Deconstructing Domain Names to Reveal Latent Topics

Cheryl J. Flynn
AT&T Labs Research
New York, NY, USA
cflynn@research.att.com

Kenneth E. Shirley[1]
Amazon
New York, NY, USA
kennysh@amazon.com

Wei Wang
AT&T Security Research
New York, NY, USA
wei.wang.2@att.com

***Abstract*—Measurement of the lexical properties of domain names enables many types of relatively fast, lightweight web mining analyses. These include unsupervised learning tasks such as automatic categorization and clustering of websites, as well as supervised learning tasks, such as classifying websites as malicious or benign. In this paper we explore whether these tasks can be better accomplished by identifying semantically coherent *groups of words* in a large set of domain names using a combination of word segmentation and topic modeling methods. By segmenting domain names to generate a large set of new domain-level features, we compare three different unsupervised learning methods for identifying topics among domain name keywords: spherical k-means clustering (SKM), Latent Dirichlet Allocation (LDA), and the Biterm Topic Model (BTM). We successfully infer semantically coherent groups of words in two independent data sets, finding that BTM topics are quantitatively the most coherent. Using the BTM, we compare inferred topics across data sets and across time periods, and we also highlight instances of homophony within the topics. Finally, we show that the BTM topics can be used as features to improve the interpretability of a supervised learning model for the detection of malicious domain names. To our knowledge this is the first large-scale empirical analysis of the co-occurrence patterns of words within domain names.**

I. Introduction

By the end of 2015, approximately 314 million domain names had been registered across all top-level domains, including approximately 15 million domain names that had been registered in the fourth quarter of 2015 alone [39]. The identification of patterns and trends in domain names is important for accurately categorizing websites, optimizing search query algorithms [10], [14], and detecting major events [32], [37]. From the perspective of a network service provider, identifying emerging malicious domain names and protecting web users from new malicious attacks is a critical business activity [3]. Until now most web mining efforts related to domain names have focused on basic, low-level characteristics of domain names, such as their lengths, the distribution of their top-level domains (TLDs), and the frequencies of characters and digits within them. However, given the interchangeability of homophones [29] and the use of typos in domain names [20], [26], it is of interest to also analyze higher-level characteristics of domain names, such as which words they contain, and to what extent words co-occur in semantically coherent groups. Latent, interpretable structure in domain names can serve as a useful predictor of emerging trends and threats by providing an informative overview of the semantic properties of domain names across the whole network. An automated method for discovering this structure provides a useful complement to rules-based approaches for security monitoring, where the rules need to be frequently updated and monitored by domain experts.

In this paper we analyze domain names by algorithmically segmenting them into one or more individual tokens, and subsequently fitting an unsupervised learning model to the collection of domain names, where each segmented domain name is treated as an individual document. In our experiments, we compare three different unsupervised learning models for identifying semantically coherent groups of words, or *topics*: spherical k-means clustering (SKM), Latent Dirichlet Allocation (LDA), and the Biterm Topic Model (BTM), since it is unknown which method will work best in this new problem domain. In each method, the user specifies how many topics (or clusters) to infer from the data, and each inferred topic (cluster) is represented by a vector of weights over all the terms in the vocabulary. We use a quantitative measure of topic coherence [25] to measure and compare the semantic coherence of the inferred topics from each method.

This paper makes the following main contributions:

- We outline a novel end-to-end text mining approach combining word segmentation with topic modeling to construct a low-dimensional representation of words within domain names.
- We implement our approach on two collections of domain names: (1) those listed in the Open Directory Project (DMOZ), and (2) a sample of domain names that were visited on the cellular network of a large U.S. telecommunications company in 2013-2014 (hereafter denoted the *cellular data*). We evaluate the inferred topics for each data set quantitatively and qualitatively, and find the topics to be interpretable, robust, and useful for revealing the complex semantic relationships between words within domain names.
- We show that the results from the fitted topic models can be used to detect changes in domain name word groups over time, and to improve the interpretability of supervised learning models for malicious domain detection.

[1]This work was done while the author was at AT&T Labs Research.

The rest of the paper is structured as follows: Section II discusses previous research in this area; Section III describes the data sets we analyze; Section IV describes the models and methods that we used to perform the analysis; Section V describes the basic results; Section VI provides a discussion of the interpretability of the learned topics; Section VII discusses using topics as features in a supervised learning setting, and Section VIII discusses potential future work.

II. Background

In this section we provide an overview of related work on text mining for domain names, applications of word segmentation algorithms for URLs and domain names, and topic modeling for short documents.

Domain Name Keywords: The increasing growth of newly registered domains has led to investigations into the properties of the keywords used in domain data. In 2006, Forbes [13] performed a descriptive study on a sample of 3.5GB `.com` domain names and reported findings on domain length distributions (based on the number of characters), the percentage of common names in the U.S. Census Bureau that have been used to register domain names, and the most common letters and digits to start a domain name. A 2008 study by McGrath and Gupta [24] further found that the length of domain names, the frequencies of certain characters, and a set of curated brand names are strong indications of phishing URLs and domains. More recently, Verisign made publicly available an API tool that makes it possible to visualize trends in domain names by providing time series plots of top-trending or user-specified words in new `.com` and `.net` domain names [38]. These studies all considered words in isolation and did not study word co-occurrences within domain names.

Word segmentation on domain names: Word segmentation, defined as breaking a single string into a sequence of one or more non-empty substrings, has been applied to domain names in the past. Wang et al. [41] performed a series of experiments performing word segmentation on full URLs, and Srinivasan et al. [36] extended this approach by incorporating the lengths of segments into their algorithm. In both cases, the goal of the analysis was to evaluate the word segmentation algorithm by checking performance against a ground-truth set of segmented domain names. In contrast, we segment domain names, not full URLs, with the goal of using the constituent tokens as input into an unsupervised learning model.

Supervised learning with segmented domain names: Segmented text from domain names and URLs has been used to create features in many supervised learning applications in which the goal is to detect malicious websites ([15], [21], [22], [33], [42]). In most cases, certain individual words (brand names, URL spoofs, and others) were found to be associated with malicious websites. In our study, we segment the domain names using the same method as [42], but use the segmented text as features for an *unsupervised* learning analysis, revealing useful relationships between the words. We then explore the value of using the topics themselves as features in a subsequent model (see Section VII).

Topic modeling for short documents: The increased availability of data from the web has led to a growing literature on applying topic models to short documents. One approach to extracting topics from short documents is to modify the LDA model assumptions, (since LDA has difficulties with short documents [18]). Two recently proposed models include the Biterm Topic Model (BTM) [44], which incorporates the co-occurrence patterns of the entire corpus into the generative model, and a Gaussian mixture model trained on vector representations of words [35]. In this paper we include standard LDA and BTM in our experiments as candidate methods for extracting semantically coherent topics from domain names.

