that intersect with the static list of authoritative bloggers used in TREC Blog Collection 2006. The first collection can then be employed for spam blog classification, and the second for tasks around blog analytics.

### **Input Format**

The input file consists of 50 independent sets of permalinks. The association of a permalinks with home-page, syndication feed, and post time-stamp is also specified with the input. Track participants can use any of them or their combination, but make explicit which fields were used.

<set>

---

<sup>2</sup><http://pubsub.com> served pings for only 37% of the blogs in TREC Blog Collection 2006

<sup>3</sup><http://blo.gs> can be employed over <http://pubsub.com>

```

<num>...</num>

<test>

<permalink>

<url>...</url>

<homepage>...</homepage>

<feed>...</feed>

<when>... </when>

</permalink>

<permalink>

...

</permalink>

...

</test>

</set>

<set>

...

</set>

```

### Output Format

The output format from a TREC run will be similar to the format used in the TREC Blog Track 2006 on Opinion Identification. Permalinks in each of the sets are to be ranked based on “splogginess” score.

**set Q0 docno rank prob runtag**

where *set* is the input permalink set, *Q0* is literal “Q0”, *docno* is the permalink identifier, *rank* is the final rank returned by the system, *prob* is the probability associated with spam judgment and *runtag* is the run’s identifier string. Participants will be judged on precision/recall across a combination of all categories of splogs.

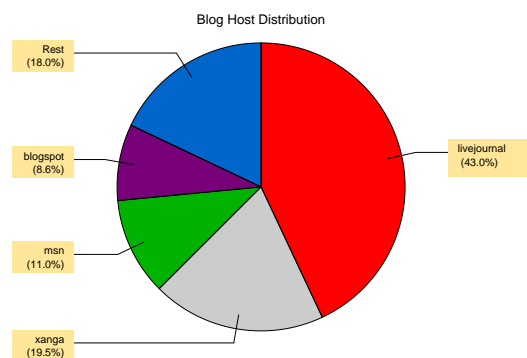


Figure V.3: Blog host distribution in the BlogPulse dataset.

## V.B Blogosphere vs. Splogosphere

BlogPulse<sup>4</sup>, a popular blog search and mining system, recently released a dataset spanning a period of 21 days in July of 2005. This dataset consists of around 1 million blogs with a total of 10 million posts from these blogs. The dataset also contains additional metadata of these posts that include out-links, tags and post-timestamps. To enable better understanding of our results, we base a large part of our analysis on this dataset. The relative frequency of various blog hosts in this dataset is shown in Figure V.3.

Our detection modules are based on analyzing the complete structure of blog home-pages and not just individual posts. Such an approach captures interesting features common to multiple posts on a blog home-page and also uses other information like blogrolls and non-post out-links before making a splog judgement. To adhere to this requirement, we extracted blog home page URLs from the BlogPulse dataset, and re-fetched the complete home-pages to analyze their content. It turns out that many of these home-pages (possibly splogs) are now non-existent either because they were detected and eliminated by blog hosts or pulled down by spammers as they were no longer useful. The number of failed blogs was as high as around 200K. Since we are not in a position to ascertain the true nature of these failed URLs with a high confidence we dropped them from consideration.

Of the remaining blog home pages we noticed that live-journal had an insignificant percentage of spam blogs<sup>5</sup>. Given that live-journal forms a large fraction of authentic blogs in the dataset we eliminated all blogs from this domain and worked with blogs from other domains and self-hosted blogs. The primary reason was to eliminate the characteristics of live-journal blogs biasing our results.

<sup>4</sup><http://blogpulse.com>

<sup>5</sup>This need not necessarily hold for blogs created as of March 2006

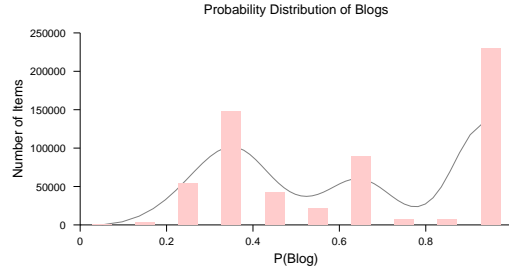


Figure V.4: Probability Distribution of Blogs in BlogPulse.

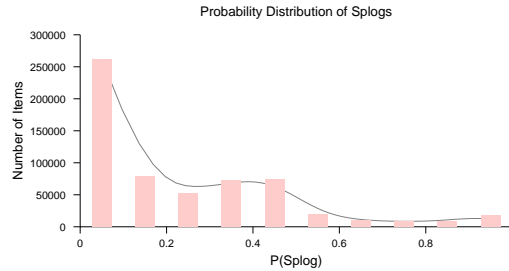


Figure V.5: Probability Distribution of Splogs in BlogPulse.

After filtering out the above mentioned blogs, and blogs that are not in English we ended up with around 500K blogs. The probability distribution provided by our blog identification module is shown in Figure V.4, and the distribution of splogs returned by the splog detection module is shown in Figure V.5.

Each bar on the x-axis represents a probability range and values on the y-axis represent the number of pages (blogs) that are within this range.

Typically, we use results from blog identification to feed into splog detection. However we ignored probability distribution of blogs and made an assumption that all blogs in the BlogPulse dataset are truly blogs. We then used the following thresholds from the splog detection module to create subsets of authentic blogs and splogs used in our characterization.

$$X \in \text{AuthenticBlog}, \text{ if } P(X = \text{Splog}/\text{Features}(X)) < 0.25$$

$$X \in \text{Splog}, \text{ if } P(X = \text{Splog}/\text{Features}(X)) > 0.8$$

In these two created subsets, the cardinality of the splog subset was around 27K. We uniformly sampled for 27K authentic blogs to have two subsets of the same cardinality. In what follows, our comparative character-

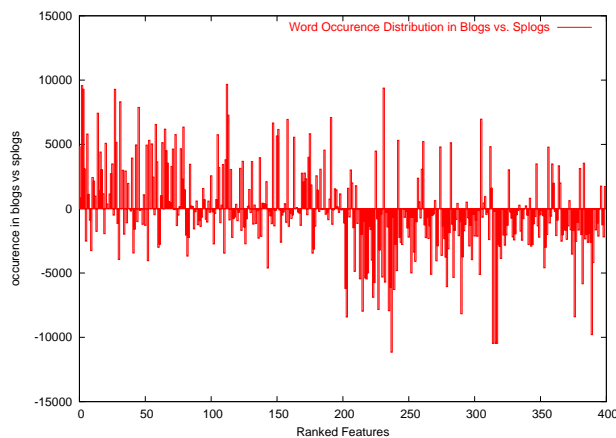


Figure V.6: Distribution of top discriminating word-features in blogs and splogs.

ization is based on 27K splogs and 27K blogs.

### V.B.1 Frequency of Words

We first analyzed the distribution of certain discriminating terms in both blogs and splogs. Since our splog detection module is built using only local features, it is these discriminating features that were employed by our system. We created a ranking of features based on weights assigned to the features by the SVM model. This list consists of 200 word features common to blogs and 200 word features common to splogs. The word features common to blogs included pronouns like “I”, “We”, “My” and words from anchor text to popular websites like flickr, Technorati etc, which were all less common in splogs. Splogs generally feature high paying adsense<sup>6</sup> keywords.

The occurrence based distribution of terms common in blogs and splogs for these top features is shown in Figure V.6. The first half on the x-axis depicts the top blog features and the second half depicts the top splog features. The y-axis represents the difference between the number of blogs in which the feature occurred to the number of splogs in which the same feature occurs. Clearly, the top blog features occur more frequently in blogs than splogs and vice-versa. Similar patterns can be observed in a comparison using 2-gram words and 3-gram words [20], and models based on such local knowledge give detection F1 estimates of close to 90%.

---

<sup>6</sup><http://google.com/adsense>



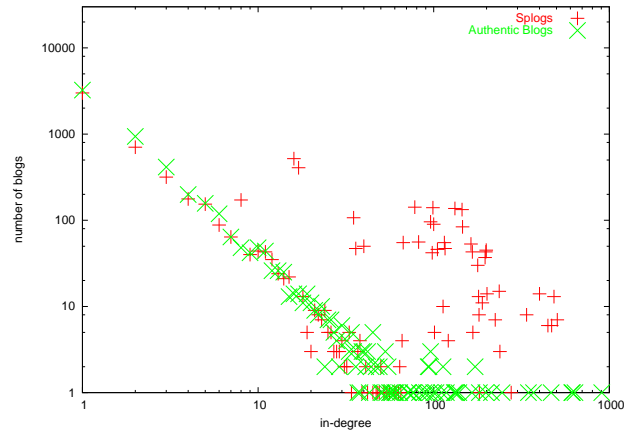


Figure V.7: In-degree distribution of authentic blogs subscribe to a power-law.

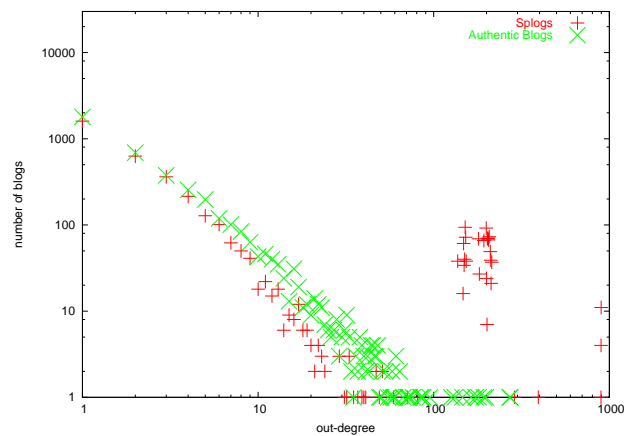


Figure V.8: Out-degree distribution of authentic blogs subscribe to a power-law.

## V.B.2 Link Structure

Splogs that escape existing filters engage in creating link-farms to increase the importance of pages in the farm, scores computed using PageRank[71]. The distribution of inlinks for splogs and authentic blogs is shown in figure V.7, where the link graph was obtained from the weblogs dataset. Blogs show a power-law that is typical to the Web in general[10], whereas splogs deviate from this norm. We also followed this up by checking for outlink distribution of splogs and blogs. Figure V.8 shows this distribution, with blogs complying with the power-law as opposed to splogs which does not adhere to it.

Since post time-stamps in the BlogPulse dataset are not normalized across blogs, we do not make an analysis of post time stamps here. Any such analysis will be similar to that put forward in our next section on spings. In addition to these characteristics, we also noticed certain patterns in other aspects of splogs. For

instance, from the tagging perspective most of the splogs are tagged as “un-categorized”. However all these discriminating features are incorporated in the word characterization discussed earlier, which incorporates all of the text (including anchor-text) on a blog.

Based on these results, and a related analysis [50], we make the following observations:

- Given the current nature of splogs, their detection is quite effective through the use of only local features. Word model of blogs based on local features create an interesting “authentic blog genre” that separate them from splogs.
- If splogs do happen to escape filters and then indulge in the creation of link-farms, many of them can be detected using spam detection algorithms on the web graph [87]. However, this approach taken alone has two disadvantages. First, it allows splogs to thrive in blog hosts and search engines for a longer period of time, and second, it fails to detect splogs which are not part of abnormal link sub-structures.

