

# Text-Based Ads-Portal Domains: Identification and Measurements

MISHARI ALMISHARI

University of California, Irvine

and

XIAOWEI YANG

Duke University

---

A text-based ads-portal domain refers to a web domain that only shows advertisements, served by a third-party advertisement syndication service, in the form of ads listing. We develop a machine-learning-based classifier to identify text-based ads-portal domains, which has 96% accuracy. We use this classifier to measure the prevalence of ads-portal domains in the Internet. Surprisingly, 28.3/25% of the (two-level) *.com/.net* web domains are ads-portal domains. Also, 41/39.8% of *.com/.net* ads-portal domains are typos of well-known domains, also known as typo-squatting domains. In addition, we use the classifier along with DNS trace files to estimate how often Internet users visit ads-portal domains. It turns out that  $\sim 5\%$  of the two-level *.com*, *.net*, *.org*, *.biz* and *.info* web domains in the traces are ads-portal domains and  $\sim 50\%$  of these accessed ads-portal domains are typos. These numbers show that ads-portal domains and typo-squatting ads-portal domains are prevalent in the Internet and successful in attracting many visits. Our classifier represents a step towards better categorizing the web documents. It can also be helpful to search engines ranking algorithms, helpful in identifying web spams that redirects to ads-portal domains, and used to discourage access to typo-squatting ads-portal domains.

---

## 1. INTRODUCTION

A text-based ads-portal domain refers to a web domain that only shows advertisements in the form of ads listing and does not have real content. Generally, third-party advertisement syndication servers (domain parking services) provide the ads content to ads-portal domains. Recently, the existence of such domains in the Internet is more apparent. It is not uncommon for Internet users to come across such domains either accidentally or on purpose. Figure 1 shows some examples of ads-portal domains.

Ads-portal domains are useful in showing related (to their domain names) ads content to users performing what so called Direct Search. Direct Search or Type-in Traffic refers to the practice of searching for a specific topic in the Internet by bypassing regular search engines and directly typing a topic-related domain name in the address bar hoping it resolves to an ads-portal page related to the sought topic [Wikipedia 2008]. For example, a user interested in nail-related topic may bypass the search engines and directly type *www.nail.com* in the browser address bar, and then the user land on the page shown in Figure 1(a). This domain is an ads-portal domain showing ads links that are nail-related. In this context, ads-portal domains could be helpful to users as they save them the hassle of search engines and immediately show them what they really need.

---

Authors' addresses: Mishari Almishari (contact author), UCI Computer Science Department, Irvine, CA 92697; email: malmisha@uci.edu; Xiaowei Yang, Duke University Computer Science Department, Durham, NC 27708; email: xwy@cs.duke.com.

Despite their usefulness, ads-portal domains are misused in at least two ways. The first way of misusing ads-portal domains is typo-squatting. Typo-squatting refers to the practice of registering domain names that are typographical errors of other well-known domains [Wang et al. 2006]. The ease of setting up ads-portal domains encourages many Internet users to register typo domains and set these typo domains to resolve to ads-portal pages. Typo-squatters abuse the ads syndication business to monetize the incoming traffic to their domains, which is meant to be to other target domains. The other way of misusing ads-portal domains is in the web redirection spam [Wang et al. 2007]. Web spam pages are web pages that mislead search engines, using questionable search engine optimization (SEO) techniques, to promote their URLs in the lists of search results. Web redirection spam is one type of web spam where the web spam page redirects the browser to another spammer-controlled page, which is mostly an ads-portal page. Wang et al. [2007] show that many web spam URLs redirect traffic to ads-portal pages controlled by web spammers<sup>1</sup>. This way, the web spammer monetize the incoming traffic to his/her spam URL.