III. Data

We analyze domain names from two different sources of data: (1) a sample of domain names that were listed in the Open Directory Project (DMOZ) in April of 2015 [12], and (2) a sample of domain names that were visited on the cellular network of a large U.S. telecommunications company during 2013-2014. In this section we describe each data set in detail.

A. DMOZ Data

We downloaded the "content" component of the DMOZ data (excluding the "Kids and Teens", "Regional", and "World" top-level categories) in April, 2015, which contained a set of URL-category pairs, where each URL in the directory was assigned to one or more categories in the DMOZ category hierarchy. From the resulting initial set of 1,088,060 unique URL-category pairs, we then extracted the effective second-level domain name from each URL. Given a full URL, we first identified the top-level domain (TLD) by utilizing the public suffix list [27]. The effective second-level domain name is defined as the combination of the TLD and the string of characters between the TLD and the second period to the left of the TLD. For example, in the URL http://dendro.cnre.vt.edu/dendrology/syllabus2/factsheet.cfm?ID=638, the TLD is `edu` and the effective second-level domain name is `vt.edu`. (Note that `dendro.cnre` is a *subdomain*, and everything that follows the TLD is part of the *URL path*, both of which we ignore in our analysis). After reducing the URLs to their second-level domains, our final DMOZ sample consisted of 659,775 unique second-level domains, which are hereafter referred to as domain names, or simply "domains".

B. Cellular Data

Our second data set contains daily "fresh domain" data collected by a large U.S. telecommunications company during 2013 and 2014. A "fresh domain" is defined as a domain name visited on a mobile device for the first time by any mobile user in the past 30 days, meaning the data contains a mixture of newly created and infrequently visited domain names. The data does not contain any additional information about domain traffic, nor does it contain any personal identifiers or search terms. By definition every fresh domain name collected within a thirty day span is unique. In order to potentially capture interesting holiday effects, we selected two week-long batches

of fresh domains from the cellular data to analyze: (1) those from the week surrounding Thanksgiving, 2013 (November 26, 2013 to December 02, 2013), and (2) those from the week surrounding Valentine's Day, 2014 (February 12, 2014 to February 18, 2014). We combined these two weeks to create one sample of data. There were 17,997 domain names that appeared during both weeks. We considered these domain names to be part of Thanksgiving week and we removed the duplicate entry in the Valentine's Day week. This resulted in a sample of 816,878 unique domain names. We extract the second-level domain names for this sample as described above.

There is some overlap between the DMOZ and cellular data, but over 97.7% of the 1,476,653 domain names across the combined samples are unique.

IV. Models and Methodology

In this section we describe the methods we use to transform the raw domain name data into a document-term matrix suitable for modeling. We also introduce the three methods we use to extract topics from the data, and the methods we use to interpret the topics after fitting the models.

A. Word Segmentation

We perform word segmentation on the domain names in the following steps. First, for each domain name, we remove the TLD and the period that divides the TLD from the rest of the domain name. Then we split the remaining domain name at the locations of hyphens (discarding the hyphens), and also at the boundaries of any length-1 or greater sequence of consecutive digits, retaining each sequence of digits as a token. Last, for each substring that remains in each domain name, we implement the word segmentation algorithm described by Norvig [30], which uses a probability model for bigrams based on the Google ngrams corpus (a sample of which is available from the author's webpage, [31]). The bigram model considers the probability of each token given the previous token, and the most likely segmentation of a string of alphanumeric characters into one or more tokens is found using dynamic programming.

The result is that we can represent our set of D domain names as a document-term matrix of dimension $D \times W$, where W is the total number of unique tokens, hereafter called *terms*, in the corpus, and each entry in the document-term matrix, x_{dw}, is the number of times that term w occurs within document d. The list of terms is called the *vocabulary*. We also denote the number of tokens in document d as N_d, where the corpus contains a total of N tokens.

B. Stop Word, Rare Words, and Short Documents

In most text mining and topic modeling analyses, the raw vocabulary contains more terms than are desired. We prune the vocabulary in our analyses by removing stop words and rare words. We chose to use a small, or conservative, stop word list, consisting only of the 26 letters "a", "b", "c", ..., "z". Among these only "a" and "i" are valid English words, and upon inspection, we found that most of the instances of individual letters as tokens in our data sets were the result of a domain name that did not consist of concatenated English words, and was therefore poorly segmented. In some applications (such as detecting algorithmically-generated domain names via supervised learning) it may be useful to retain these single-letter terms in the vocabulary, but in this work we focus on co-occurrences of interpretable terms. We do not, however, use a common, larger stop word list, because we choose not to assume that terms such as "the", "of", and "and" (which are commonly discarded as stop words in analyses of other corpora, such as news articles, scientific article abstracts, etc.) are meaningless in the context of domain names. Rather, we retain all occurrences of these terms to learn empirically whether or not, in the context of domain names, they co-occur with other terms in any interesting patterns. Hong and Davison [18] also included stop words and did not perform stemming in their Twitter data study. We also remove from the vocabulary all occurrences of terms that occur fewer than 10 times across the whole corpus.

The BTM is trained on the set of biterms (pair of co-occurring tokens) that occur within documents. Thus, to be consistent across methods, we exclude from our training data any document consisting of a single token. This is consistent with the pre-processing routine in [44].

C. Methods for Identifying Topics

The first method we use to identify groups of co-occurring words within documents is Spherical k-means Clustering, introduced by Dhillon and Modha [11]. This method partitions the documents into K clusters, where K is specified by the user, such that the sum of the cosine distances between each document and the centroid of the cluster to which it is assigned is minimized. The model is fit by initially assigning random cluster IDs to each document, and then alternating between computing optimal cluster centroids given the cluster IDs, and computing optimal cluster IDs given the set of cluster centroids, until the algorithm converges. The result of fitting the model is a cluster ID for each document, and a $K \times W$ matrix whose rows contain the length-W cluster centroids for each of the K clusters. For the rest of the paper we refer to the length-W centroids of the clusters as *topics*, or groups of frequently co-occurring words.

