



# Breaking Bad: Detecting malicious domains using word segmentation

Wei Wang, Kenneth E. Shirley

AT&T Security Research Center, AT&T Labs Research

2015-05-21, WEB 2.0 Security & Privacy 2015

# Who We Are

Our location:  
33 Thomas Street, New York, NY



Wei Wang  
AT&T Security  
Research



Kenny Shirley  
AT&T Statistics  
Research



# Our main question

How well can we predict whether a website is malicious  
*using only information from its domain name?*

Explore the solution with

- (1) word segmentation and
- (2) machine learning



# Toy example

Is it safe to visit the domain [safestplaceintown.com](http://safestplaceintown.com)?

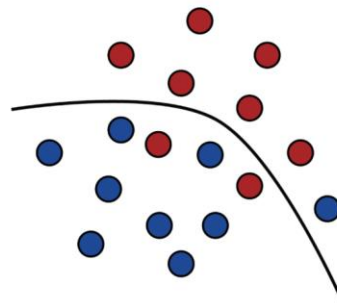
“safest”

“place”

“in”

“town”

some other predictors



17.2% chance of  
being malicious

(1) word segmentation

(2) machine learning

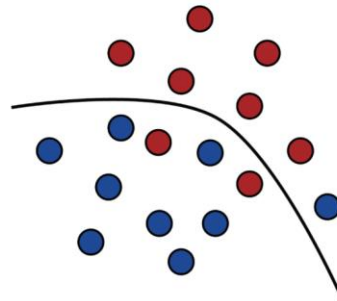
(3) make prediction



# Toy example

What about the domain [freecashandnikejerseys.com](http://freecashandnikejerseys.com)?

“free”  
“cash”  
“and”  
“nike”  
“jerseys”  
some other predictors



99.9% chance of  
being malicious

(1) word segmentation

(2) machine learning

(3) make prediction



# Outline

## Background

### Data and Experimental Setup

- Data (Domains + Outcome Variable)
- Experiments
- Features

### Results

### Conclusions



# Problem: Malicious Domains

What are malicious domains used for?

1. Malware binary download site
2. Phishing/scam site
3. Botnet command-and-control (C&C)
4. Data exfiltration site
5. Site for obfuscation to avoid detection



# Previous Machine Learning Approaches (1)

## Features based on the content of webpages

- Download the page and analyze/characterize its content
- Highly accurate, but potentially slow





# Previous Machine Learning Approaches (2)

## Features based on domain names, URLs, hosts:

### 1. Lexical characteristics

- length of domain name, number of digits, etc.
- keywords (i.e. manually curated list of brand names)
- Markov models for character-to-character transitions

### 2. DNS and host-based features

- # of distinct IP addresses and other DNS and WHOIS information

Ref: Garera '07, McGrath '08, He et al '10, Bilge et al '11



## Previous Machine Learning Approaches (3)

Large-scale machine learning on full URLs [Ma et al 2009 (KDD and ICML), 2011] **Most Relevant Work**

- Combine lexical characteristics of full URL (including bag-of-words model on URL path) with host-based features (from DNS and WHOIS queries)
- High accuracy, but real-time implementation requires 3-4 seconds per URL



## Our idea...

Q: Can we extract any more features without sacrificing speed?

A: *Word segmentation* of the domain name



# Thousands of new features: word segmentation

Word segmentation = break a string into one or more substrings

- Recent applications to domain names, Twitter hashtags, etc.
- Methodological research with the goal of recovering a true, known segmentation of a domain name [Wang '11, Srinivasan '12]
- [Norvig '09] a book chapter that included an introduction to word segmentation using language models (and code!)