The first goal of our research is to develop a methodology that accurately distinguishes ads-portal domains from other web domains. Identifying ads-portal domains is a challenging problem as many ads-portal domains adopt different patterns in showing their ads. What makes it even more difficult is that some non-ads-portal domains, such as web directories and web guides, have their look-and-feel similar to ads-portal domains. Figures 2 shows snapshots of such confusing non-ads-portal domains. To address this challenge, we first explore a number of content-based properties of ads-portal domains. Then, we verify the effectiveness of these properties, in terms of distinguishing ads-portal domains, by analyzing their distributions. Subsequently, we employ machine learning techniques and our effective properties to produce a binary classifier that has 94% accuracy in identifying ads-portal domains. Finally, we enhance the performance of the classifier by adding other keyword-based features to the feature vector. The accuracy of the resulting classifier increases to 96%<sup>2</sup>.

It is known that ads-portal domains exist in the Internet and that typo-squatters are using them to monetize the traffic to their typo-domains. How prevalent ads-portal domains are in the Internet? Do they represent a trivial or a major ratio of Internet web domains? What percentage of those domains are typo domains? How often ads-portal domains are accessed by Internet users? Do Internet users access ads-portal domains because of the typos they make? The second goal of our research is to perform measurements that are intended to answer these questions. In our measurements, we use our identification methodology to identify ads-portal domains and we use third-party services to identify typos. Specifically, we use the well-known typo correction services that are provided by Google [Google 2006] and Yahoo [Yahoo 2008]. If either Google's corrector or Yahoo's corrector corrects a given domain, we consider that domain as a typo.

To measure the prevalence of (typo) ads-portal domains in the Internet, we sampled *.com* and *.net* top-level domain (TLD) zone files<sup>3</sup>, downloaded the sampled domains,

<sup>1</sup>Specifically, the authors in [Wang et al. 2007] queried 323 popular spam keywords at three popular search engines and analyzed the top fifty returned results. They were able to identify 4,803 redirection spam URLs.

<sup>2</sup>Currently, the classifier is designed to work for domains that are in English language. However, the accuracy results reported in Section 6.7 show that the classifier has a similar accuracy values in data sets that probably include non-English domains

<sup>3</sup>The zone files were collected from VeriSign [www.verisign.com](http://www.verisign.com).

and then we fed these domains to our classifier. Surprisingly, we found that 28.3/25% of the (two-level) *\*.com/\*.net* web domains were ads-portal domains and 41/39.8% of these ads-portal domains were typos<sup>4</sup>. To measure how often users access these (typo) ads-portal domains, we collected two-month DNS trace files from UCI's DNS name resolvers. Then, we randomly sampled the two-level *\*.com*, *\*.net*, *\*.org*, *\*.biz* and *\*.info* domains, downloaded them, and fed them to our classifier. We found that  $\sim 5\%$  of these accessed web domains were ads-portal domains and  $\sim 50\%$  of these accessed ads-portal domains were typos. These measurements show the considerable prevalence of ads-portal domains in the Internet and their success in attracting many visits. Also, The measurements show that typo-squatting domains represent a huge ratio of ads-portal domains and many ads-portal domains are accessed because they are typos.

The importance of this work is multifold. This accurate identification methodology represents a step towards better mining and categorizing the web domains and documents as 28.3/25% of the (two-level) *\*.com/\*.net* web domains can be accurately identified by our classifier. Also, our classifier can be helpful to search engines ranking algorithms because knowing a domain is an ads-portal may help the search engines to lower-rank this domain in the search results or at least tag it (or even avoid indexing/storing it), as many users using the search engines may not be interested in seeing ads-portal domains. In addition, our classifier may also be helpful to search engines in avoiding indexing web spam pages that redirect to ads-portal domains. Moreover, our methodology can be used, along with some typo function, by Internet browsers to avoid access to typo-squatting domains and making such a practice less profitable. Note that relying on a typo-function only to identify typo-squatting domains may lead to huge ratio of false positives<sup>5</sup>. That means that using our identification methodology is helpful in better identifying typo-squatting domains and not confusing them with legitimate domains that happen to be typos to some well-known domains.