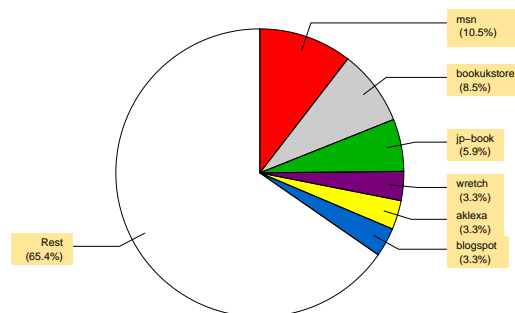


Figure V.9: Host distribution of pings received by an Update Ping Server.

## V.C Splogs and Ping Servers

Blogs notify update ping servers when new posts are made; these servers then route such pings to systems that index and analyze blog content. Independent from the BlogPulse dataset we also analyzed pings received by a popular blog update ping server<sup>7</sup> that makes these pings public. We analyzed around 15 million pings over a period of 20 days from November 20, 2005 to December 11, 2005 to check how many of these are spings, i.e., from splogs. The relative frequency of hosts pinging the update server is shown in Figure V.9.

Ping Servers define standard interfaces that can be used by blogs to notify new (or updated) posts. Information about the blog home-page and blog title<sup>8</sup> typically accompany these pings. Additional information like syndication feed location can also be specified, but is less common. Other than restrictions on their frequency, no other restriction is usually placed on pings. Driven by this restriction-free nature, and the improved search engine exposure (both blog search and web search) ping servers provide, splogs overwhelm ping servers.

Ping Servers are faced with two kinds of spam - (i) pings from non-blogs, and (ii) pings from splogs, both of which are referred to as spings. We used a similar approach to the one we used for splog detection in the BlogPulse dataset. However to scale up to the number of pings that have to be processed, we used simpler techniques and made some exceptions. We used URL based heuristics for blog identification and did not pass pings from the info domain through our filters. However for all other pings, we fetched the home-pages of pings to make a splog judgment. We also identified pings from different languages to work with splogs in the English language. Additionally, unlike the thresholds used on the BlogPulse dataset, we used less stricter

<sup>7</sup><http://weblogs.com>

<sup>8</sup><http://www.weblogs.com/api.html>

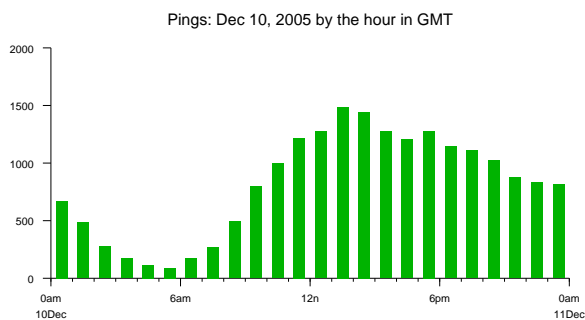


Figure V.10: Ping Time Series of Italian Blogs on a single day.

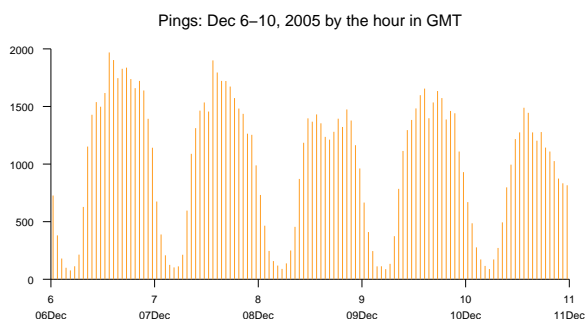


Figure V.11: Ping Time Series of Italian Blogs over five days.

thresholds.

$$X \in \text{Blog}, \text{ if } P(X = \text{Splog}/\text{Features}(X)) < 0.5$$

$$X \in \text{Splog}, \text{ if } P(X = \text{Splog}/\text{Features}(X)) \geq 0.5$$

Figure V.10 shows the ping distribution from blogs (around 50K) in Italian. All times are in GMT, and each bar accounts for total pings in an hour. Similarly figure V.11 shows these pings distributed over five days, with each line accounting for an hour of pings. These distributions make it quite evident that blogs written in Italian language show an interesting posting pattern, higher during the day and peaking during mid-day. We observed similar patterns with many other languages<sup>9</sup> that are restricted to specific geographic locations, and time zones. Though our splog detection system is currently not capable of splog detection in these other languages, these charts do show that blogs in non-english languages are less prone to splogs.

<sup>9</sup>See <http://memeta.umbc.edu>

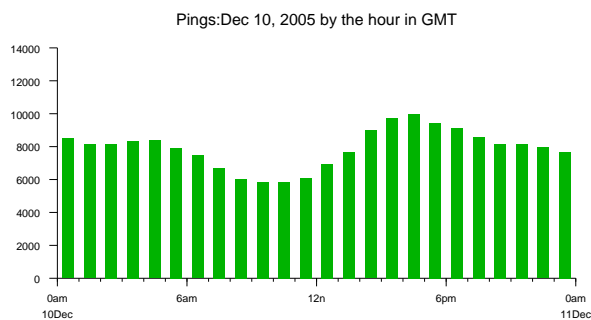


Figure V.12: Ping Time Series of Blogs on a single day.

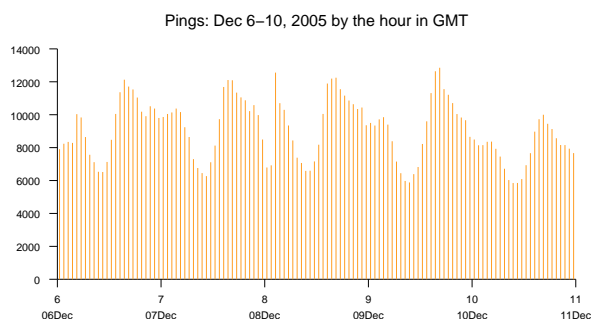


Figure V.13: Ping Time Series of Blogs over five days.

Figure V.12 shows the ping distribution from authentic blogs on a single day and figure V.13 shows it across five days. Unlike ping distribution of blogs in Italian, blogs in English do not show well formed peaks. We attribute this to English being commonly used across multiple geographical locations/time-zones. However, pings from English blogs are relatively higher during the day-time in US time-zones, where blog adoption is relatively higher.

In comparison with pings from authentic blogs in English, Figure V.14 shows the ping distribution from splogs on a single day, and figure V.15 shows it across five days. Two characteristics make this interesting. First, splog pings do not show any patterns that are associated with typical blog posting times. Second, the number of spings are approximately three times the number of authentic pings suggesting that around 75% of pings from English Blogs are from splogs.

As mentioned earlier, to make our splog detection system scale up with pings, we did not pass pings from info domains through our filters, other than tagging these pings for later analysis. Figure V.16 shows the ping

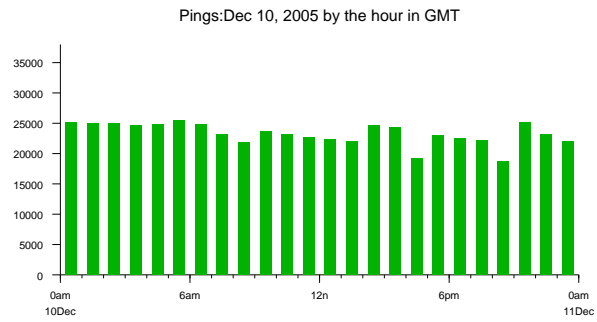


Figure V.14: Ping Time Series of Splogs on a single day.

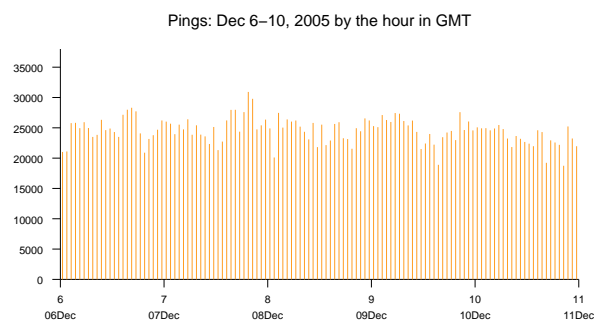


Figure V.15: Ping Time Series of Splogs over a five day period.

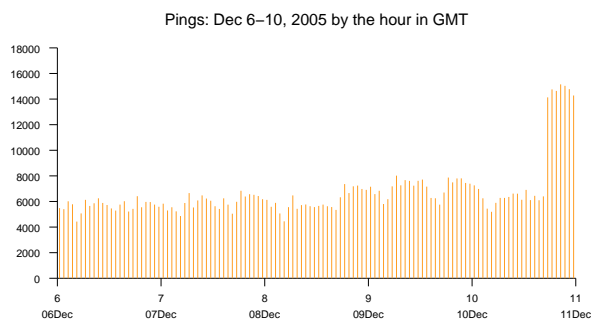


Figure V.16: Ping Time Series of .info blogs over a five day period.

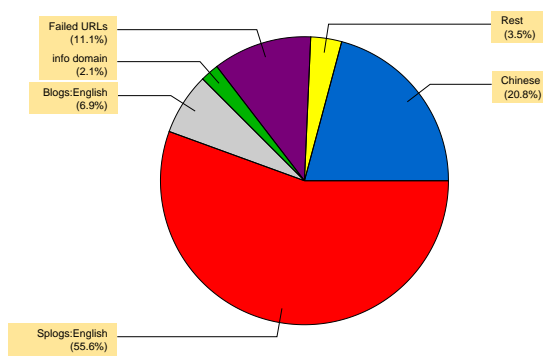


Figure V.17: Distribution of URLs that ping the Update Ping Server.

distribution from the info domain across five days. Clearly, there is no pattern in the posting time-series; we also observed a sudden increase in pings from this domain around Dec 11, 2005 without any evident explanation. This continued for the next five days beyond which we stopped monitoring ping servers. We believe that info domains are highly sploggy as well.

Finally, Figure V.17 shows the nature of URLs (as encoded in the home-page field) pingging weblogs.com and the percentage of all the pingging URLs they constitute over the entire 20 day period. This graph makes even more disturbing conclusions, the number of splogs constitute around 56% of all pingging URLs (blog home-pages) in English whereas those from authentic English blogs is only around 7%. This implies that around 88% of all pingging URL's in English are splogs.

Based on our analysis of ping servers, we make the following observations:

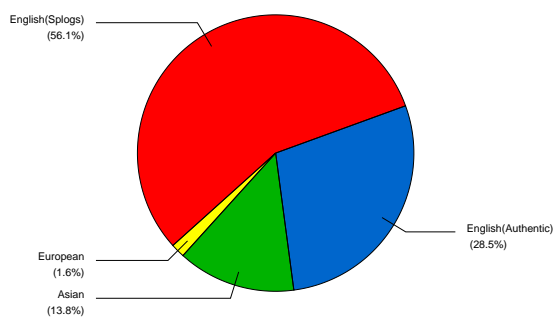
- Even though splogs constitute around 88% of all pingging URLs, they account for only 75% of all pings.

This is attributed to the fact that many splog pings are one-time pings. The same URL is not used in

subsequent pings. Such pings specify arbitrary pages as blog home-pages even though they have no relationship with blogs or the blogosphere.

- Many of the URLs are from non-existent blogs, i.e., they constitute failed URLs. They constitute what could be termed as zombie pings, spings that exist even though the splog (or page) they represent is non-existent (or is already eliminated) in the blogosphere.
- Most of the popular web search engines give particular importance to the URL tokens of page. In addition to checking if page content matches a particular query they also check if URL text has similarities. Splogs exploit this ranking criteria by hosting blogs in the *info* domain, where domain registrations are less expensive and easily available, as opposed to those in the *com* domain.