# Our Approach: A combination of two methods

Word Segmentation on domain names (not full URLs)

+

Machine learning



# Outline

Background

Data and Experimental Setup

- Data (Domains + Outcome Variable)
- Experiments
- Features

Results

Conclusions



# Review: Definition of a domain name

Full URL                      `http://www.more.example.com/path-to-url.html`

Top level domain            `http://www.more.example.com/path-to-url.html`

Second-level domain        `http://www.more.example.com/path-to-url.html`

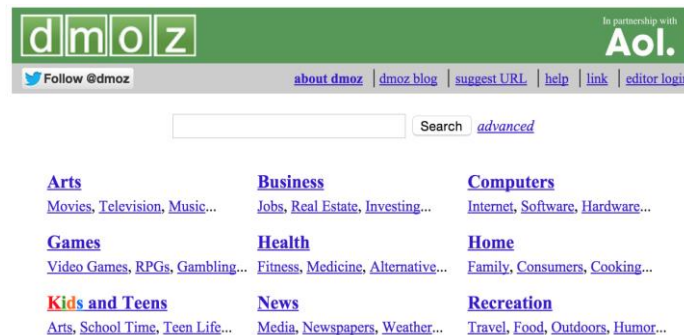
Domain name may only consist of:

- Alphanumeric characters
- Hyphens
- Top-level-domain (TLD)



# Data – Two Sources

1. A sample of domains visited on a cellular network
  - ~ 1.3 million unique domain names from Sept. 2014
1. Domains from DMOZ, the Open Directory Project
  - 30,000 randomly sampled domain names from Nov. 2014





# How to define “malicious”: Web of Trust

What: Web of Trust (WOT), at [www.mywot.com](http://www.mywot.com)



- crowd-sourced website reputation and review service
- The ratings are validated with trusted third party information

Each domain has:

rating	confidence score	category
[0,100]	[0,100]	“trustworthiness” <del>“child safety”</del>

~261,000 out of 1,372,120 (20%) cellular domains had non-empty WOT scores



# Three experiments

## 1. “Balanced Data”

- Why: to compare to Ma’09 and other studies that used DMOZ data.
- Malicious = 1 if rating < 60
- All DMOZ are benign.

	DMOZ	Cellular	Row total
Training	15,000 (benign)	15,000 (malicious)	30,000
Testing	15,000 (benign)	15,000 (malicious)	30,000



# Three experiments

## 2. “Unfiltered Cellular”

- Use all cellular data
- Malicious = 1 if rating < 60
- Baseline rate ~ 14.6% malicious

	Cellular
Training	80%
Testing	20%
Column total	100% (~261,000)



# Three experiments

## 3. “Filtered Cellular”

- Attempt to use only high-quality cellular data
- Remove those with rating in  $[40, 59]$  or confidence  $< 10$
- Malicious = 1 if rating  $< 40$
- Baseline rate  $\sim 24.5\%$  malicious

	Cellular
Training	80%
Testing	20%
Column total	100% ( $\sim 80,000$ )



# Feature sets

## 1. “Basic” (22 features)

- number of characters; number of hyphens; number of digits; number of numbers

(discretized to allow for non-linear relationships)

“4downs-10yards.com”

→ 14 characters, 3 digits, 1 hyphen, and 2 numbers

## 2. “Character indicators” (36 features)

- Indicator(a-z, 0-9)



# Feature sets

## 3. “Character Markov model log-likelihood” (22 features)

- top 1/3 million unigrams from the Google Ngrams corpus to train a 1<sup>st</sup> order Markov model
- transition probability between characters (11 bins)

## 4. “Top level domains (TLDs)” (~400 features)

## 5. “Words” (~94,000 features) \* this is our innovation



# Word segmentation

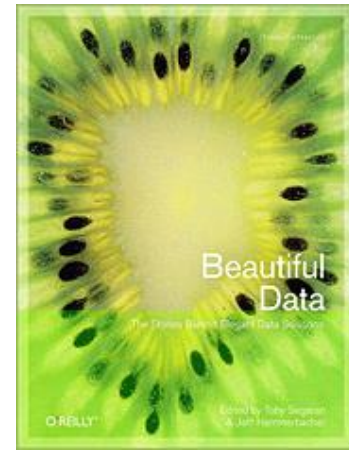
“duckduckgo.com” → {“duck”, “duck”, “go”} (3 tokens)  
→ {“duck”, “go”} (2 words)

Dynamic programming algorithm

[Norvig'09, *Beautiful Data*]

Find the most likely segmentation of a string of characters into a set of one or more tokens based on Google bigrams corpus