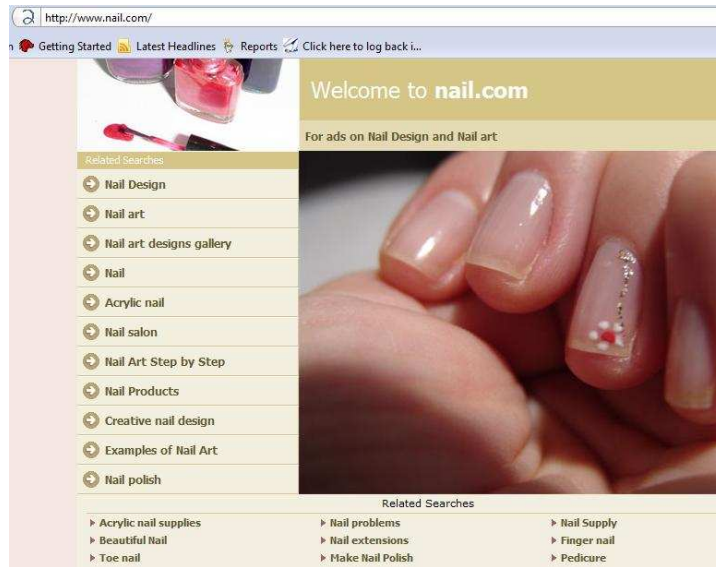
The rest of this paper is organized as follows: In Section 2, we provide some background information about parking services. In Section 3, we describe the data sets we use for training the classifier and performing the measurements. In Section 4, we describe a set of content-based properties of ads-portal domains. In Section 5, we show how we employ machine learning techniques to identify ads-portal domains. In Section 6, we present several experiments that show the prevalence of ads-portal domains in the Internet and how often they are accessed by Internet users. In Section 7, we discuss our future directions. In Section 8, we discuss the related work, and finally, in Section 9, we conclude.

## 2. DOMAIN PARKING OVERVIEW

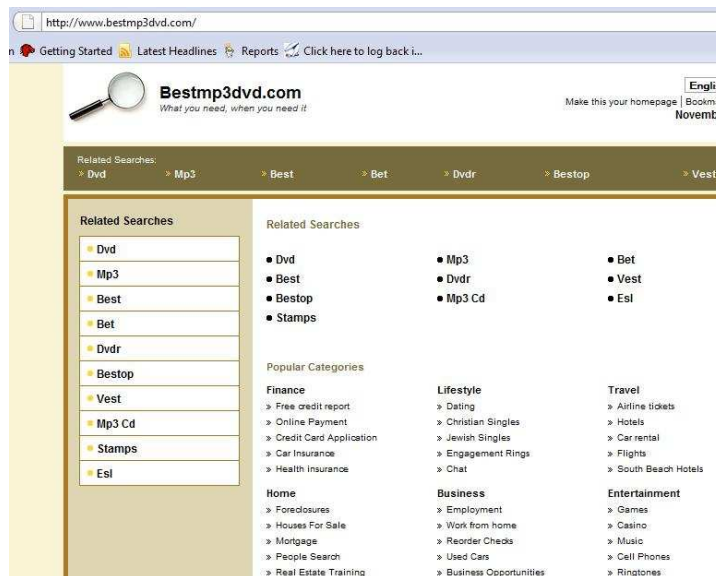
Parked domain is an unused domain that maps to an ads-portal page, which fetches its ads from a parking service (an advertisement syndication service). Most of the time, parked domains are hosted at parking services' servers, but parked domains could be hosted at other servers. The ads-content served by a domain parking service forms the main content

<sup>4</sup>The above TLD zone files have been collected in July, 2008. We have also downloaded the TLD zone files of March of 2009 and repeated the same experiment. The measurement results are comparable to the ones in July, 2008. Specifically, we found that in March of 2009, 25.9/24.2% of the (two-level) *\*.com/\*.net* web domains were ads-portal domains and 41.4/39.3% of these ads-portal domains were typos.