The second method we use to identify topics within domain names is Latent Dirichlet Allocation (LDA) [6]. LDA states that the probability of token j within document d, denoted by the random variable W_{dj}, is:

$$\mathrm{P}(W_{dj} = w) = \sum_{k=1}^{K} \mathrm{P}(W_{dj} = w \mid z_{dj} = k)\mathrm{P}(z_{dj} = k),$$

for documents $d = 1, ..., D$, tokens $j = 1, ..., N_d$, terms $w = 1, ..., W$, and topics $k = 1, ..., K$, where z_{dj} is the latent topic assignment of the j^{th} token in document d. The matrices $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ are commonly referred to as the set of topic-term distributions and the document-topic distributions, respectively, where the rows of $\boldsymbol{\phi}$, denoted $\boldsymbol{\phi}_k$ for $k = 1, ..., K$, contain the

length-W discrete distributions over terms for each of the K topics, and the rows of $\boldsymbol{\theta}$, denoted $\boldsymbol{\theta}_d$ for $d = 1, ..., D$, contain the length-K discrete distributions over topics for each of the D documents.

Each topic-term distribution shares a common prior distribution, where $\boldsymbol{\phi}_k \sim$ Dirichlet($\boldsymbol{\beta}$) for topics $k = 1, ..., K$ and length-W vector $\boldsymbol{\beta}$, and likewise each document-topic distribution shares a common prior distribution, where $\boldsymbol{\theta}_d \sim$ Dirichlet($\boldsymbol{\alpha}$) for documents $d = 1, ..., D$ and length-K vector $\boldsymbol{\alpha}$.

The third model we fit to our data is the Biterm Topic Model (BTM), introduce by Yan, et al [44]. The BTM is similar to LDA except that instead of each token in the data having a latent topic assignment, each biterm, or pair of words within a document, has a latent topic assignment. Furthermore, instead of each document being modeled as a mixture over the K topics, the set of all biterms in the corpus is modeled as a mixture over the K topics, with the Dirichlet prior $\boldsymbol{\alpha}$ applied to the corpus-wide topic mixture, and the Dirichlet prior $\boldsymbol{\beta}$ applied to each topic, just as is the case in LDA. The output of the BTM is the estimated vector of topic proportions across the whole corpus, the set of topic-term distributions ϕ_k for $k = 1, ..., K$, and even though it is not explicitly modeled, the proportion of tokens within each document coming from each topic (analogous to θ in LDA) can be estimated.

Note that for each method, the tokens within each document are exchangeable; in other words, they are all examples of so-called "bag-of-words" models. Spherical k-means is, in some sense, the simplest of the three methods, since it models each document as belonging to a single cluster. On the other hand, LDA and BTM allow documents to be comprised of a mixture of topics, where the latter is expected to be well suited for short documents.

D. Topic Interpretation

To compute a quantitative measure of topic interpretability, we measure the topic coherence for each topic in the models we fit. There are essentially two types of coherence measures for topics: extrinsic (e.g. [28]) and intrinsic (e.g. [25]). We choose *not* to use an extrinsic measure of topic coherence because such methods rely on a comparison of the properties of inferred topics in the corpus of interest to word co-occurrences in a large external corpus, such as newspaper or Wikipedia articles. Given that we don't expect the properties of our corpus of interest (domain names) to match those of any existing external corpus, the coherence of topics within domain names may not be appropriately measured by such a method. In light of this, we compute coherence using the measure proposed by Mimno et al. [25], which is an intrinsic measure that computes the coherence of a topic using the co-document frequency of the top M most probable terms for that topic. Specifically, the topic coherence for topic k is given by

$$\text{coherence}_k = \sum_{j=2}^{M} \sum_{i=1}^{j-1} \log \frac{D(w_j^k, w_i^k) + 1}{D(w_i^k)}$$

where w_j^k is the jth most probable term within topic k, $D(w_j^k, w_i^k)$ is the co-document frequency of terms w_j^k and w_i^k and $D(w_i^k)$ is the number of documents containing term w_i^k. In each of our analyses, we order the topics in decreasing order of coherence.

V. Results

Here we describe the results of the analyses of the DMOZ data and the cellular data. First we report summary statistics of each data set related to the pre-processing of the domain names to represent them as document-term matrices. Then we discuss how we selected the number of topics for our analysis. Last, we report results from the fits of the topic models.

A. Data Preparation

Tables I and III contain summary statistics related to processing both data sets into document-term matrices. Here we describe in detail the processing of the DMOZ data; identical steps were taken for the cellular data.

We segmented the original $D = 659,775$ unique domain names in the DMOZ data using the word segmentation algorithm described by Norvig 2009. There were 1,497,947 total tokens among the 659,775 domain names, resulting in an average of 2.27 tokens per domain name, with a mode of 1 token per domain name and a range of 1-16 tokens per domain name. This average number of tokens per domain is comparable to that found in analyses in [41] (2.66 tokens/domain) and [36] (2.21). The DMOZ sample originally contained 118,955 unique token types (terms).

To see how well the word segmentation algorithm worked on the DMOZ data in general, we compare the most frequent terms from the DMOZ data with the 1/3 million most frequent unigrams from Google's ngram corpus [7]. We found that among the most frequent 10,000 terms from the DMOZ domains, 9,913 of them were among the 1/3 million most frequent Google unigrams, and the 87 terms that were not in the unigram list were all numbers, and 90% of these DMOZ terms were among the most frequent 28,000 Google unigrams, indicating a substantial level of agreement between the two corpora.

We then removed all occurrences of the 26 stop words "a", "b", ..., "z", and all occurrences of the 101,292 terms that occurred fewer than 10 times across the $D = 659,775$ documents. The removal of stop words and rare words resulted in deleting 101,318 terms from the vocabulary and 295,394 tokens from the corpus, leaving a vocabulary with $W = 17,636$ terms, and a corpus with $N = 1,202,553$ tokens and $D = 598,710$ documents. Last, we removed 155,444 documents that consisted of a single token, resulting in a total of $N = 1,047,109$ total tokens and $D = 443,266$ documents in the corpus, where each remaining document contained two or more tokens. The size of the vocabulary was not affected by the removal of single-token documents. Table II contains the 20 most common terms in the processed DMOZ sample.

TABLE I
SUMMARY OF THE RAW SEGMENTED TEXT

	DMOZ	Cellular
Unique Domain Names	659,775	816,878
Total Tokens	1,497,947	2,130,896
Avg Tokens/Domain Name	2.27	2.61
Max Tokens/Domain Name	16	17
Min Tokens/Domain Name	1	1
Mode Tokens/Domain Name	1	2
Unique Token Types (terms)	118,955	128,611

TABLE II
DMOZ TERM FREQUENCIES

Rank	Term	Freq	Rank	Term	Freq
1	the	11661	11	inc	3242
2	of	6526	12	golf	3058
3	and	5705	13	web	3002
4	club	4509	14	on	2849
5	in	4411	15	design	2727
6	st	3953	16	group	2612
7	church	3761	17	4	2505
8	art	3568	18	to	2455
9	online	3438	19	2	2327
10	law	3245	20	world	2323

B. Tuning Parameters

We implement all three topic modeling methods using open-source code. We used the software MALLET [23] to fit the LDA model using Gibbs Sampling, and the R package `skmeans` [19] to implement spherical k-means. For the BTM, we implemented the `C++` code made available by the BTM authors [43].