2.34 tokens/domain on average



# The most frequent words

	Word	Freq		Word	Freq		Word	Freq
1	a	6772	11	on	1638	21	shop	1168
2	the	6186	12	my	1593	22	world	1157
3	i	3515	13	is	1426	23	music	1143
4	of	3183	14	club	1393	24	city	1078
5	s	2789	15	web	1380	25	it	1056
6	in	2630	16	art	1297	26	center	1047
7	online	2554	17	inc	1234	27	news	1034
8	and	2495	18	co	1187	28	st	1032
9	e	1847	19	for	1181	29	free	1017
10	to	1772	20	de	1171	30	group	1012

30 most frequent words: some stop words, some common  
“web” words

Total Vocabulary Size: 94,050 words





# Model: lasso-regularized logistic regression

Model: logistic regression with lasso penalty (binary classification)

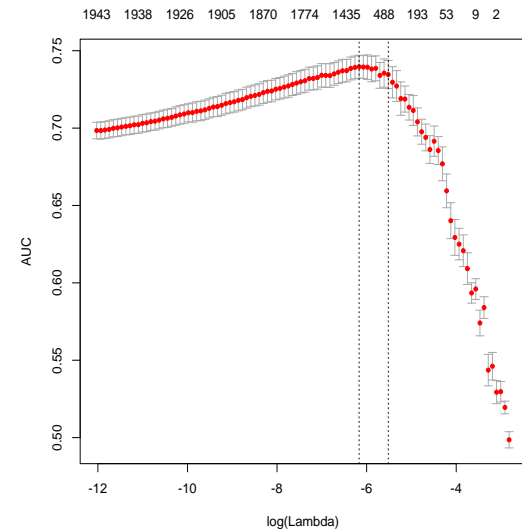
R package glmnet

[Ma'09 found that this was roughly as accurate as SVM and Naïve Bayes]

Training: 10-fold cross-validation

Sparse coefficients (many zeroes)

Features are of the same scale → the coefficients are interpretable



# Outline

Background

Data and Experimental Setup

- Data (Domains + Outcome Variable)
- Experiments
- Features

Results

Conclusions



# Results

		Feature Sets	MCR	AUC	# Features	# $\neq$ 0
Individual set	M1	Basics				
	M2	Characters				
	M3	TLD				
	M4	Log-likelihood				
	M5	Words				
focus of this study!	M6	M1 + M2 + M3 + M4				
	M7	M6 + Words				

MCR results are based on a naive threshold of 0.5



# Summary of model fits to “Balanced Data” (Experiment 1)

	Feature Sets	MCR	AUC	# Features	# $\neq 0$
M1	Basics	0.435	0.595	22	6
M2	Characters	0.478	0.536	36	29
M3	TLD	0.288	0.763	489	101
M4	Log-likelihood	0.492	0.512	22	13
M5	Words	0.373	0.667	24772	4588
M6	M1 + M2 + M3 + M4	0.297	0.771	569	104
M7	M6 + Words	<b>0.277</b>	<b>0.813</b>	25341	2928

- M7 decreases the MCR of M6 by 2% and increases the AUC by about 4%
- M7 has slightly fewer than 3000 active (nonzero) features



# Summary of model fits to “Unfiltered Cellular” data (Experiment 2)

	Features	MCR	AUC	# Features	# $\neq$ 0
M1	Basics	0.146	0.563	22	13
M2	Characters	0.145	0.576	36	27
M3	TLD	0.140	0.657	479	99
M4	Log-likelihood	0.146	0.542	22	18
M5	Words	0.137	0.708	77866	12568
M6	M1 + M2 + M3 + M4	0.137	0.696	559	158
M7	M6 + Words	<b>0.125</b>	<b>0.779</b>	78425	9938

- Improvement in MCR is smaller than the “balanced”
- M7 is substantially better than M6 in AUC (about 8% higher)