<sup>5</sup>In fact, we found that 68% of the domains in our DNS traces that were one-error-away from one of the top 100 US domains, taken from *alexa.com*, were legitimate non-ads-portal domains

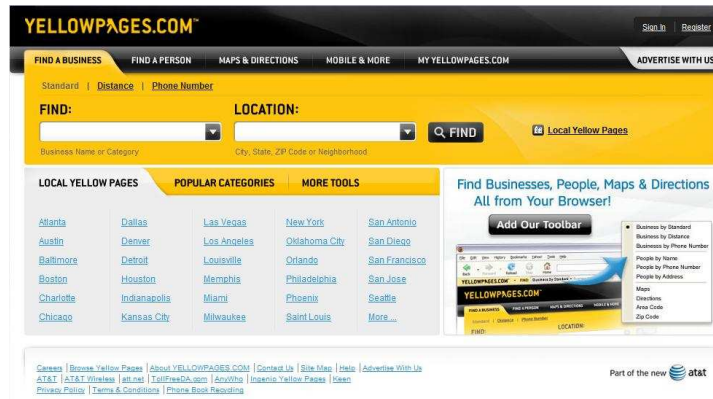


(a)



(b)

Fig. 1. Ads-Portal Domains



(a)



(b)

Fig. 2. Confusing Non-Ads-Portal Domains

of the served ads-portal page. Thus, the served ads content, from a parking service, collectively looks like a complete web page. Figure 1 shows screen shots of parked domains from two different parking services<sup>6</sup>. In contrast, ads-content served by other Internet advertisement syndication services, such as Google AdSense [Google 2008] forms only a section of a web page that has other real content.

Parked domains show ads-portal pages through several methods. A parked domain could have a third-party URL that fetches the ads content from a parking service; i.e., a parked domain could be frame-based and have a third-party URL that fetches the ads content from a parking service. Also, a parked domain could forward the user request to a parking service which shows the ads content. The forwarding could be at the DNS level, HTTP level, or HTML level. Forwarding at the DNS level [Mockapetris 1987] means that a parked domain name resolves to the IP address of a parking service. Forwarding at the HTTP level means that a parked domain redirects HTTP traffic to a parking service's server

<sup>6</sup>More screen shots of parked domains can be found at [http://research.microsoft.com/URLTracer/Ads\\_on\\_Parked\\_Domains.htm](http://research.microsoft.com/URLTracer/Ads_on_Parked_Domains.htm)

through 3XX HTTP reply message [Fielding et al. 1999]. Forwarding at the HTML level means that a parked domain returns HTML content that forwards to a parking service’s server through META tags [Raggett et al. 1998].

The ads-portal page pointed to by the parked domain can be in one of two forms: one-click page and two-click page. In one-click page, the sponsored ads are shown immediately to the user. In two-click page, the user lands into a page that show ads subcategories (index page). When the user clicks on one of the shown subcategories, he/she will be shown a page with sponsored links falling under the clicked category.

Recently, the domain parking business has pushed the edge and some services started serving, in addition to the ads content, some useful information for the purpose of better monetizing the traffic. However, in this paper, the term Parking Service refers to the typical parking service that provide text-based ads content and no other real content. In this paper, we use “parking service” and “advertisement syndication service” interchangeably. Also, when we refer to ads-portal domains, we mean text-based ads-portal domains.

### 3. DATA SETS COLLECTION

As stated in Section 1, this work has two goals: developing an ads-portal domain identifier and measuring the prevalence of (typo) ads-portal domains in the Internet and how often users access (typo) ads-portal domains. To address these goals, we created eight different data sets: COM-Zone, NET-Zone, COM-Trace, NET-Trace, ORG-Trace, BIZ-Trace, INFO-Trace and Positive-Negative-Samples. The following sections explain the data sets in details.

#### 3.1 Zone Data Sets

COM-Zone and NET-Zone data sets were extracted from a couple of top-level domain (TLD) zone files: the *.com* zone file and the *.net* zone file<sup>7</sup>. The *.com/.net* zone file consisted of all the domains that are registered under *.com/.net* TLD<sup>8</sup>. The *.com/.net* zone file had  $\sim 76/12$  million domains. We randomly selected 200,000 domains from each of the zone files. For each domain in the sampled sets, we tried to download its web content<sup>9</sup>. COM-Zone/NET-Zone represents the domains that were successfully downloaded from the 200,000 random sample of the *.com/.net* zone file. Table I shows the number of domains in COM-Zone and NET-Zone sets. We created these sampled zone sets for two reasons. First, we used them to create Positive-Negative-Samples data set, which was used in devising the identification methodology. Second, we used them, as described in Section 6, to measure the prevalence of (typo) ads-portal domains in the Internet of ads-portal domains.