For LDA and BTM, we used symmetric, relatively non-informative prior distributions for both the topic-term distributions and the document-topic distributions, so that the posterior inference is driven mostly by the data rather than the priors. Specifically, we set $\beta_w = 0.01$ for terms $w = 1, ..., W$, and we set $\alpha_k = 0.1/K$ for topics $k = 1, ..., K$.

Before fitting the models, one must choose how many topics, K, to estimate, which is a notoriously difficult problem. Since LDA is a generative model, one method for choosing K for LDA is to split the corpus into a set of training documents and a set of test documents, and then fit several LDA models to the set of training documents using a different value of K each time, and then choose the value of K which maximizes the log-likelihood of the tokens within the set of test documents [6]. This procedure is objective and relatively straightforward (see [40] for a discussion of different techniques); we use it to guide our choice of K for all three methods.

TABLE III
SUMMARY OF THE FINAL PROCESSED TEXT

	DMOZ	Cellular
N (total tokens)	1,047,109	1,625,750
W (terms in vocabulary)	17,636	22,513
D (documents)	443,266	624,280

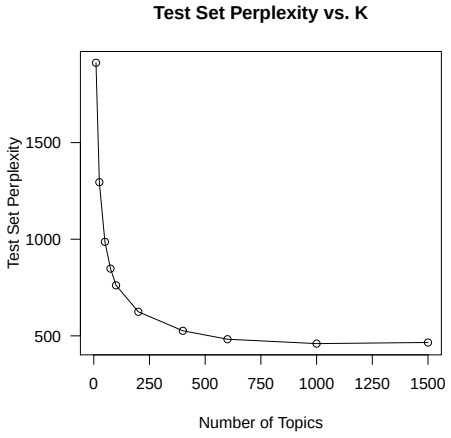

Fig. 1. DMOZ Perplexity vs. Number of Topics

We split our data into training and test sets and we measure the test set perplexity, as computed in [16], to inform our choice of K. For the DMOZ data, using a randomly chosen 80% of the documents as training data and fitting models with $K = \{10, 25, 50, 75, 100, 200, 400, 600, 1000, 1500\}$ topics, we find that the perplexity on the test documents is minimized using a model with approximately $K = 1000$ topics. Figure 1 contains an illustration of the test set perplexity as a function of the number of topics for the DMOZ data.

It has been shown, however, that choosing K this way does not necessarily result in choosing the most interpretable, semantically meaningful topics, as measured by a large-scale user study described in Chang et al. [8]. In fact, Chang et al. found that, in their experiments, as topics became more fine-grained (in models with a large number of topics), topic interpretability decreased. In light of these findings, we decrease the number of topics we use in our analysis and experiment with values of $K = 50, 100, 250,$ and 500 topics. For each of these values of K, the model still achieves a relatively low perplexity, while at the same time being parsimonious and not too burdensome to interpret. We fit models with the same values of K for spherical k-means and BTM to be consistent across the experiments.

We performed a similar training/test split on the cellular data, and found that the test set perplexity was minimized for $K = 1500$ topics. Applying the same argument as above, we ultimately fit the topic models with $K = 50, 100, 250,$ and 500 topics for all three methods.

C. Topic Comparisons

For each model and for each choice of K, we computed the topic model fits based on 1,000 iterations (LDA or BTM) or until convergence (SKM) for 10 random initializations. For LDA and BTM, we compute a point estimate of ϕ_k for each topic k based on the topic assignments from last iteration for each run. In each case we compute the average coherence

across the K topics based on the top $M = 5, 10,$ and 20 terms per topic, following guidance in [25].

Figure 2 illustrates the mean average coherence across topics and runs for each method for different values of M, with bars showing the standard deviations across the 10 runs. For the cellular data, the BTM topic fits produce the most coherent topics on average for all values of K. The results depend more heavily on the choice of K and M for the DMOZ data, but the BTM topic fits are competitive if not the best in general. For the rest of the paper, we will focus on interpreting the fits of the BTM on each data set in more detail.

VI. Topic Interpretation

In this section we compare, discuss and interpret the topics found in our BTM fits. We focus on the BTM with $K = 250$, and present results for the run with the highest average coherence. Given the short average length of the documents, our first objective in this section is to evaluate the interpretability of the topics based on human inspection. Second, we compare the topics across data sets, showing that our topic inferences are robust with respect to the data on which the model was trained. Third, we compare topics inferred from the cellular data across two different time periods. Last, we present examples of homophones and typos that are detected by the topics.

A. Topic Model Fits

The 250 topics varied in size for each of the two data sets. In the DMOZ data, the probability of the largest topic was 4.7%, and the probability of the smallest topic was 0.06%. In the cellular data, the probability of the largest topic was 3.7%, and the probability of the smallest topic was 0.05%.

To interpret the topics we rank terms within topics using the *relevance* metric introduced by Sievert and Shirley [34]. In this scheme, the relevance of term w to topic k is defined as a convex combination of the logarithm of the term's probability within a topic, $\log \phi_{kw}$, and the logarithm of the term's lift, $\log(\frac{\phi_{kw}}{p_w})$, where p_w is the marginal probability of term w across the corpus. Formally:

$$\text{relevance}_{kw} = \lambda \log(\phi_{kw}) + (1 - \lambda) \log\Big(\frac{\phi_{kw}}{p_w}\Big).$$

We used a weight of $\lambda = 0.6$, as recommended in [34], to down-weight terms that are common across the entire corpus. Ranking terms within topics according to a mixture of their probability and exclusivity is similar to approaches described in [4], [5].

Setting $M = 5$, we order the topics in decreasing order of coherence. Table IV summarizes the 10 most coherent topics in each data set. The most coherent topics in both model fits are easily interpretable from examining a short list of the most relevant terms, and user-specified labels are provided. The least coherent topics are more variable, but some themes still emerge, such as Topics 242 {law, and, smith, miller, thomas} and 243 {david, alan, md, jones, michael} for the cellular data, which contain names combined with a profession (law and medical practices for Topics 242 and 243, respectively). Across both data sets, we found several hundred of the topics to be interpretable.