# Summary of model fits to “Filtered Cellular” data (Experiment 3)

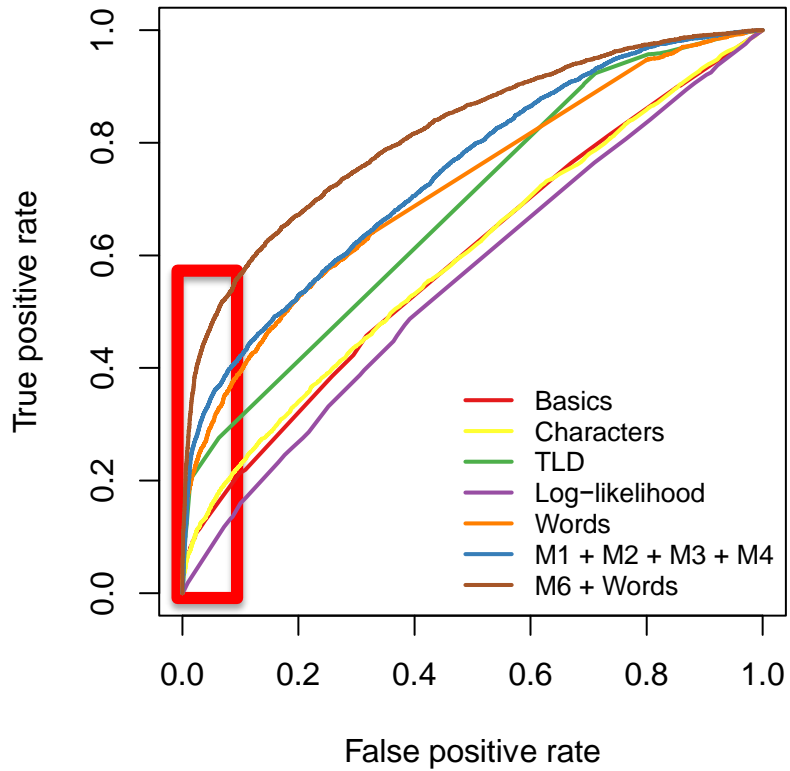
	Features	MCR	AUC	# Features	# $\neq 0$
M1	Basics	0.240	0.593	22	11
M2	Characters	0.242	0.597	36	23
M3	TLD	0.211	0.681	323	33
M4	Log-likelihood	0.250	0.557	22	9
M5	Words	0.212	0.720	38025	8578
M6	M1 + M2 + M3 + M4	0.199	0.744	403	119
M7	M6 + Words	<b>0.167</b>	<b>0.817</b>	38428	5416

- Similar results to those in the “unfiltered cellular”
- MCR rates decrease faster, and the AUC is 4% higher than M7 in the “unfiltered cellular”



# Results – ROC Curves

## Full ROC Curves (Filtered Cellular)



# Words associated with malicious domains

- 1) Brand names: **rayban, oakley, nike, vuitton, hollister, timberland, tiffany, ugg**
- 2) Shopping: **dresses, outlet, sale, dress, offer, jackets, watches, deals**
- 3) Finance: **loan, fee, cash, payday, cheap**
- 4) Sportswear: **jerseys, kicks, cleats, shoes, sneaker**
- 5) Basketball Player Names (associated with shoes): **kobe, jordan, jordans, lebron**
- 6) Medical/Pharmacy: **medic, pills, meds, pill, pharmacy**
- 7) Adult: **webcams, cams, lover, sex, porno**
- 8) URL spoof: **com**





# Words associated with benign domains

- 1) Locations: **europaean, texas, india, europe, vermont, zealand, washington, colorado**
- 2) Hospitality Industry: **inn, ranch, motel, country**
- 3) Common Benign Numbers: **2000, 411, 911, 2020, 365, 123, 360**
- 4) Realty: **realty, builders, homes, properties, estate**
- 5) Small Businesses: **rentals, outfitters, lumber, audio, funeral, flower, taxidermy, inc, golf, law, farm, chamber, farms, rider, photo**
- 6) Geographical Features: **creek, hills, lake, ridge, river, valley, springs, grove, mountain, sky, island**



# Outline

Background

Data and Experimental Setup

- Data (Domains + Outcome Variable)
- Experiments
- Features

Results

Conclusions



# Conclusion

1. Word segmentation added substantial predictive power to a logistic regression model
2. Models are interpretable
3. Highly predictive words may change over time
4. Potential complementary method to more accurate, but expensive and time-consuming approaches



# Future work

1. Use different source(s) for the outcome variable
2. Long term evaluation of the system
3. Online learning with streaming data



# Q & A