#### 3.2 DNS Trace Data Sets

The trace data sets are COM-Trace, NET-Trace, ORG-Trace, BIZ-Trace and INFO-Trace. These data sets were extracted from DNS trace files. We collected two-month DNS trace files corresponding to the months of June and July of 2008. The DNS trace files were collected from the name servers of UCI -University of California, Irvine- campus. The

<sup>7</sup>The zone files have been provided by VeriSign ([www.verisign.com](http://www.verisign.com)) during the month of July, 2008.

<sup>8</sup>VeriSign ([www.verisign.com](http://www.verisign.com)) has the policy of excluding from the zone files all the registered domains that are not associated with any name server.

<sup>9</sup>When we download a sampled domain, we first try to download it with the *www* prefix. If that fails, we try to download it without the *www* prefix.

Table I. Data Sets

Data Set	#Domains
COM-Zone	168,298
NET-Zone	160,554
COM-Trace	89,063
NET-Trace	86,985
ORG-Trace	92,686
BIZ-Trace	4,305
INFO-Trace	11,581
Positive-Negative-Samples	2400

trace files consisted of resource records that corresponded to user resolution requests for domains other than *\*.uci.edu*. The main reason of using the trace files was to collect a representative sample of web domains that users, within the campus, visited. Since HTTP requests are typically preceded by a DNS request of A-type or CNAME-type [Mockapetris 1987], we filtered out, from the trace files, all the resource records that were not of A-type or CNAME-type. This resulted in a set of  $\sim 15$  million domains<sup>10</sup>.

From this filtered set<sup>11</sup>, we created five sample sets: COM-Sample, NET-Sample, ORG-Sample, BIZ-Sample and INFO-Sample. COM-Sample was 100,000 random samples of all the *\*.com* domains found in the filtered traces that were of two levels, such as *yahoo.com*. The other sets were sampled in the same way as the first one except that we used different top level domains (*\*.net*, *\*.org*, *\*.biz* and *\*.info* for NET-Sample, ORG-Sample, BIZ-Sample and INFO-Sample, respectively)<sup>12</sup>. Similar to COM-Zone and NET-Zone, we tried to download the web content of those sampled domains. COM-Trace, NET-Trace, ORG-Trace, BIZ-Trace and INFO-Trace represent the domains that were successfully downloaded from COM-Sample, NET-Sample, ORG-Sample, BIZ-Sample and INFO-Sample, respectively. Table I shows the number of web domains in the trace sets.

The reason for creating the trace sets is to measure how often users access (typo) ads-portal domains. Note that, when sampling the traces, we considered the three-level domains that had the prefix *www* as two-level domains since most of the time the two-level domains map to the *www* subdomain; i.e., *anydomain.com* maps to *www.anydomain.com*. Hence, an access to *www.anydomain.com* is roughly considered an access to *anydomain.com*.

### 3.3 Positive and Negative Samples Data Set

Positive-Negative-Samples set consists of 2,400 web domains. Six hundred of them are positive (ads-portal) samples and the remaining are negative (non-ads-portal). This data set is used to in Sections 4 and 5 to evaluate the effectiveness of several content-based properties and the performance of the identification methodology.

For the negative samples, we collected 1,800 different domains from all Yahoo Directory Top categories [Yahoo 2007] to cover a wide range of different types of web documents.

<sup>10</sup>Note that an/a A/CNAME resource record could correspond to a non-web domain resolution request or a web domain resolution requests that is not initiated by a user/browser. We acknowledge this limitation.

<sup>11</sup>Note that we did exclude, from the filtered set, the traffic initiated by the machines involved in the data collection.

<sup>12</sup>The sizes of BIZ-Sample and INFO-Sample were less than 100K because the total number of the two-level accessed *\*.biz* domains was 5,185 domains and total number of the two-level accessed *\*.info* domains was 15,457. Thus, we set BIZ-Sample/INFO-Sample to all the two-level accessed *\*.biz/\*.\*.info* domains

The negative samples were distributed almost equally over the different categories<sup>13</sup>. For the positive samples, the process of collecting them went through the following steps:

(1) We obtained a list of fourteen well-known parking services and for each, we collected few samples of ads-portal (parked) domains. Most of those samples were retrieved from the parking services' web sites and the remaining were collected from other domains such as *parkquick.com*. This set of ads-portal domains was by no mean comprehensive since it had only few samples for each parking service.