Due to space restrictions we cannot summarize all $K = 250$ topics that we inferred from each of the two data sets in the paper. Therefore we have posted summaries of the topics and all the relevant output from our modeling of the DMOZ data to our website at http://www.kennyshirley.com/domains, and we invite readers to browse the results themselves to assess the usefulness and interpretability of the topics found.

TABLE IV
THE FIVE MOST RELEVANT TERMS FOR THE TEN MOST COHERENT TOPICS IN EACH DATA SET

Human Label {top-5 most relevant terms}
DMOZ
1: Greek {phi, sigma, alpha, gamma, beta}
2: Costa Rica {costa, rica, loss, weight, contra}
3: Hebrew {beth, temple, shalom, bnai, israel}
4: Christian {holy, lutheran, trinity, cross, blessed}
5: Animal Health {animal, hospital, clinic, vet, veterinary}
6: Los Angeles {angeles, los, backers, pike, speak}
7: Dog Breeds {short, shepherds, hair, german, australian}
8: Home Types {log, homes, timber, cabin, cedar}
9: Golf {golf, club, course, disc, tour}
10: Fishing {fishing, fly, fish, reels, carp}
Cellular
1: Cyber Monday {monday, cyber, 2013, deals, ugg}
2: Black Friday {friday, black, beats, 2013, dre}
3: Surveys {surv, lng, ys, you, nu}
4: Weight Loss {loss, weight, diet, fat, lose}
5: Greek {sigma, phi, alpha, omega, delta}
6: Car Brands {chrysler, dodge, jeep, nissan, ram}
7: Spa {spa, salon, hair, beauty, and}
8: Adult {porn, sex, tube, xxx, gay}
9: Homophones {vow, chr, hear, here, reel}
10: Shoes {shoes, nike, cheap, jordan, lebron}

B. Sample Comparisons

In this section we compare the BTM topics from the two data sets to assess the robustness of our results, where a high degree of similarity between topics across the two data sets would indicate that the topics represent signal in domain name patterns, rather than noise. To measure the similarity of a pair of topics, we first compute the Hellinger distance between each pair of topics across the two data sets [16]. The vocabularies of the two data sets are not the same, so for the Hellinger distance computation, we compute the element-wise differences in square roots across the *union* of the vocabularies from each data set.

Figure 3 plots the matrix of pairwise distances between the $K = 250$ topics inferred from the each of the cellular and DMOZ data sets, where topics were matched by minimizing the sum of pairwise distances (i.e. by solving the linear assignment problem [16]). From the diagonal of this plot, it is evident that there is overlap between the topics. Table V lists the five most relevant terms for the the ten most similar pairs of topics. These topics are interpretable and closely matched across data sets, suggesting that the topics learned from the BTM models are robust with respect to the data on which the models are trained.

Fig. 2. Mean coherence values across topics for different data sets and choices of K and M.

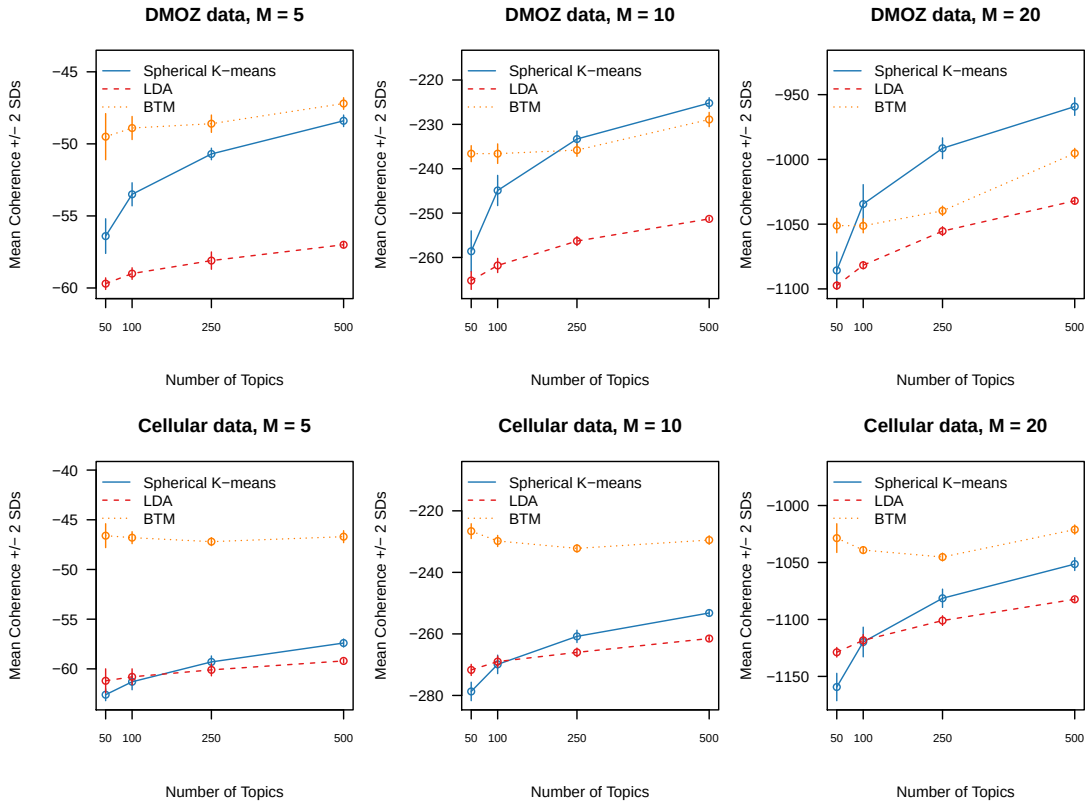

TABLE V
THE FIVE MOST RELEVANT TERMS FOR THE TEN MOST SIMILAR TOPICS

Cellular	DMOZ
photography, and, dr, art, scott	and, photography, law, david, john
st, saint, johns, louis, paul	st, saint, parish, marys, johns
church, baptist, christ, christian, bible	church, baptist, christ, cowboy, first
lawyer, injury, attorney, criminal, law	lawyer, injury, attorney, criminal, law
new, york, england, orleans, zealand	new, york, england, zealand, orleans
law, firm, office, offices, group	law, firm, office, offices, elder
of, the, fame, house, taste	of, christ, the, lady, our
sigma, phi, alpha, omega, delta	phi, sigma, alpha, gamma, beta
bc, 2, law, 3, pm	as, bc, is, 2, pc
of, city, center, west, valley	of, club, symphony, city, aikido

C. *Topics Across Time Periods*

In this section we use the BTM topic fits from the cellular data to explore changes in groups of domain name keywords over time. Since a single topic model was fit to the two weeks of cellular data, where the weeks are separated by about two months, we can compare the topic frequencies across weeks. To compute the distribution of topics for each week, we first need to infer the topic distribution for each domain name. Although the BTM is not a generative model, we follow the guidance of [44] and assume that the document topic proportions are equal to the expected biterm topic proportions based on the biterms in each domain name. We then assign each domain name to it's dominant topic, and compute the topic distribution for each week based on the distribution of topic assignments.