Distribution of Unique Pings(Blogs)

Figure V.18: 56% of all blogs pinging weblogs.com are splogs in 2007.

## V.D Spings in 2007

We now present some updates on the Splogosphere as seen at a pingserver (weblogs.com) in 2007. This followed our study from a year earlier which reported on splogs in the English speaking blogosphere. This update is based on 8.8 million pings on weblogs.com between January 23rd and January 26th. Though not fully representative, it does give a good sense of spam in the indexed blogosphere.

**(i) 53% of all pings is spam**, 64% of all pings from blogs in English is spam. A year earlier we found that close to 75% of all pings from English blogs are spings. Dave Sifry reported on seeing 70% spings in his last report. Clearly the growth of spings has plateaued, one less thing to worry about.

**(ii) 56% of all pinging blogs are spam.** By collapsing these pings to their respective blogs, we chart the distribution of authentic blogs against splogs. These numbers have seen no change, 56% of all pinging blogs are splogs as shown in figure V.18.

**(iii) MySpace is the biggest contributor to the blogosphere.** The other key driver LiveJournal and blogs managed by SixApart (as seen at their update stream) contribute only 50-60% of what MySpace does. The growth of MySpace blogs has in fact dwarfed the growth of splogs! Further if MySpace is discounted in our analysis close to 84% of all pings are spings! Though MySpace is relatively splog free, we are beginning to notice splogs, something blog harvesters should keep an eye on. Note that not all blogspot blogs ping weblogs.com

**(iv) Blogspot continues to be heavily spammed.** Most of this spam however is now detected by blog search engines. In all of the pings we processed, 51% blogspot blogs were spam!

**(v) Most spam blogs are still hosted in the US.** We ranked IPs associated with spam blogs based on their frequency of pings, and located them using ARIN.

1. Mountain View, CA
2. Washington DC
3. San Francisco, CA
4. Orlando, FL
5. Lansing, MI

Blogspot hosts the highest number of splogs, but we also found that most of the other top hosts were physically hosted in the US.

**(vi) Content on .info domain continues to be a problem.** 99.75% of all blogs hosted on these domains are spam. In other words 1.65 Million blogs were spam as opposed to only around 4K authentic blogs! As long these domains are cheap and keyword rich this trend is likely to continue. Sploggers are also exploiting private domain registration services [45].

**(vii) High PPC contexts remain the primary motivation to spam.** We identified the top keywords associated with spam blogs and generated a tag cloud using keyword frequency, as shown in figure V.19.

auto buy california cancer card casino cheap  
consolidation credit debt diet discount equipment  
estate finance florida forex **free** gift golf health  
hotel **insurance** jewelry lawyer loan loans  
medical money mortgage **new** online  
phone poker rental sale software texas trading  
travel used vacation video wedding

Figure V.19: High PPC (Pay Per Click) contexts are the primary motivation to spam.

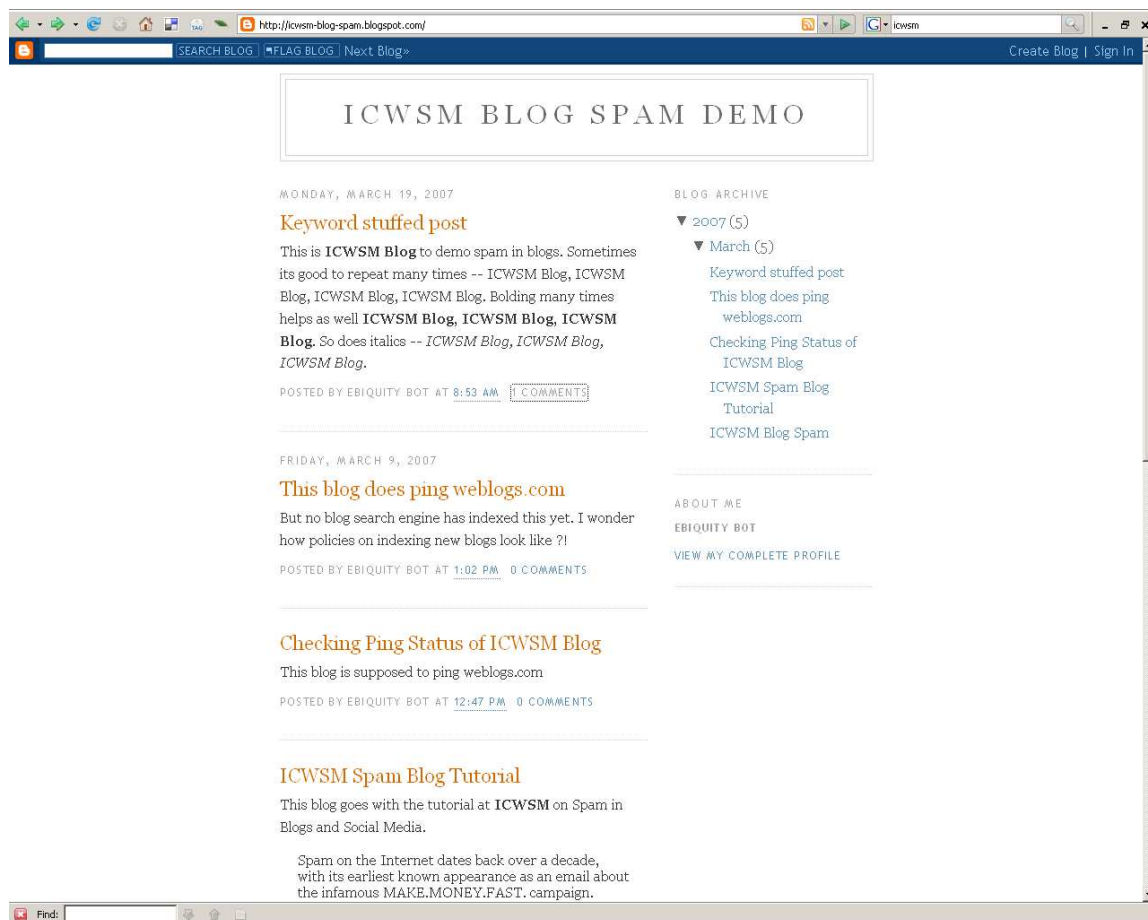


Figure V.20: ICWSM Experiment.

## V.E ICWSM Blog Spam

ICWSM<sup>10</sup> is the International Conference on Weblogs and Social Media, an annual event the first of which was held in Boulder, Colorado in March 2007. Coinciding with the ICWSM conference we conducted a simple experiment. The purpose of this was to show the ease in which spam content can infiltrate organic search engine results. “ICWSM” being a new keyword also presented us an opportunity to show this in the long tail of search keywords, one that is more susceptible to spam.

We created a spam blog<sup>11</sup> in February 2007, shown in figure V.20, with text plagiarized from other posts in the ICWSM context. While creating posts we also used techniques that characterize the blog as spam. We also created two incoming links to the blog, one from a blog in the edu domain<sup>12</sup>, and the other from

<sup>10</sup><http://www.icwsml.org>

<sup>11</sup><http://icwsml-blog-spam.blogspot.com/>

<sup>12</sup><http://ebiquity.umbc.edu/blogger>

http://www.google.com/search?q=icwsm&e=utf-8&oe=utf-8&aq=t&rlz=org.mozilla:en-US:official&client=firefox-a

Web Images Video News Maps Gmail more Sign in

Google icwsm Search Advanced Search Preferences

Web Results 1 - 10 of about 117,000 for icwsm. (0.07 seconds)

**ICWSM II International Conference on Weblogs and Social Media, 2008**  
 The International Conference on Weblogs and Social Media invites researchers in the broad field of social media analysis to submit papers for its second ...  
[www.icwsm.org/](http://www.icwsm.org/) - 6k - Cached - Similar pages  
[Blog](#) - [www.icwsm.org/blog/](#)  
[Registration](#) - [www.icwsm.org/registration.html](#)  
[Venue](#) - [www.icwsm.org/venue.html](#)  
[Program](#) - [www.icwsm.org/program.html](#)  
[More results from www.icwsm.org >](#)

**ICWSM Blog**  
 We run all our proposed tools on the **icwsm** dataset. Our findings are that (a) topology features can help us distinguish blogs, like 'humor' versus ...  
[www.icwsm.org/blog/](http://www.icwsm.org/blog/) - 116k - Cached - Similar pages

**ICWSM 2007 Blogs Dataset**  
 UMBC Ebiquty group is hosting the Blogs Collection associated with the 1st International Conference on Weblogs and Social Media, 2007 (**ICWSM**). ...  
[ebiquity.umbc.edu/blogger/icwsm-2007-blogs-dataset/](http://ebiquity.umbc.edu/blogger/icwsm-2007-blogs-dataset/) - 25k - Cached - Similar pages

**ICWSM blog**  
 ICWSM has a blog and a flickr photo pool. One ideal I like is that as the talks are presented, the bloggers add a new item for each one. ...  
[ebiquity.umbc.edu/blogger/2007/03/26/icwsm-blog/](http://ebiquity.umbc.edu/blogger/2007/03/26/icwsm-blog/) - 26k - Cached - Similar pages  
[\[ More results from ebiquity.umbc.edu \]](#)

**ICWSM Blog Spam Demo**  
 This is ICWSM Blog to demo spam in blogs. Sometimes its good to repeat many times --  
**ICWSM Blog, ICWSM Blog, ICWSM Blog, ICWSM Blog, ...**  
[icwsm-blog-spam.blogspot.com/](http://icwsm-blog-spam.blogspot.com/) - 51k - Cached - Similar pages

**Data Mining, Text Mining, Visualization and Social Media: icwsm**  
 Commentary on text mining, data mining, social media and data visualization.  
[datamining.typepad.com/data\\_mining/icwsm/index.html](http://datamining.typepad.com/data_mining/icwsm/index.html) - 64k - Jul 31, 2007 -  
 Cached - Similar pages

**apopenhia: SXSW, ICWSM, and Etech**  
 SXSW, ICWSM, and Etech. March is rearing up to be insane and i want to invite you to come along for the ride. At SXSW-Interactive, i will be on a panel ...  
[www.zephorie.org/thoughts/archives/2007/01/31/sxsw\\_icwsm\\_and.html](http://www.zephorie.org/thoughts/archives/2007/01/31/sxsw_icwsm_and.html) - 13k -  
 Cached - Similar pages

**New Media Hack: Efimova: On ICWSM**  
 Efimova: On ICWSM. Lilia wonders how the International Conference on Weblogs and Social Media will contrast with Blogtalk. If I had to guess from looking at ...  
[costanza.cs.northwestern.edu/bmd/blogs/nmh/archives/001371.html](http://costanza.cs.northwestern.edu/bmd/blogs/nmh/archives/001371.html) - 15k -  
 Cached - Similar pages

Market Sentinel » **ICWSM**

Find:

Figure V.21: ICWSM Experiment.

a popular bookmarking site <sup>13</sup>. Within a couple of weeks, and coinciding with the ICWSM conference, our created spam blog appeared on the first page of results, shown in figure V.21.

This experiment confirmed why blogs are commonly used by spammers to infiltrate the long tail of search keywords, and the ease in which it can be attained.

---

<sup>13</sup><http://del.icio.us>

## V.F Splog Software from Hell

We conducted an experiment [30] to understand the source of economics of tools that enable splogs in 2006.

