

Content-based Web Spam Detection

Gordon V. Cormack

University of
Waterloo



When you have a hammer ...

Everything looks like a nail!

Hammer:

- Content-based *email* spam filters

 - Dynamic Markov Compression (DMC)

 - Orthogonal Sparse Bigrams (OSBF-Lua)

- Stacking multiple filter results

 - Combining results with logistic regression

Nail:

- The Web Spam Challenge





Detailed results Task B:

<i>Rank Task B</i>	<i>Average AUC</i>	<i>Team</i>
1	0.946469	Gordon Cormack University of Waterloo, Canada
2	0.918251	Nikolaos Trogkanis , National Technical University of Athens, Greece Georgios Paliouras , National Center of Scientific Research "Demokritos", Greece
3	0.907398	Kushagra Gupta, Vikrant Chaudhary, Nikhil Marwah, Chirag Taneja Inductis India Pvt Ltd
4	0.899241	D'yakonov Alexander Moscow State University, Russia
5	0.893333	Wenyuan Dai Apex Data & Knowledge Management Lab, Shanghai Jiao Tong University

DMC *colors* spam red, non-spam green

Spam

www.lmg2-dvd.co.uk
www.f2films.co.uk
www.gifthunt.co.uk
www.home-loans-online.co.uk
www.abfinance.co.uk
www.insurance-quote.co.uk
irish-swingers.connect4fun.co.uk

Non-spam
(normal)

lib1.leeds.ac.uk
www.babyfriendly.org.uk
www.learning-services.org.uk
www.preparingforemergencies.gov.uk
www.hintsandthings.co.uk
www.guardian.co.uk
www.psnc.org.uk

DMC applied to what?

Text! (actually, any stream of bits)

hostname

of host to be classified

of incoming links

of outgoing links

html content

page(s) on host (*which pages?*)

text, markup, formatting (*just a bit stream to DMC*)

excerpts of pages (*first or last 2500 bytes*)

http server response

10 filters in total (9 DMC, 1 OSBF-Lua)

Combining the 10 filter runs

Each run yields a *spamminess score* s_n for each host

Convert to *log-odds* L_n using training data

$$L_n = \log\left(\frac{|\{i \mid s_i \leq s_n \text{ and ith message is spam}\}| + \epsilon}{|\{i \mid s_i \geq s_n \text{ and ith message is ham}\}| + \epsilon}\right)$$

Naïve combination

sum L_n over all runs

Slightly better combination

logistic regression to compute weighted sum

Results (10-fold cross validation)

<i>Method</i>	<i>AUC</i>	<i>F₁</i>	<i>weight</i>
homebig	.939	.634	.064
homebig.tail	.938	.626	.056
httponly	.867	.481	.124
bodyonly	.933	.627	.184
wget	.942	.622	.121
wget.tail	.942	.619	.135
wget.osbf	.929	.635	.200
hostname	.864	.424	.095
ingraph	.952	.639	.383
outgraph	.834	.289	.021
log-odds	.975	.796	-
logistic	.980	.803	-

Collaborative Proposal

Combine all Web Spam Challenge Submissions!

Really, really naïve approach

spamminess = # spam votes among participants

Naïve approach

requires training results for log-odds calculation

Logistic regression

requires training results for weight calculation

Let's build the ultimate filter

send me your data (training + test)