The Chi-squared test of independence is highly significant, suggesting that the distribution of topics varies between the two weeks. To dig deeper into these differences, we use a two-sample proportion test to detect significant changes in topic proportions. We use the test statistics to rank topics rather than report p-values to avoid issues related to multiple comparisons.

The top six most significant topics and their top 10 relevant keywords are listed in Table VI. Except for the second and sixth topics, the most significant differences highlight changes in groups of domain name keywords due to holiday events. In particular, comparing Thanksgiving week to Valentine's Day week, there is a much larger presence of keywords related to Black Friday, Cyber Monday, Discount Shopping, and

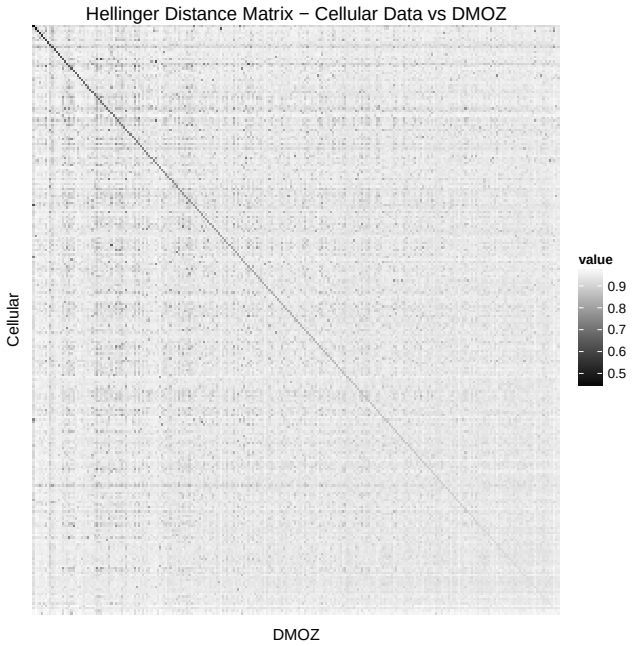

Fig. 3. Hellinger distances between each pair of topics in the DMOZ data and cellular data.

TABLE VI
LARGE TOPIC DIFFERENCES ACROSS WEEKS

Topic	Z-Score	Keywords
2	11.61	friday, black, beats, 2013, dre
9	-9.85	vow, chr, hear, here, reel
42	7.94	tree, christmas, farm, trees, family
19	7.09	outlet, kors, boots, sale, cheap
1	6.83	monday, cyber, 2013, deals, ugg
3	6.16	surv, lng, ys, you, nu

Christmas Trees. From the second and sixth most significant topics, it is also evident that we may be able to detect changes in keyword groups due to what appears to be spam behavior. A previous study [17] found that new spammer domain names were typically registered in bulk and likely to have various combinations of common words that relate to the spam campaign itself. Both topics contain evidence of sound- and typo-squatting, where Topic 9 is significantly more prevalent in Valentine's Day week, and Topic 3 is more prevalent in Thanksgiving Day week. These are presumably the weeks in which the spam campaigns were either launched or gained mainstream exposure. We discuss these topics in more detail in Section VI-D.

D. Homophones and Typos

The fits of the topic models can also be used to reveal examples of homophones and typos in domain names.

Typosquatting and soundsquatting are known behaviors in domain name registrations that take advantage of the human tendency to confuse homophonous terms (terms with the same pronunciation, but different meanings) and to make typos. These patterns are often associated with malicious domain names, so it is important to detect emerging trends indicative of these behaviors. As mentioned in the previous section, Topics 3 and 9 from the cellular data model pick up on several potential incidences of typosquatting and soundsquatting. To provide an expanded view of these topics, the top 15 most relevant terms are

Topic 3: surv, lng, ys, you, nu, spark, slick, fer, starry, new, kool, gnu, fancy, for, 4

Topic 9: vow, chr, hear, here, reel, stay, er, heer, moz, prmo, deel, erp, dealz, kool, pr, deals, of, rmos, rz, mos

From the above, we identified a series of domain names associated with Topics 3 and 9 that used the terms in the sets ("4", "for", "fer"), ("new", "nu"), and ("hear", "here", "heer") interchangeably. The topics also picked up on the misspellings of the terms surveys ("surv" + "ys"), sparkling ("spark" + "lng"), and promos ("pr" + "mos"). Upon inspection of the actual domain names in the data, we confirmed that these were typos that were not recognized by the word segmentation algorithm, because they do not constitute words in the English language. However, by learning the term co-occurrence patterns, we are able to identify them from the topic models. In the next section we further explore the usefulness of topic models for the identification of malicious domain names.

VII. SUPERVISED LEARNING

In this section, we investigate whether the topics learned from our unsupervised learning models can be used as features in a supervised learning task to improve model performance. Our supervised learning task is to detect malicious domain names, where maliciousness is encoded as a binary variable, and indicates that a website is untrustworthy (usually suspected of hosting malware or a phishing attack). As noted by Ma et al. [22], lightweight models for detecting suspicious websites are a crucial safety tool for network providers because blacklists and other traditional website reputation scoring methods can be slow to adapt to new threats. Models based solely on lexical characteristics of a domain name, such as those introduced in this paper, can serve as useful complements (rather than substitutes) to more expensive methods that use, for example, website content or domain registrant information as features. In security applications, [1] point out that the quality of an intrusion detection system depends on both the predictive accuracy of its underlying model as well as the time required of an analyst to take action based on the model. For this reason, we evaluate our model performance with respect to both its predictive accuracy as well as its interpretability, the latter of which provides necessary context to analysts.

For this experiment we obtained a subset of data introduced in [42], where a domain name was labeled as malicious based on a crowd-sourced website reputation tool known as the Web of Trust [2]. We narrowed the sample from [42] by requiring the domain names to contain at least two tokens after being segmented, resulting in a labeled sample of 167,179

domain names. These domain names are from the same source as the cellular data (i.e. "fresh" domain names visited by customers of a large U.S. telecommunications carrier), but from a different time period, and having no overlap with the cellular data sample introduced in Section III.