(2) From these samples, we extracted the signatures of the fourteen parking services. The signature of a parking service is a regular expression commonly found in the ads-portal (parked domain) pages served by that parking service. Mostly, the signature of a parking service included the domain name/URL of the parking service. The signatures were extracted manually.

(3) From this small set of ads-portal domains, we created a larger more comprehensive set using the fourteen different signatures. We used our COM-Zone and NET-Zone data sets (see Section 3.1) to find all the domains in these two sets that had any of the fourteen signatures. The results of this step was fourteen different large sets for fourteen different parking services.

(4) For each of the fourteen different parking sets, we randomly selected 100 domains and manually inspected them to eliminate the false positives.

(5) Using these manually filtered random sets, we extracted 600 ads-portal domains distributed equally, if possible, over the fourteen parking services.

The reason for collecting the parked domains samples from many different parking services was that we wanted our positive set to be comprehensive enough to cover most ads-portal templates and patterns. Note that we cannot rely on parking service signatures as a way of detecting ads-portal domains for the following reasons:

- (1) Relying on signatures means that we cannot identify ads-portal domains fetching ads from advertisement syndication services with unknown signatures. In Section 6.3, we show that  $\sim 26\%$  of the ads-portal domains detected by our identification methodology do not match any signature of our fourteen parking signatures.
- (2) Detecting ads-portal domains through signatures involves many false positives. From our manual inspection, we found that many false positives are involved. In fact, one of the signatures lead to more than 20% false positives.

We download the web content for all the domains in the Data Set. When we download the web content of a domain, we download all the embedded files needed to display the HTML index page and treat them as one page. For example, If the index page is frame-based, we download all the documents to which a frame refers. If the domain is forwarding to another domain, we download the web content corresponding to the final destination.

#### 4. CONTENT-BASED PROPERTIES

As we look through different ads-portal domains in Positive-Negative-Samples data set, we observe that they promote their ads-content in similar fashion. Hence, we identify a

<sup>13</sup>We believe that the categorization of the web documents in Yahoo Directory *dir.yahoo.com* is highly accurate because the categorization is done manually by staff editors.

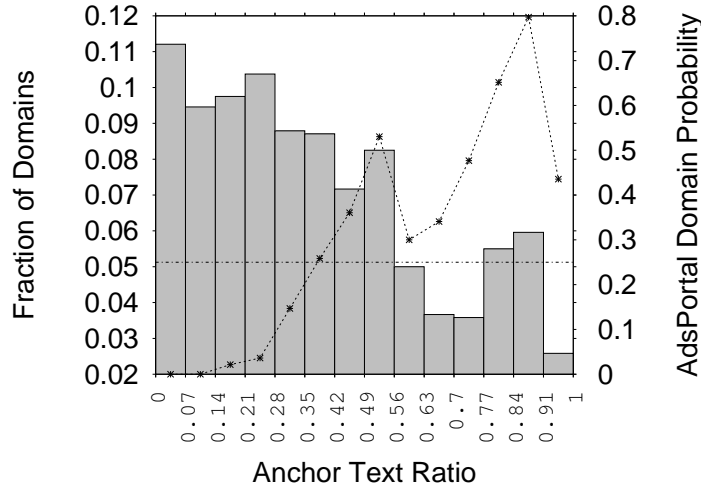


Fig. 3. This figure shows the posterior probability (the line with points and the right y-axis) and the prior probability (the horizontal line and the right y-axis) of being an ads-portal domain. The fractions of domains for different value ranges are also shown by the bar graph and the left y-axis.

set of content-based properties/features that distinguish ads-portal domains from others. In the remainder of this section, we explore several content-based properties and show their effectiveness, in terms of highly increasing/decreasing the ads-portal likelihood, through detailed statistical analyses. Note that in this section, when we refer to “data set”, we mean Positive-Negative-Samples data set, which is defined in Section 3.3. We use similar techniques, in analyzing the properties, to the ones presented in [Ntoulas et al. 2006], but we use an additional metric (*APCR*), which is explained in the following section, to evaluate the effectiveness of the properties. In Section 4.1, we explain our metrics and the plots in the figures. This description also applies to the other sections.