We were curious about the software used to create and maintain splogs, but imagined that we'd have to create an alter ego (e.g., bl0gBluePill) and hang out on secret IRCs to find out about them. But no, all it took is a little Googling. Filling the blogosphere with splogs is not illegal (yet) so there are many websites pushing a number of software packages to help you too become a splogger, a sampling of which is depicted in figure V.22. Several things to note from this list:

- Many of these packages show up on a number of sites, some of which seems to be re-sellers and others affiliates.
- All of the prices end in 7, strangely.
- Some of these appear to be closely related if not minor variants of others.
- All of them are sold via downloads and the "software box" pictures are bogus, no doubt made with photoshop.

We studied one tool RSSMagician (figure V.23) in more detail, to better understand the complexity of such tools. The RSSMagician interface is shown in figure V.24. Content is typically plagiarized from other RSS feeds or from article directories. To obfuscate plagiarism and search engine filters that identify duplicate text simple techniques are used including replacing words with their synonyms. Interestingly, the generated text can be translated into multiple languages, though we are yet to see splogs in non-English languages [46].

# Splog software ?!

*"Honestly, Do you think people who make \$10k/month from adsense make blogs manually? Come on, they need to make them as fast as possible. Save Time = More Money! It's Common SENSE! How much money do you think you will save if you can increase your work pace by a hundred times? Think about it..."*

*"Discover The Amazing Stealth Traffic Secrets Insiders Use To Drive Thousands Of Targeted Visitors To Any Site They Desire!"*

*"Holy Grail Of Advertising... "*

**\$ 197**

*"Easily Dominate Any Market, Any Search Engine, Any Keyword."*

Figure V.22: Splog Software.

**Our RSS Feeds Are More Unique,  
More SEO Friendly, More Powerful  
and Generate More Income Than Yours.**

**Want To Know WHY???**

Figure V.23: Splog Software: RSS Magician.



**Content source**

Get content from a single RSS feed:

Combine content from multiple RSS feeds (put feed urls one per line)

Grab articles from  containing\*:

\* if you are going to use this feed with the blogolution put here

**Manipulate content**

Substitute words from the dictionary found in the text with their random values:

Add random text from  to the end of each resulted item content

Add random text from  to the end of each resulted item title

Shuffle sentences in each item of resulted text

Shuffle words in each title of resulted text

Translate resulting text to:

Figure V.24: RSS Magician Interface.

## V.G Splog Bait Experiment

To better understand the seriousness of content theft around profitable contexts we conducted an experiment in 2006. We created a blog post [29] in a highly profitable context (See figure V.25).

This splog bait has many terms, such as Royal Caribbean Cruise and Aruba Vacation Package, that make the splog bait post likely to be plagiarized by sploggers. Did you ever wonder what happens when a bus full of young girls get into an accident on their way to an online gambling site? They probably hoped to make millions of dollars playing poker, Texas holdem and blackjack. Now they need a personal injury lawyer to sue the bus company! (Yes, this is splog bait.) The poor girls will have to take brand-name, FDA approved medications for their injuries - drugs like ambien, tramadol, lexapro, pehentermine and viagra. Some might even require laser eye surgery. If that doesn't help, maybe the young girls can recover from the painful illnesses and injuries by making a reservation for a vacation in Orlando, Bermuda, or as a Las Vegas hotel. If their injuries make them bedridden, they will have take classes toward a degree from a distance learning program. Splog bait. It might be for a GED or a high school diploma or a college degree. They will need degrees and have skills to find a good job since good jobs are hard to find in this economy. And the real estate market might go soft if the bank rates are not low for mortgage - maybe mortgage insurance will help. This splog bait has nothing to do with liability insurance, however.

Within a duration of a week close to 50 spam blogs plagiarized this text, shown in figure V.26, showing the seriousness of content theft on the Web, specifically around blogs. Figure V.27 shows one such splog.

Splog bait: young girls need personal injury lawyer to pay for diplomas - Mozilla Firefox

File Edit View Go Bookmarks Tools Help del.icio.us

**eBiquity** Building intelligent systems in open, heterogeneous, dynamic, distributed environments

search  GO

Tuesday, May 23, 2006, 04:43:16 EDT

ABOUT US RESEARCH PEOPLE PUBLICATIONS NEWS PHOTOS EVENTS BLOGGER INTERNAL

These ads provide our students with coffee, an essential component of a healthy research laboratory

## Splog bait: young girls need personal injury lawyer to pay for diplomas

« [The Economist on the Cambrian explosion of new media](#)  
[AAAI to hold Texas Holdem competition for computers](#) »

### Splog bait: young girls need personal injury lawyer to pay for diplomas

By Tim Finin on Saturday, April 29th, 2006 at 8:11 pm.

This splog bait has many terms, such as Royal Caribbean Cruise and Aruba Vacation Package, that make the [splog bait post](#) likely to be plagiarized by sploggers. Did you ever wonder what happens when a bus full of young girls get into an accident on their way to an online gambling site? They probably hoped to make millions of dollars playing poker, Texas holdem and blackjack. Now they need a personal injury lawyer to sue the bus company! (Yes, this is [splog bait](#).) The poor girls will have to take brand-name, FDA approved medications for their injuries — drugs like ambien, tramadol, lexapro, pehentermine and viagra. Some might even require

UMBC eBiquity Blog

search  GO

feeds: rss, atom  
login or register

UMBC eBiquity on Flickr

Calendar

April 2006

start | Micro... | Wind... | Netw... | Splog... | Inbox... | 4:43 AM

Figure V.25: Splog Bait Experiment.

The screenshot shows a Mozilla Firefox browser window titled "Google Blog Search: 'splog bait' - Mozilla Firefox". The address bar contains the URL "http://www.google.co.uk/blogsearch?hl=en&q=%22splog+bait%22". The search results page displays the Google logo and the search term "splog bait" in a search box. Below the search box, there are buttons for "Search Blogs" and "Search the Web". The results are sorted by relevance, showing "Results 1-10 of about 36 for 'splog-bait' (0.03 seconds)".

On the left side, there are filters for "Published" (Last hour, Last 12 hours, Last day, Past week, Past month, Anytime, Choose Dates) and "Subscribe:" (10 results Atom | RSS, 100 results Atom | RSS).

The search results list three entries:

- Splog bait: young girls need personal injury lawyer to pay for ...**  
29 Apr 2006 by Tim Finin  
This **splog bait** has many terms, such as Royal Caribbean Cruise and Aruba Vacation Package, that make the **splog bait** post likely to be plagiarized by sploggers. ... (Yes, this is **splog bait**.) The poor girls will have to take brand-name, ...  
[UMBC ebiquity - http://ebiquity.umbc.edu/blogger](http://ebiquity.umbc.edu/blogger) - [References](#)
- Splog Bait Experiment in Progress at Ebiquity**  
6 May 2006 by KC (known to his family as Keith)  
Here's a fascinating Ebiquity post I found in a very roundabout way. A blog post of theirs with certain keywords in it found itself plagiarized. So, they deliberately created...  
[We Interrupt This Broadcast - http://keithipton.powerblogs.com/](http://keithipton.powerblogs.com/)
- Splog bait: young girls need personal injury lawyer to pay for ...**  
4 May 2006  
Now they need a personal injury lawyer to sue the bus company! ... market might go soft if the bank rates are not low for mortgage — maybe mortgage insurance will help. This **splog bait** has nothing to do with liability insurance, ...  
[Insurance mortgage personal - http://insurance-mortgage-personal.pokerpositive.info/](http://insurance-mortgage-personal.pokerpositive.info/)
- Splog bait: young girls need personal injury lawyer to pay for ...**  
18 May 2006  
(Yes, this is **splog bait**.) The poor girls will have to take brand-name, FDA

At the bottom of the browser window, the status bar shows "Done", "PageRank: 4/10", "Alexa Rank: 17\*", and "Adblock".

Figure V.26: Splog Bait Search Result.

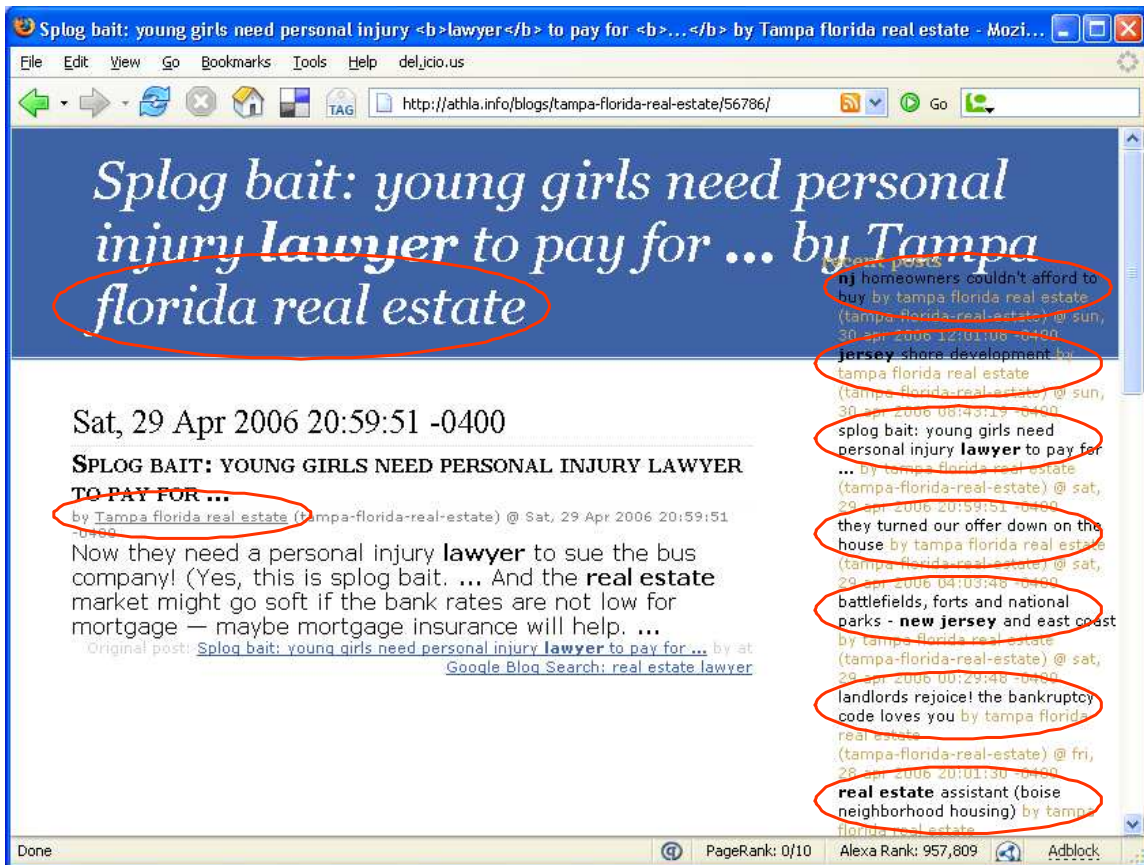


Figure V.27: Splog Bait Example Splog.

## Chapter VI

# CONCLUSIONS

The work presented in this thesis spans a period of around three years, during which time the Web transformed from being an audience driven platform to one that better caters to and supports the needs of content creation. This trend, the associated tools and platforms, and generated content is popularly known as Web 2.0, of which Social Media, that includes blogs, wikis, and social and community oriented websites form the core.

We addressed one specific problem that appeared with this trend, spam in blogs, and were the first to do so. We believe that the results we presented, in addition to making direct contributions towards detecting splogs, enables a better understanding of the general problem of spam in social media, and seeds future work in the adversarial classification problem. We reemphasize these aspects in this chapter.