We split the labeled sample into a training and a test set, with 80% randomly chosen for training, and where the baseline rate of maliciousness is 15.7%. We then fit a logistic regression model with a lasso penalty to various combinations of feature sets, resulting in a total of eleven models, numbered M1, M2, ..., M11. The first seven models (M1 - M7) are the same baseline models that were previously fit in [42]. These models include combinations of the following five feature sets: (1) basic characteristics of domain names such as length, the presence of hyphens and digits, etc., (2) indicators for individual characters, (3) the TLD, (4) the likelihood of the sequence of characters based on a character-level Markov model trained on external data, and (5) the individual words resulting from segmentation. We then construct three new models M8 - M10, which are the best-performing models whose features include only the topics learned from SKM, LDA and BTM, respectively. The best performer for each of these three types of models was chosen by selecting the model with the highest AUC among those fit with $K = 50, 100, 250, 500$ topics. Finally, M11 uses the feature set chosen from the best model among M8 - M10, along with all feature sets in M7. We are especially interested to see whether M11 results in a higher AUC than M7, and also if the fit of M11 is more interpretable than the fit of M7, indicating that using topics as features is advantageous.

We computed the misclassification rate (MCR) based on a naive threshold of 0.5 and the AUC on the test set for each model. We also measured the number of nonzero features estimated by the lasso-penalized logistic regression in each model. Table VII summarizes the fits of the eleven models. Among models M1-M7, model M7 performs the best, with an AUC of 0.797; it was also the best model in the experiments in [42] (recall that although the models are the same, the data here is a subset of the previously analyzed data, thus, the results could have been different). Among models M8 - M10, the model using 50 BTM topics (M10) provided the best performance, with an AUC 5% larger than the worst of these three models, the 500-topic SKM (M8), and using only 23 features compared to 369 for M8. Overall the AUC values were lower for M8 - M10 compared to M5, whose feature sets contained individual words, showing that by using topics instead of words (i.e. reducing the dimensionality of the feature set), one loses some predictive accuracy in exchange for a smaller, more interpretable model, with several thousand fewer nonzero features.

In the presence of all the basic feature sets, including individual words, however, adding topics as features results in *a slight improvement* in predictive accuracy: the AUC for M11 is 0.5% larger than for M7 (80.2% vs. 79.7%). This difference is not statistically significant, according to [9], but it is encouraging, and we plan to conduct further research to assess the predictive improvement provided by topics in similar settings.

TABLE VII
SUMMARY OF 11 MODEL FITS TO UNFILTERED CELLULAR DATA, WHERE $|F|$ DENOTES THE NUMBER OF FEATURES, AND "#$\neq 0$" DENOTES THE NUMBER OF NONZERO FEATURES

	Feature sets	MCR	AUC	$\lvert F\rvert$	#$\neq 0$
1	Basics	0.156	0.566	22	14
2	Characters	0.154	0.584	36	23
3	TLD	0.152	0.638	338	23
4	Log-likelihood	0.156	0.547	22	9
5	Words	0.132	0.761	30401	6538
6	M1+M2+M3+M4	0.147	0.686	418	110
7	M6 + Words	0.125	0.797	30819	5551
8	SKM ($K = 500$)	0.154	0.665	500	369
9	LDA ($K = 100$)	0.150	0.673	100	77
10	BTM ($K = 50$)	0.148	0.717	50	23
11	BTM ($K = 50$) + M7	0.125	0.802	30869	4262

The biggest advantage to including topics as features in our malicious domain detection model is increased interpretability. Of the 50 BTM topics included as features in M11, 36 were selected by the lasso regularization as nonzero, and more importantly, the number of individual words selected as nonzero features was reduced from 5,443 in model M7 to 4,138 in model M11, requiring the interpretation of over 1,300 fewer features. As can be seen in Table VIII, the topics that are most and least associated with malicious domains are highly interpretable, and are, of course, learned automatically, rather than manually constructed as they were in [42]. The top four most malicious topics are related to discount shoes, adult content, financial scams, and drug/pharmaceutical offerings, respectively, which are all very well-known phishing strategies [22], [24]. Note that these topics were learned automatically in an upsupervised fashion, and are easily interpreted by reviewing just the top-6 words within each topic. The most benign topics are related to municipalities, geographical features (often associated with golf courses, real estate offerings, and campgrounds), personal photography websites, and churches.

TABLE VIII
THE SIX MOST RELEVANT TERMS FOR THE FIVE MOST MALICIOUS AND FIVE MOST BENIGN TOPICS

Topic	Keywords
	Most Malicious
26	sale, cheap, shoes, nike, outlet, 2014
27	sex, porn, tube, teen, girls, gay
33	payday, loan, credit, loans, cash, hour
35	top, online, best, viagra, 24, buy
12	my, free, 2, the, 4, web
	Most Benign
7	county, of, city, chamber, society, hospital
50	creek, inn, lake, mountain, farm, river
25	club, north, coast, west, golf, lakes
37	and, photography, david, dr, photo, by
21	st, saint, parish, mary, marys, johns

VIII. FUTURE WORK

In this paper, we've demonstrated the effectiveness of topic models for domain names on two different data sets. In both

cases we found that topic models can reveal meaningful and interpretable topics for domain names despite the short length of each document, and that these topics capture meaningful patterns in domain names that would be difficult to identify from low-level characteristics such as keyword frequencies in isolation. In a comparative study, we found that the Biterm Topic Model provided more semantically coherent topics than spherical k-means clustering or LDA. We also found that including topics as features in a supervised learning application increased the interpretability of the model with no decrease in predictive accuracy.

Future work includes investigating the effectiveness of dynamic topic models for event detection on new domain name registration data (possibly over a long time span), and further studying the use of topics as features in supervised learning problems for malicious site detection, and other supervised learning tasks.

REFERENCES

[1] *Network Security Through Data Analysis: Building Situational Awareness*. OReilly Media, Inc., 2014.

[2] Web of Trust Safe Browsing Tool, November 2014.

[3] G. Aaron, R. Rasmussen, and A. Routt. Global phishing survey: trends and domain name use in 1H2014. Technical report, Anti-Phishing Working Group, September 2014.

[4] J. M. Bischof and E. M. Airoldi. Summarizing topical content with word frequency and exclusivity. In *Proceedings of the 29th International Conference on Machine Learning (ICML)*, 2012.

[5] D. M. Blei and J. D. Lafferty. Topic models. In *Text Mining: Theory and Applications*. Taylor and Francis, London, UK, 2009.