gvcormac@uwaterloo.ca

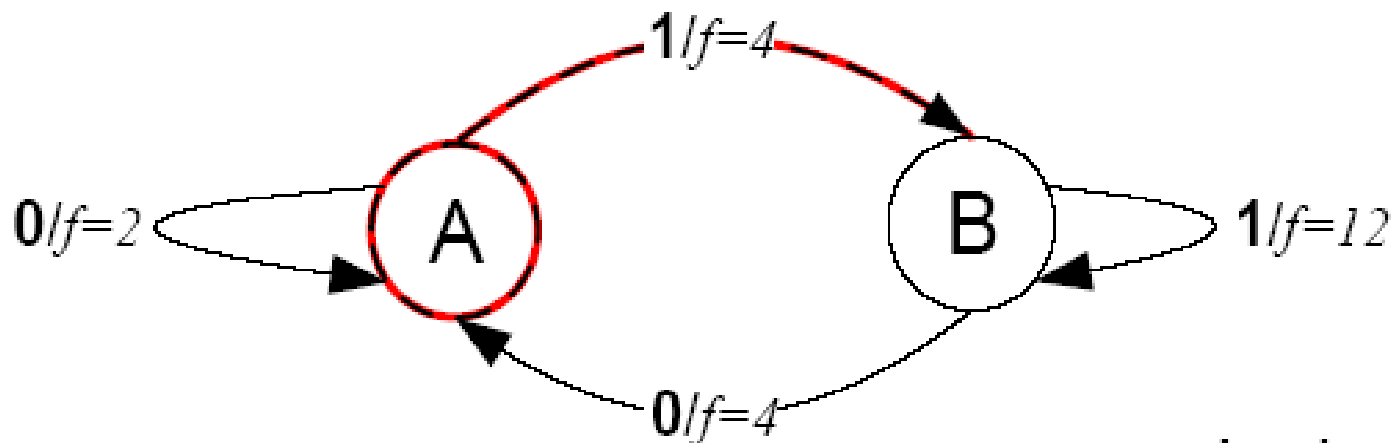
Content-based Web Spam Detection

Gordon V. Cormack

University of
Waterloo



Dynamic Markov model (DMC)



This example implements a 1st order Markov model

A means *following 0*; **B** means *following 1*

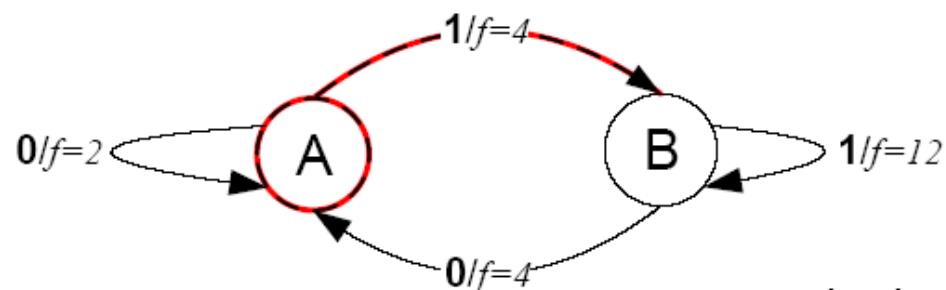
Outputs f on edges are frequencies

$$\text{Prob}(1 \text{ following } \mathbf{A}) = 4 / (2 + 4) = 0.667$$

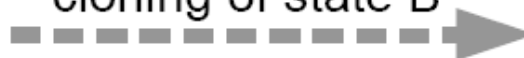
f incremented after each transition

DMC State Cloning

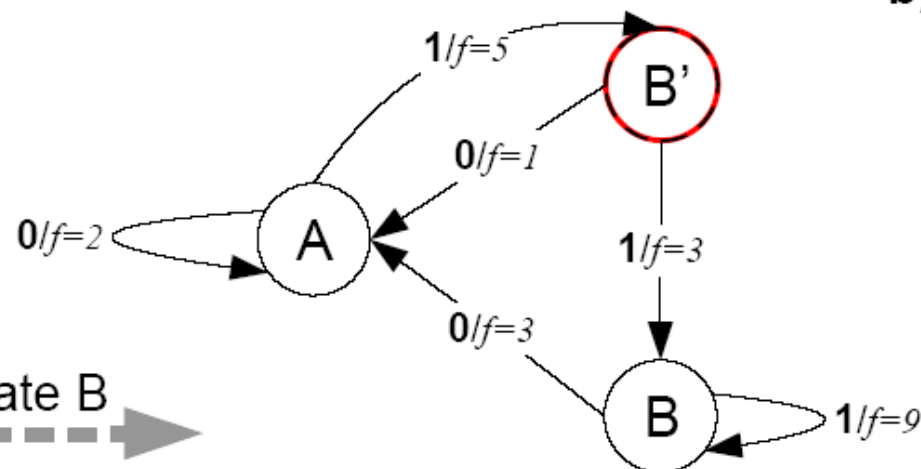
a)



cloning of state B



b)



State A, input 1, Prob 0.67

B cloned to create B'

B visited 16 times previously

f divided in 4:12 ratio in proportion to previous visits

4 from A; 12 from elsewhere

B should be cloned because it is visited from distinct contexts several times

f incremented as usual