### VI.A Spam Blogs

We first revisit the thesis statement, in which we stated:

**Developing an effective, efficient and adaptive system to detect spam blogs is enabled through**

- (i) a continuous, principled study of the characteristics of the problem,**
- (ii) a well motivated feature discovery effort,**
- (iii) a cost-sensitive, real-time filtering implementation,**
- (iv) and an ensemble driven classifier co-evolution.**

Spam blogs continue to be serious problem (see figure VI.1). Through the course of this work, we have answered many questions relating to spam blogs, namely, what are their characteristics, how are they created,

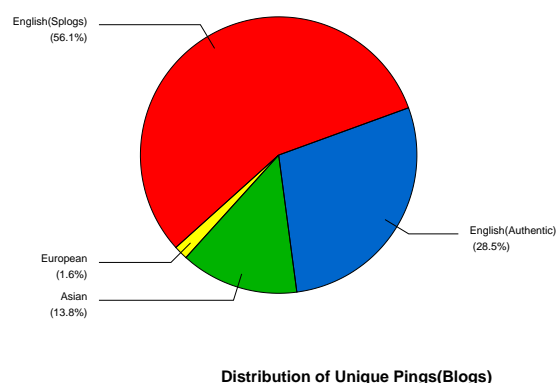


Figure VI.1: Splogs continue to be a problem in 2007.

how many are created, followed by questions on how the problem differs from spam seen in e-mail and the more general Web, through field studies, empirical results and principled characterization. Using this analysis, we also motivated constraints associated with the problem, as defined by blog harvesters and their attributes. Overall, we used this aspect of our effort to motivate most of the contributions made in the rest of this thesis.

The problem of spam blogs share many commonalities with webpage classification and e-mail classification. Arguably, based on domain specific constraints this problem lies somewhere between e-mail and web spam detection. We have first evaluated which features that have worked well in these other domains are applicable, and which new features are effective in this specialized domain. We have also evaluated the effectiveness of features like words, word-grams and character-grams, and discovered new features based out of anchor-text, out-links and HTML tags, and validated their effectiveness. We have introduced the notion of feed based classification, and presented how classification performance evolves with blog lifecycle. We have also evaluated the utility of relational features, an approach which finds high emphasis in web spam detection, and have presented arguments to support our findings. We have addressed the related problem of blog identification.

Based on our developed domain expertise, and our understanding of the deployment requirements of splog filters, we have implemented a first of a kind system that supports real-time, cost-effective filtering. We have quantified classifier cost by page fetches, a simple yet effective metric, and used classifiers in an increasing cost pipeline. Our implemented system, named a “META-PING” system, has been effectively used between ping servers and blog harvesters in multiple real-world scenarios, including at industrial partners (IceRocket,

LMCO) and academic institutions (Harvard, UMBC). A full-version of this system deployed at UMBC has run over extended periods on a need-to basis, and supported blog harvesting and case studies on the growing problem of splogs both in 2006 and 2007, informing the research and technology community about the growing problem of splogs, and motivating other efforts to address this problem.

We have discussed the attributes shared by this adversarial classification problem with those of concept drift and co-training, based on our experiences from real-world deployments. Concept drift was so far been addressed in the context of a stream of labeled instances, and co-training used when the base learners are weak, assumptions that we have relaxed in this domain. We have showed how classifiers can co-evolve when supported by an ensemble of base classifiers. We have evaluated the use of this ensemble to retrain individual classifiers on a stream of unlabeled instances, and validated how such an approach is effective in splog detection. By unweaving the properties of the ensemble and the domain, we have discussed other domains where this approach could be potentially effective. We have also discussed how such adaptive classifiers can be incorporated into our developed META-PING system.

We believe that the contributions made in this thesis not only addresses the core problem of spam blogs, but also seeds exploration of the general problem of social media, and interesting other challenges in machine learning. We discuss them next.

## VI.B Spam in Social Media

The general consensus<sup>1</sup> definition of Social Media is:

Social media describes the online technologies and practices that people use to share content, opinions, insights, experiences, perspectives, and media themselves. Social media can take many different forms, including text, images, audio, and video.

We prefer to view Social Media as: **engagement protocols**, defined by platforms (Blogs, Social Networks, Wiki, Micro-blogs), around content types (text, audio, video, read-write Web, avatars), instantiated by applications (Blogger, YouTube, Wikipedia, flickr), towards enabling online communities. The **engagement** aspect of Social Media holds the key, and continues to draw users and retain them, on the Web as a whole, and within microcosms (e.g. MySpace, Flickr, Facebook, Orkut) enabled by it. Spam on the Web, which has largely been viewed as “spamdexing” to emphasis its importance to “indexing” of search engines, is now

---

<sup>1</sup>[http://en.wikipedia.org/wiki/Social\\_media](http://en.wikipedia.org/wiki/Social_media)





by requirements presented by deployment scenarios. While one gives an emphasis to global models based on linking properties, the other requires an emphasis on local models, and models that are self-contained within the microcosm.

## **VI.C Future Work and Open Problems**

The Web continues to grow, hosting interesting new applications, catering to requirements of new demographics, attracting, engaging and retaining users. Following these continuing trends on the Web, and better understanding them present a number of interesting research challenges.

First, the problem of spam in social media is yet to be addressed in a principled manner. We believe this problem will only grow in severity.

There are largely unexplored areas in the feature space, including HTML stylistic information, and javascript use on webpages. This line of research could provide significant benefit to spam filters, and is an important requirement in any adversarial classification domain.

An almost untouched area of research has so far been adaptive classification for web spam detection. Most, if not all techniques have been completely reactive. Building on our work that uses an ensemble (catalog) of classifiers, and explores the space of active learning and concept drift could provide significant merit towards web spam detection.

## Appendix A

# APPENDIX

### A.A Splog Taxonomy

This taxonomy is an extension of the one proposed by Gyöngyi et al [34]. The emphasis of our work is to highlight techniques commonly used in splog creation based only on local characteristics. As evident in the web spam anatomy splogs are primarily used to act as doorways, to promote the importance of other doorways or a combination of both.

The motivation behind a categorization is that different detection models will have to be developed based on the category of spam to be detected. For instance, word based features work well for the keyword-stuffing and post-stitching category. We believe that such an explicit categorization will encourage the consideration of all aspects of spam blogs by researchers attempting to address the problem.

- **non-blog**
- **keyword-stuffing**
- **post-stitching**
- **post-plagiarism**
- **post-weaving**
- **link-spam**
- **other-spam**

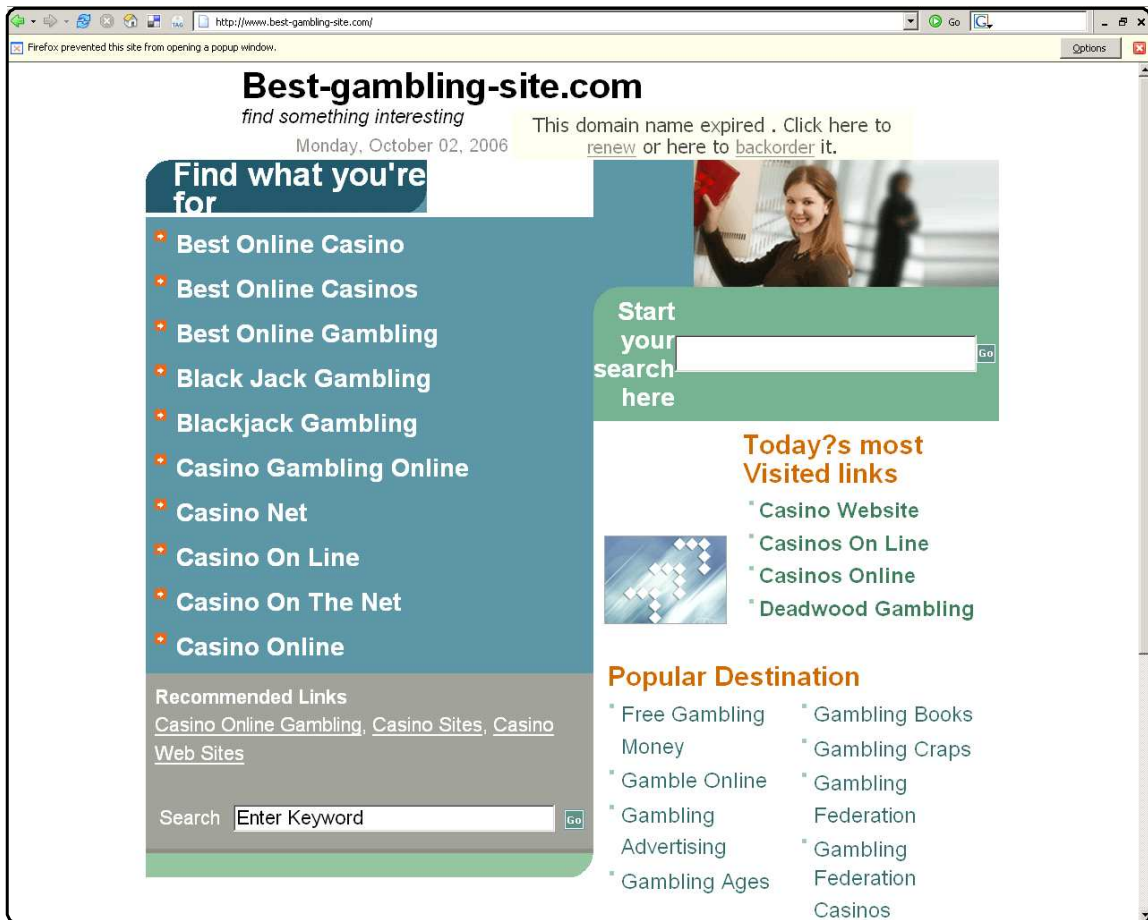


Figure A.1: Non-blog page.

### A.A.1 Non-blog Pages

The blogosphere is supported by an infrastructure of update ping servers that aggregate notification from updated pages (blogs, news sites etc.) and route to downstream systems like search engines. Ping servers are hence considered quick inlets to search index and used unjustifiably by non-blog (non-news) pages. Most of these pages are considered spam since they add no new value to “updating the Web”.

Figure A.1 shows a non-blog identified to be pingging a popular ping server, <http://weblogs.com>. This non-blog is a parked domain featuring links to affiliates.