[6] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[7] T. Brants and A. Franz. Web 1t 5-gram version 1. https://catalog.ldc.upenn.edu/LDC2006T13, September 2006.

[8] J. Chang, S. Gerrish, C. Wang, J. L. Boyd-Graber, and D. M. Blei. Reading tea leaves: How humans interpret topic models. In *Advances in Neural Information Processing Systems 22*, pages 288–296, 2009.

[9] C. Cortes and M. Mohri. Confidence intervals for the area under the ROC curve. In L. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 305–312. MIT Press, 2005.

[10] M. Cutts. https://twitter.com/mattcutts/status/251784203597910016, September 2012. @mattcutts: "Minor weather report: small upcoming Google algo change will reduce low-quality 'exact-match' domains in search results.".

[11] I. S. Dhillon and D. S. Modha. Concept decompositions for large sparse text data using clustering. *Machine Learning*, 42(1):143–175, 2001.

[12] DMOZ. http://www.dmoz.org, April 2015.

[13] D. Forbes. Interesting facts about domain names. https://dennisforbes.ca/index.php/2006/03/29/interesting-facts-about-domain-names, March 2006.

[14] D. Forrester. Domain name importance in ranking. https://blogs.bing.com/webmaster/2014/01/15/domain-name-importance-in-ranking, January 2014.

[15] S. Garera, N. Provos, M. Chew, and A. D. Rubin. A framework for detection and measurement of phishing attacks. In *Proceedings of the 2007 ACM workshop on Recurring malcode*, pages 1–8, 2007.

[16] B. Grun and K. Hornik. topicmodels: An R package for fitting topic models. *Journal of Statistical Software*, 40(13):1–30, 2011.

[17] S. Hao, M. Thomas, V. Paxson, N. Feamster, C. Kreibich, C. Grier, and S. Hollenbeck. Understanding the domain registration behavior of spammers. *Proc. ACM SIGCOMM IMC*, 2013.

[18] L. Hong and B. D. Davison. Empirical study of topic modeling in twitter. In *Proceedings of the First Workshop on Social Media Analytics*, 2010.

[19] K. Hornik, I. Feinerer, M. Kober, and C. Buchta. Spherical k-means clustering. *Journal of Statistical Software*, 50(10):1–22, 2012.

[20] D. Kesmodel. *The domain game: how people get rich from Internet domain names*. Xlibris Corporation, 2008.

[21] A. Le, A. Markopoulou, and M. Faloutsos. Phishdef: URL names say it all. In *INFOCOM*, 2011.

[22] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker. Beyond blacklists: Learning to detect malicious web sites from suspicious urls. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1245–1254, 2009.

[23] A. K. McCallum. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu, 2002.

[24] D. K. McGrath and M. Gupta. Behind phishing: An examination of phisher modi operandi. In *Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats*, 2008.

[25] D. Mimno, H. M. Wallach, E. Talley, M. Leenders, and A. McCallum. Optimizing semantic coherence in topic models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 262–272, 2011.

[26] Y. min Wang, D. Beck, J. Wang, C. Verbowski, and B. Daniels. Strider typo-patrol: Discovery and analysis of systematic typo-squatting. microsoft research. Technical report, the 2nd Usenix Workshop on Steps to Reducing Unwanted Traffic on the Internet (SRUTI 06)), http://research.microsoft.com/URLTracer, 2006.

[27] Mozilla. Mozilla public suffix list. https://publicsuffix.org/list/effective_tld_names.dat, November 2014.

[28] D. Newman, J. H. Lau, K. Grieser, and T. Baldwin. Automatic evaluation of topic coherence. In *Proceedings of the 2010 Annual Conference of the North American Chapter of the ACL*, pages 100–108, 2010.

[29] N. Nikiforakis, M. Balduzzi, L. Desmet, F. Piessens, and W. Joosen. Soundsquatting: Uncovering the use of homophones in domain squatting. In S. Chow, J. Camenisch, L. Hui, and S. Yiu, editors, *Information Security*, volume 8783 of *Lecture Notes in Computer Science*, pages 291–308. Springer International Publishing, 2014.

[30] P. Norvig. Natural language corpus data. In T. Segaran and J. Hammerbacher, editors, *Beautiful Data*, chapter 14, pages 219–242. O'Reilly Media, 2009.

[31] P. Norvig. Homepage for Natural Language Corpus Data. http://norvig.com/ngrams, October 2014.

[32] J. Pot. 5 events that have triggered a domain name registration gold rush. http://www.makeuseof.com/tag/5-events-that-have-triggered-a-domain-name-registration-gold-rush, May 2013.

[33] J. Raghuram, D. J. Miller, and G. Kesidis. Unsupervised, low latency anomaly detection of algorithmically generated domain names by generative probabilistic modeling. *Advanced Research*, 5(4):423–433, 2014.

[34] C. Sievert and K. E. Shirley. LDAvis: A method for visualizing and interpreting topics. In *ACL 2014 Workshop on Interactive Language Learning, Visualization, and Interfaces*, 2014.

[35] V. K. R. Sridhar. Unsupervised topic modeling for short texts using distributed representations of words. In *Proceedings of NAACL-HLT*, pages 192–200, 2015.

[36] S. Srinivasan, S. Bhattacharya, and R. Chakraborty. Segmenting web-domains and hashtags using length specific models. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1113–1122, 2012.

[37] Verisign. Domain registrations: Is bitcoin going mainstream? blogs.verisigninc.com/blog/entry/domain_registrations_is_bitcoin_going?cmp=blog, December 2014.

[38] Verisign. Top trending words on domain names. https://domainview.verisignlabs.com, 2015.

[39] Verisign. The domain name industry brief. https://www.verisign.com/assets/domain-name-report-april2016.pdf, April 2016.

[40] H. M. Wallach, I. Murray, R. Salakhutdinov, and D. Mimno. Evaluation methods for topic models. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*, pages 1105–1112, 2009.

[41] K. Wang, C. Thrasher, and B.-J. P. Hsu. Web scale nlp: a case study on URL word breaking. In *Proceedings of the 20th international conference on World Wide Web (WWW)*, pages 357–366, 2011.

[42] W. Wang and K. E. Shirley. Breaking bad: Detecting malicious domains using word segmentation. In *IEEE Web 2.0 Security and Privacy Workshop (W2SP)*, 2015.

[43] xiaohuiyan. BTM. https://github.com/xiaohuiyan/BTM, 2016.

[44] X. Yan, J. Guo, Y. Lan, and X. Cheng. A biterm topic model for short texts. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1445–1456. International World Wide Web Conferences Steering Committee, 2013.