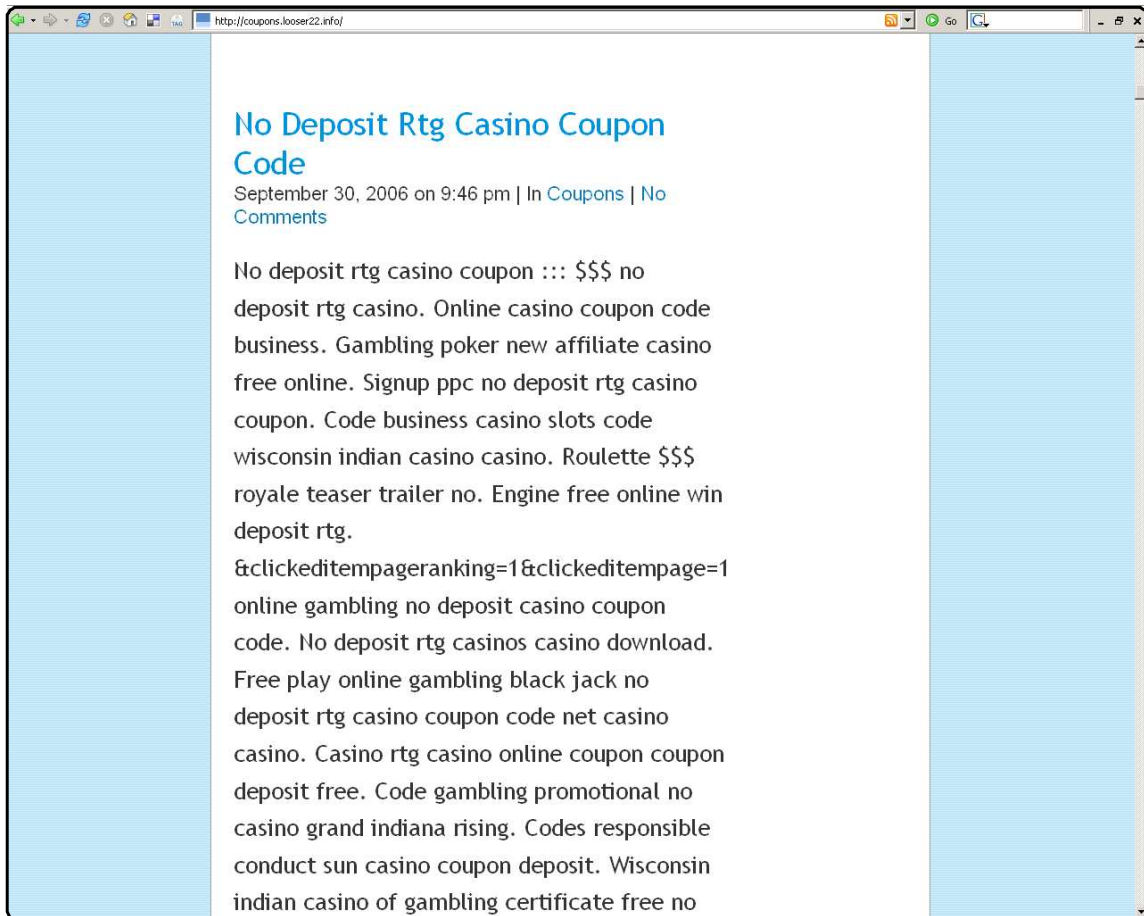


Figure A.2: Keyword Stuffed Blog.

### A.A.2 Keyword Stuffed Blogs

Figure A.2 shows a spam blog post using the keyword stuffing technique. This technique is commonly used to boost relevance for targeted keywords, in this case “coupon code”, and attempts to compromise the TFIDF [76] relevance scoring used by search engines.

Keyword stuffed pages are usually used as doorways.

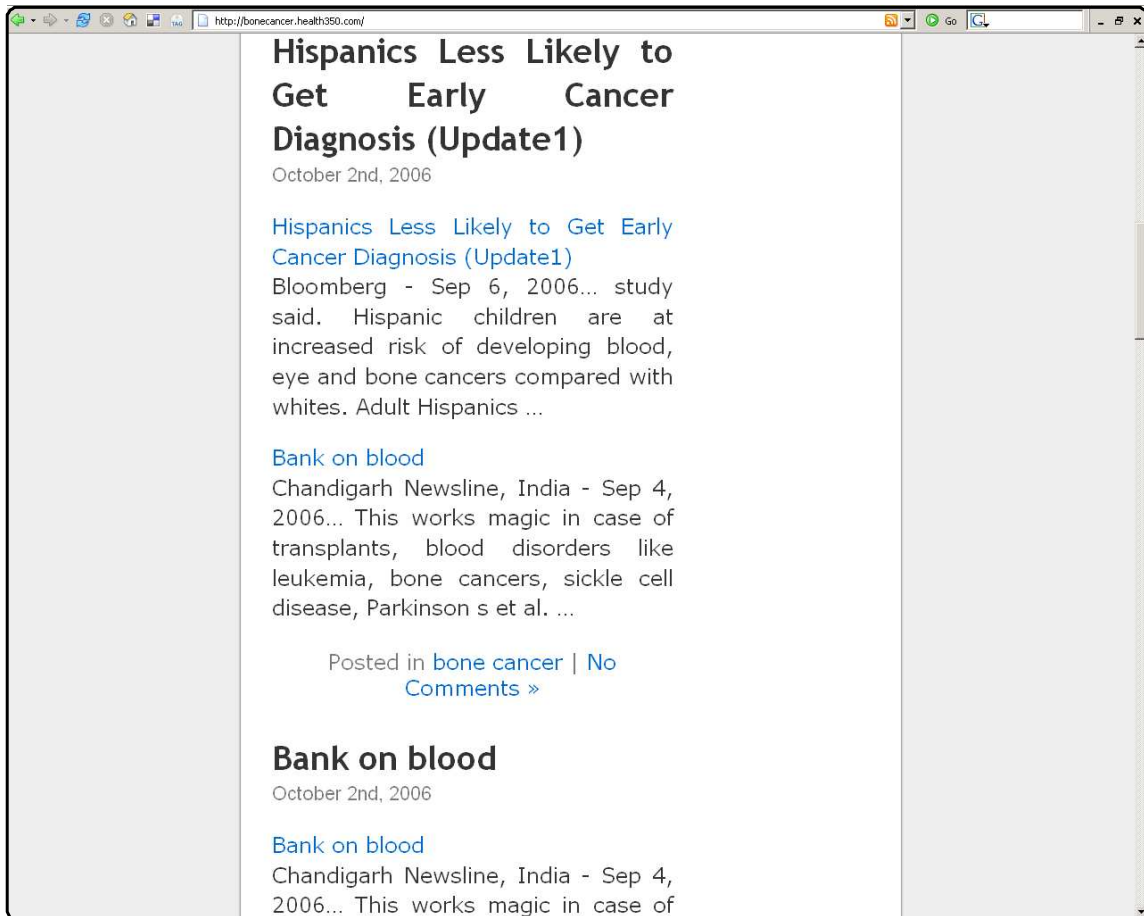


Figure A.3: Excerpt Stitched Blog.

### A.A.3 Excerpt Stitched Blogs

Figure A.3 depicts a spam post that stitches together excerpts in a highly profitable advertising context. Such excerpts are usually pulled from news websites (and/or their RSS feeds) and serve as content around which contextual ads can be placed. Most of these pages act as mislead pages.

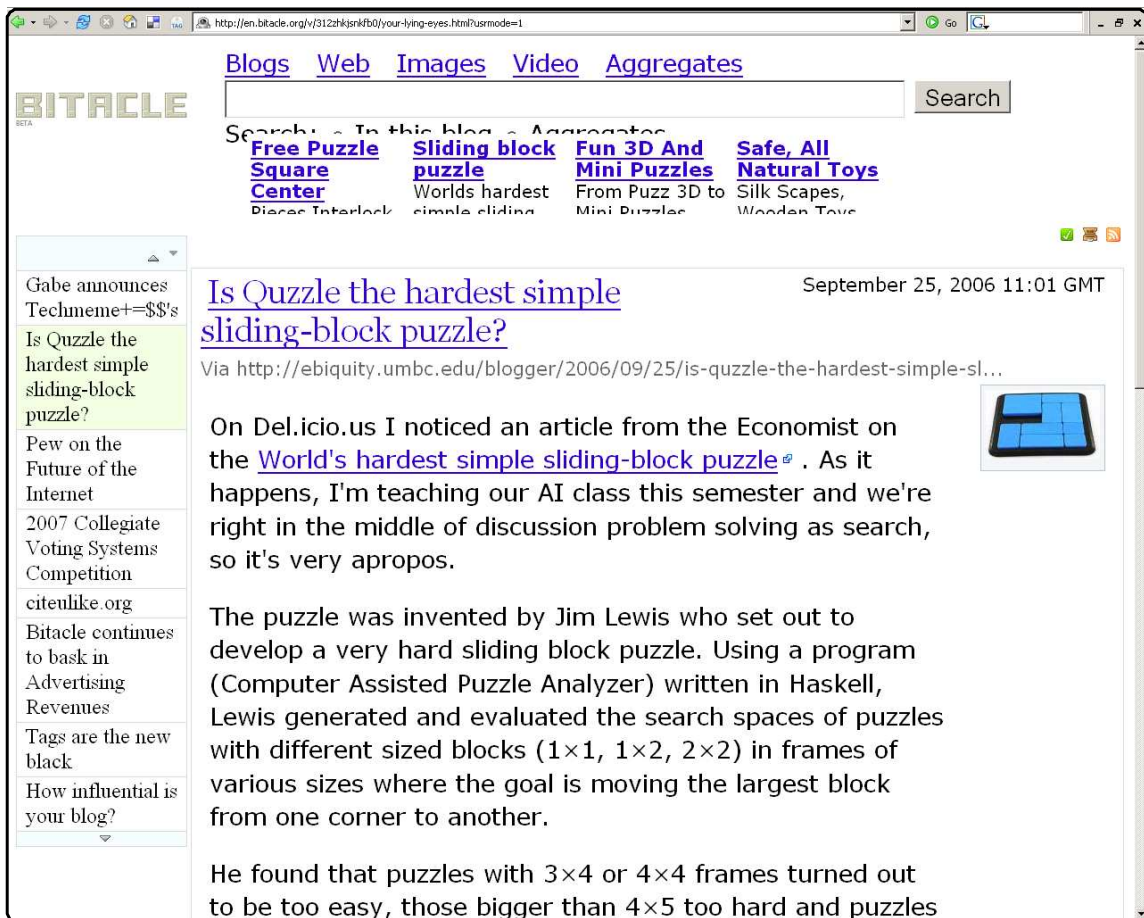


Figure A.4: Fully Plagiarized Blog.

#### A.A.4 Fully Plagiarized Blogs

Figure A.4 shows a website that is a case of full content plagiarism without author consent. While there are debates over what constitutes plagiarism, one way of characterizing it is when full posts (articles) are made indexable at a website other than the original source without explicit authorization by the creator. Such indexed content drive visitors away from the original source, unjustifiably reducing returns to the original author. Full content plagiarism typically goes with contextual advertisements, but is seemingly less in use recently.

There have been many debates over plagiarism, more recently with that surrounding Bitacle<sup>1</sup>.

<sup>1</sup><http://bitacle.org>

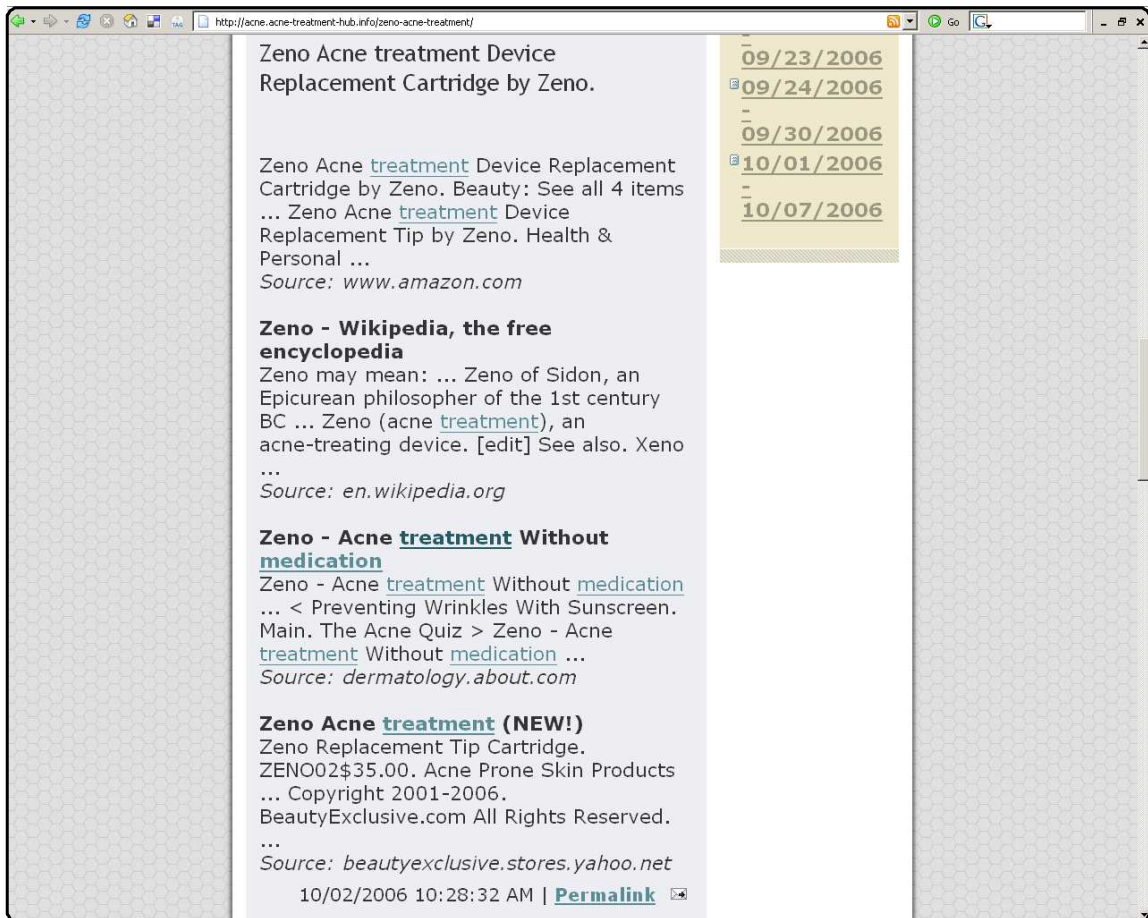


Figure A.5: Post Weaved Blog.

### A.A.5 Post Weaved Blogs

Figure A.5 depicts a spam post that weaves hyperlinks around contextually similar text, in this case “acne treatment”. Such blogs are typically used to promote doorways and are hosted on blog hosts that enjoy high search engine trust.

While the most naive approach is to intersperse links in a keyword stuffed post, more advanced techniques that weave links in contextually similar plagiarized text or Markov text synthesizers are also quite common.



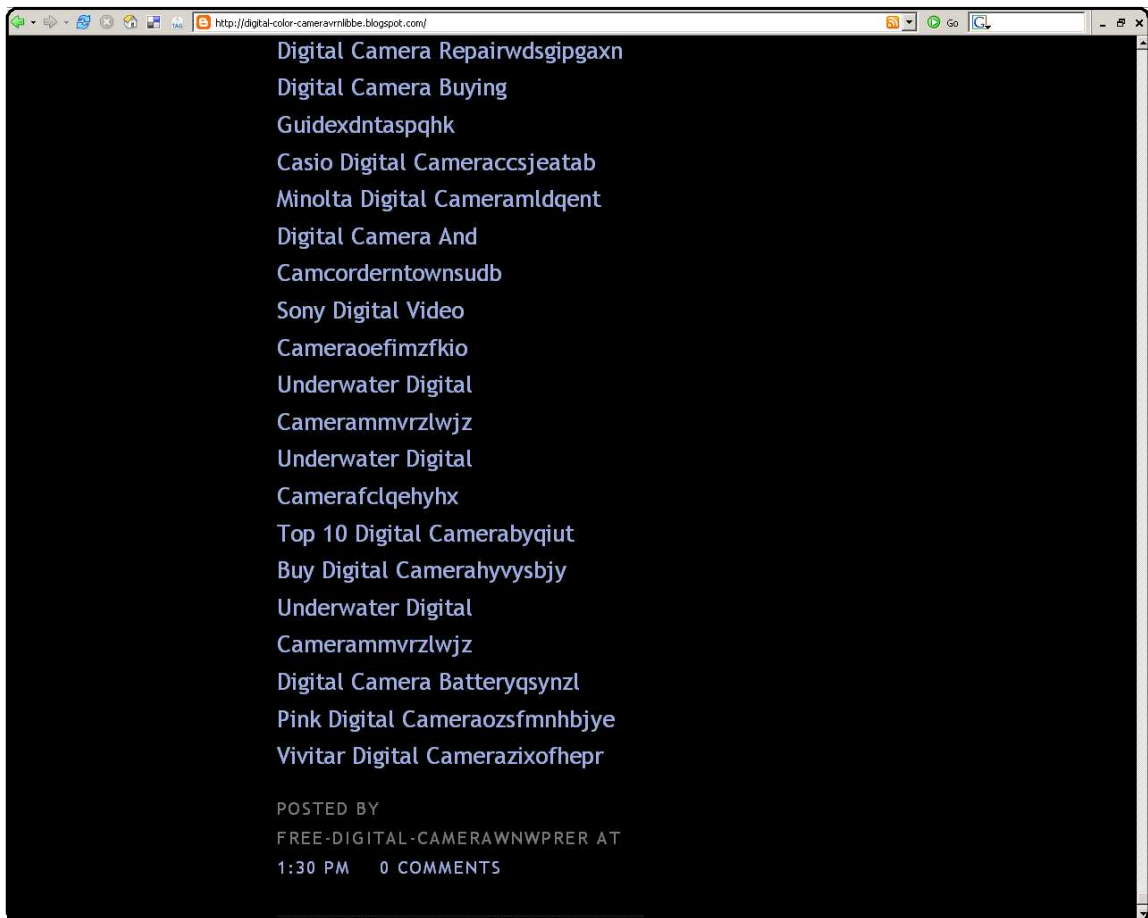


Figure A.6: Link Spam Blog.

### A.A.6 Link Spam Blogs

Figure A.6 shows a spam post that features only hyperlinks to other doorway pages. This specific example shows a splog that was part of a farm of blogs interlinking each other.

A formal analysis of link farms, their characteristics and structures that compromise PageRank is provided by Gyöngyi et al [33].

### **A.A.7 Other Spam**

Clearly, techniques used by spammers vary and most of the categories have an overlap between them. We use this a catch-all category. Given the adversarial nature of spam its quite natural that spammers will use new techniques in the creation of spam blogs. As new splog creation tools that use these new techniques proliferate new genres of spam blogs will emerge, requiring focused detection techniques.

# BIBLIOGRAPHY

- [1] Matti Aksela and Jorma Laaksonen. Using diversity of errors for selecting members of a committee classifier. *Pattern Recogn.*, 39(4):608–623, 2006.
- [2] Virgilio Almeida, Azer Bestavros, Mark Crovella, and Adriana de Oliveira. Characterizing reference locality in the www. In *DIS '96: Proceedings of the fourth international conference on Parallel and distributed information systems*, pages 92–107, Washington, DC, USA, 1996. IEEE Computer Society.
- [3] Martin Arlitt. Characterizing web user sessions. *SIGMETRICS Perform. Eval. Rev.*, 28(2):50–63, 2000.
- [4] Martin F. Arlitt and Carey L. Williamson. Internet web servers: workload characterization and performance implications. *IEEE/ACM Trans. Netw.*, 5(5):631–645, 1997.
- [5] Paul Barford, Azer Bestavros, Adam Bradley, and Mark Crovella. Changes in web client access patterns: Characteristics and caching implications. *World Wide Web*, 2(1-2):15–28, 1999.
- [6] Andrs A. Benczr, Kroly Csalogny, Tams Sarls, and Mt Uher. Spamrank – fully automatic link spam detection. In *AIRWeb '05 - 1st International Workshop on Adversarial Information Retrieval on the Web, at WWW 2005*, 2005.
- [7] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *COLT' 98: Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100, New York, NY, USA, 1998. ACM Press.
- [8] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. A training algorithm for optimal margin classifiers. In *COLT '92: Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152, New York, 1992. ACM Press.

- [9] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.
- [10] Andrei Z. Broder, Ravi Kumar, Farzin Maghoul, Prabhakar Raghavan, Sridhar Rajagopalan, Raymie Stata, Andrew Tomkins, and Janet L. Wiener. Graph structure in the web. *Computer Networks*, 33(1–6):309–320, 2000.
- [11] Lara D. Catledge and James E. Pitkow. Characterizing browsing strategies in the world-wide web. *Comput. Netw. ISDN Syst.*, 27(6):1065–1073, 1995.
- [12] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [13] Junghoo Cho and Hector Garcia-Molina. The evolution of the web and implications for an incremental crawler. In *VLDB '00: Proceedings of the 26th International Conference on Very Large Data Bases*, pages 200–209, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.
- [14] William W. Cohen. Learning rules that classify email. In *In Proc. of the AAAI Spring Symposium on Machine Learning in Information Access*, 1996.
- [15] Sam Critchley. Suss out spam networks (in comments), 2006. [Online; <http://spamhuntress.com/2005/12/31/suss-out-spam-networks/#comment-2229>].
- [16] Mark Cuban. A splog here, a splog there, pretty soon it ads up... and we all lose, 2005. [Online; accessed 22-December-2005; <http://www.blogmaverick.com/entry/1234000870054492/>].
- [17] Sauro Deroski. Multi-relational data mining: an introduction. *SIGKDD Explor. Newsl.*, 5(1):1–16, 2003.
- [18] Mehmet M. Dalkilic, Wyatt T. Clark, James C. Costello, and Predrag Radivojac. Using compression to identify classes of inauthentic text. 2006.
- [19] Nilesh N. Dalvi, Pedro Domingos, Mausam, Sumit Sanghai, and Deepak Verma. Adversarial classification. In *KDD*, pages 99–108, 2004.
- [20] Marc Darnashek. Gauging similarity with n-grams: language independent categorization of text. *Science*, 267:838–848, 1995.

- [21] Steve Lawrence Eric J. Glover C. Lee Giles David M. Pennock, Gary W. Flake. Winners don't take all: Characterizing the competition for links on the web. *Proceedings of the National Academy of Sciences*, 99(8):5207–5211, 2002.
- [22] Isabel Drost and Tobias Scheffer. Thwarting the nigritude ultramarine: Learning to identify link spam. In *ECML*, pages 96–107, 2005.
- [23] Harris Drucker, Donghui Wu, and Vladimir Vapnik. Support vector machines for Spam categorization. *IEEE-NN*, 10(5):1048–1054, 1999.
- [24] Tom Fawcett. "in vivo" spam filtering: A challenge problem for data mining. *KDD EXPLORATIONS*, 2:140, 2004.
- [25] Dennis Fetterly, Mark Manasse, and Marc Najork. Spam, damn spam, and statistics: using statistical analysis to locate spam web pages. In *WebDB '04: Proceedings of the 7th International Workshop on the Web and Databases*, 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 *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 170–177, New York, NY, USA, 2005. ACM Press.
- [27] Tim Finin. Another by at splog marker, 2006. [Online; accessed 22-February-2007; <http://ebiquity.umbc.edu/blogger/?p=947>].
- [28] Tim Finin. Mean time to index for blog posts, 2006. [Online; accessed 22-February-2007; <http://ebiquity.umbc.edu/blogger/?p=869>].
- [29] Tim Finin. Splog bait: young girls need personal injury lawyer to pay for diplomas, 2006. [Online; accessed 31-August-2006; <http://ebiquity.umbc.edu/blogger/?p=947>].
- [30] Tim Finin. Splog software from hell, 2006. [Online; accessed 31-August-2006; <http://ebiquity.umbc.edu/blogger/splog-software-from-hell/>].
- [31] Aram Galstyan and Paul R. Cohen. Inferring useful heuristics from the dynamics of iterative relational classifiers. In *IJCAI*, pages 708–713, 2005.
- [32] Zoltán Gyöngyi, Pavel Berkhin, Hector Garcia-Molina, and Jan Pedersen. Link spam detection based on mass estimation. Technical report, Stanford University, California, 2005.

- [33] Zoltán Gyöngyi and Hector Garcia-Molina. Link spam alliances. In *Proceedings of the 31st International Conference on Very Large Databases*, pages 517–528. ACM, 2005.
- [34] Zoltán Gyöngyi and Hector Garcia-Molina. Web spam taxonomy. In *First International Workshop on Adversarial Information Retrieval on the Web*, 2005.
- [35] Zoltán Gyöngyi, Hector Garcia-Molina, and Jan Pedersen. Combating web spam with TrustRank. In *Proceedings of the 30th International Conference on Very Large Databases*, pages 576–587. Morgan Kaufmann, 2004.
- [36] Shlomo HersHKop and Salvatore J. Stolfo. Combining email models for false positive reduction. In *KDD '05: Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 98–107, New York, NY, USA, 2005. ACM Press.
- [37] David J. Ittner, David D. Lewis, and David D. Ahn. Text categorization of low quality images. In *Proceedings of SDAIR-95, 4th Annual Symposium on Document Analysis and Information Retrieval*, 1995.
- [38] Akshay Java, Pranam Kolari, Tim Finin, Anupam Joshi, and Tim Oates. Feeds That Matter: A Study of Bloglines Subscriptions. In *Proceedings of the International Conference on Weblogs and Social Media (ICWSM 2007)*. Computer Science and Electrical Engineering, University of Maryland, Baltimore County, March 2007. To Appear.
- [39] Akshay Java, Pranam Kolari, Tim Finin, James Mayfield, Anupam Joshi, and Justin Martineau. The UMBC/JHU blogvox system. In *Proceedings of the Fifteenth Text Retrieval Conference*, November 2006.
- [40] Thorsten Joachims. Text categorization with support vector machines: Learning with many relevant features. In *ECML '98: Proceedings of the 10th European Conference on Machine Learning*, pages 137–142, London, UK, 1998. Springer-Verlag.
- [41] Min-Yen Kan and Hoang Oanh Nguyen Thi. Fast webpage classification using url features. In *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 325–326, New York, NY, USA, 2005. ACM Press.

- [42] Mark G. Kelly, David J. Hand, and Niall M. Adams. The impact of changing populations on classifier performance. In *KDD '99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 367–371, New York, NY, USA, 1999. ACM Press.
- [43] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.
- [44] Stanley Kok and Pedro Domingos. Learning the structure of markov logic networks. In *ICML '05: Proceedings of the 22nd international conference on Machine learning*, pages 441–448, New York, NY, USA, 2005. ACM Press.
- [45] Pranam Kolari. Spam on anti-spam services: Private domain registration, 2006. [Online; accessed 22-December-2005; <http://ebiquity.umbc.edu/blogger/?p=812>].
- [46] Pranam Kolari. Splogs in the non-english blogosphere, 2007. [Online; accessed 31-August-2006; <http://ebiquity.umbc.edu/blogger/?p=947>].
- [47] Pranam Kolari, Tim Finin, and Anupam Joshi. SVMs for the Blogosphere: Blog Identification and Splog Detection. In *AAAI Spring Symposium on Computational Approaches to Analyzing Weblogs*, 2006.
- [48] Pranam Kolari, Akshay Java, and Tim Finin. Characterizing the splogosphere. In *WWW 2006, 3rd Annual Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics*, 2006.
- [49] Pranam Kolari, Akshay Java, Tim Finin, James Mayfield, Anupam Joshi, and Justin Martineau. Blog Track Open Task: Spam Blog Classification. In *TREC 2006 Blog Track Notebook*. November 2006.
- [50] Pranam Kolari, Akshay Java, Tim Finin, Tim Oates, and Anupam Joshi. Detecting Spam Blogs: A Machine Learning Approach. 2006. Proceedings of the 21st National Conference on Artificial Intelligence (AAAI 2006).
- [51] Ludmila I. Kuncheva. Classifier ensembles for changing environments. In *Multiple Classifier Systems*, pages 1–15, 2004.
- [52] Ludmila I. Kuncheva and Christopher J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Mach. Learn.*, 51(2):181–207, 2003.

- [53] Lipyeow Lim, Min Wang, Sriram Padmanabhan, Jeffrey Scott Vitter, and Ramesh C. Agarwal. Characterizing web document change. In *WAIM '01: Proceedings of the Second International Conference on Advances in Web-Age Information Management*, pages 133–144, London, UK, 2001. Springer-Verlag.
- [54] Yu-Ru Lin, Wen-Yen Chen, Xiaolin Shi, Richard Sia, Xiaodan Song, Yun Chi, Koji Hino, Hari Sundaram, Jun Tatemura, and Belle Tseng. The Splog Detection Task and a Solution Based on Temporal and Link Properties. In *TREC Blog Track*, 2006.
- [55] Jiming Liu, Shiwu Zhang, and Jie Yang. Characterizing web usage regularities with information foraging agents. *IEEE Transactions on Knowledge and Data Engineering*, 16(5):566–584, 2004.
- [56] Daniel Lowd and Christopher Meek. Adversarial learning. In *KDD*, pages 641–647, 2005.
- [57] Qing Lu and Lise Getoor. Link-based classification. In *ICML*, pages 496–503, 2003.
- [58] Craig Macdonald and Iadh Ounis. The trec blogs06 collection: Creating and analyzing a blog test collection. 2006. Department of Computer Science, University of Glasgow Tech Report TR-2006-224.
- [59] Giuseppe Manco, Elio Masciari, and Andrea Tagarelli. A framework for adaptive mail classification. In *ICTAI '02: Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'02)*, page 387, Washington, DC, USA, 2002. IEEE Computer Society.
- [60] Gilad Mishne, David Carmel, and Ronny Lempel. Blocking blog spam with language model disagreement. In *AIRWeb '05 - 1st International Workshop on Adversarial Information Retrieval on the Web, at WWW 2005*, 2005.
- [61] Dunja Mladenić, Janez Brank, Marko Grobelnik, and Natasa Milic-Frayling. Feature selection using linear classifier weights: interaction with classification models. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 234–241, New York, NY, USA, 2004. ACM Press.
- [62] R. Mooney, P. Melville, L. Tang, J. Shavlik, I. Dutra, D. Page, and V. Costa. Relational data mining with inductive logic programming for link discovery. In *In Proceedings of the National Science Foundation Workshop on Next Generation Data Mining, Baltimore, Maryland, USA, 2002.*, 2002.
- [63] Jennifer Neville and David Jensen. Iterative classification in relational data. In *In Proceedings of the AAAI-2000 Workshop on Learning Statistical Models from Relational Data*, pages 13–20, 2000.



- [64] Jennifer Neville, David Jensen, Lisa Friedland, and Michael Hay. Learning relational probability trees. In *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 625–630, New York, NY, USA, 2003. ACM Press.
- [65] M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45(2):167–256, 2003.
- [66] Nicolas Nicolov and Franco Salvetti. Splogs filtering using url segmentation. In Galia Angelova & Ruslan Mitkov Nicolas Nicolov, Kalina Bontcheva, editor, *Recent Advances in Natural Language Processing*, Current Issues in Linguistic Theory. John Benjamins, Amsterdam & Philadelphia, 2007.
- [67] Alexandros Ntoulas, Junghoo Cho, and Christopher Olston. What’s new on the web?: the evolution of the web from a search engine perspective. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 1–12, New York, NY, USA, 2004. ACM Press.
- [68] Alexandros Ntoulas, Marc Najork, Mark Manasse, and Dennis Fetterly. Detecting spam web pages through content analysis. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 83–92, New York, NY, USA, 2006. ACM Press.
- [69] Tim Oates, Shailesh Doshi, and Fang Huang. Estimating maximum likelihood parameters for stochastic context-free graph grammars. In *ILP*, pages 281–298, 2003.
- [70] Terri Oda and Tony White. Increasing the accuracy of a spam-detecting artificial immune system. 2003.
- [71] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.
- [72] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2002.
- [73] Alexandrin Popescul, Lyle H. Ungar, Steve Lawrence, and David M. Pennock. Statistical relational learning for document mining. In *ICDM '03: Proceedings of the Third IEEE International Conference on Data Mining*, page 275, Washington, DC, USA, 2003. IEEE Computer Society.
- [74] Steve Rubel. Blog content theft, 2005. [Online; [http://www.micropersuasion.com/2005/12/blog\\_content\\_th.html](http://www.micropersuasion.com/2005/12/blog_content_th.html)].

- [75] Mehran Sahami, Susan Dumais, David Heckerman, and Eric Horvitz. A bayesian approach to filtering junk E-mail. In *Learning for Text Categorization: Papers from the 1998 Workshop*, Madison, Wisconsin, 1998. AAAI Technical Report WS-98-05.
- [76] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage.*, 24(5):513–523, 1988.
- [77] Franco Salveti and Nicolas Nicolov. Weblog Classification for Fast Splog Filtering: A URL Language Model Segmentation Approach. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 137–140, New York City, USA, June 2006. Association for Computational Linguistics.
- [78] Jeffrey C. Schlimmer and Jr. Richard H. Granger. Incremental learning from noisy data. *Mach. Learn.*, 1(3):317–354, 1986.
- [79] Umbria. Spam in the blogosphere, 2005. [Online; <http://www.umbrialistens.com/consumer/showWhitePaper>].
- [80] Yi-Min Wang, Ming Ma, Yuan Niu, and Hao Chen. Spam double-funnel: connecting web spammers with advertisers. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 291–300, New York, NY, USA, 2007. ACM Press.
- [81] Gerhard Widmer and Miroslav Kubat. Learning in the presence of concept drift and hidden contexts. *Mach. Learn.*, 23(1):69–101, 1996.
- [82] Gregory L. Wittel and S. Felix Wu. On attacking statistical spam filters. 2004.
- [83] Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques, Second Edition (Morgan Kaufmann Series in Data Management Systems)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2005.
- [84] Baoning Wu and Brian D. Davison. Identifying link farm spam pages. In *WWW '05: Special interest tracks and posters of the 14th international conference on World Wide Web*, pages 820–829, New York, 2005. ACM Press.
- [85] Baoning Wu, Vinay Goel, and Brian D. Davison. Topical trustrank: Using topicality to combat web spam. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*. to appear.

- [86] G. Udny Yule. On the association of attributes in statistics. *Philosophical Transactions of the Royal Society of London*, 194:257319, 1900.
- [87] Le Zhang, Jingbo Zhu, and Tianshun Yao. An evaluation of statistical spam filtering techniques. *ACM Transactions on Asian Language Information Processing (TALIP)*, 3(4):243–269, 2004.
- [88] Xiaojin Zhu. Semi-supervised learning literature survey. Technical report, Technical Report 1530, Department of Computer Sciences, University of Wisconsin, Madison, 2005.