#### 4.1 Anchor Text Ratio

The content of an ads-portal domain is mainly ads-related without any real content. Mostly, the ads are shown in the form of lists of textual hyperlinks. Since the main content of an ads-portal page is ads-related anchor text, we would expect that most of the characters shown in the page belong to the anchor text. In light of this observation, we define a feature called Anchor Text Ratio that measures the intensity of the anchor text in a web page. Specifically, Anchor Text Ratio of a domain  $D$  is defined as follows:

$$\text{AnchorTextRatio}(D) = \frac{\text{Number of Characters in the Anchor Text of } D}{\text{Total Number of NonMarkup Characters in } D}$$

Note that only alpha-numeric characters are counted. Due to the nature of most ads-portal domains, we intuitively think that Anchor Text Ratio of ads-portal domains would be high and distinguishable from many of non-ads ones. To investigate the effect of Anchor Text Ratio, we plot several distributions that collectively show the effectiveness of Anchor Text Ratio. The distributions are shown in Figure 3.

Figure 3 - like all other figures in this section - combines two different distributions:

domain fraction and posterior probability distributions. In addition, the figure shows the prior probability. All the distributions are computed over Positive-Negative-Samples data set. The x-axis depicts a set of value ranges of Anchor Text Ratio (in Figure 3, the first range refers to the web domains in the data set having Anchor Text Ratio values between 0 and 0.07). The bar graph and the left y-axis depicts the percentage of web domains in our data set (Positive-Negative-Samples) that falls into a specific range. For example, the left-most bar in the graph indicates that 11.2% of the web domains in the data set have Anchor Text Ratio values between 0 and 0.07. The line points and the right y-axis depicts the posterior probability

$$P_p = P\{D \text{ is an Ads Portal Domain} \mid \text{AnchorTextRatio}(D) \in \text{Interval } I\}$$

distribution over the value ranges ( $I$ 's) of Anchor Text Ratio. The horizontal line and the right y-axis depicts the prior probability of being an ads-portal domain

$$P_a = P\{D \text{ is an Ads Portal Domain}\}$$

, which is equal to the fraction of the ads-portal domains in our data set (Positive-Negative-Samples); i.e.,  $600/2400 = 0.25$ . The prior probability horizontal line is plotted so that the difference between the prior and posterior probabilities becomes clearer.

Figure 3 shows how the posterior probability changes when Anchor Text Ratio changes. We can observe from the figure that, except for few cases, the ads-portal likelihood increases as Anchor Text Ratio increases. Thus, a (low) high Anchor Text Ratio value is a good indicator for being an (a non-)ads-portal domain.

In Figure 3, we can observe that the posterior probabilities  $P_p$  highly vary from the prior probability in almost the full range of the Anchor Text Ratio values. There are only few ranges (for instance, the range [0.35, 0.42]) at which the posterior probability and the prior one are close to each other. To quantify the posterior effect of the feature (how it affects the posterior distribution), we introduce a new metric called Average Posterior Change Ratio *APCR*. The definition of *APCR* is as follows:

$$APCR = \sum_{\forall \text{ Interval } I} P_p \text{ Change Ratio in } I \times \text{Fraction of Domains in } I \quad (1a)$$

where :

$$P_p \text{ Change Ratio in } I = \frac{|P_p \text{ in } I - P_a|}{\text{Max } P_p \text{ Displacement in } I} \quad (1b)$$

and :

$$\text{Max } P_p \text{ Displacement in } I = \begin{cases} 1 - P_a & \text{if } P_p \text{ in } I > P_a \\ P_a & \text{otherwise} \end{cases} \quad (1c)$$

Basically, the Posterior Change Ratio in  $I$  measures the relative displacement of the posterior probability  $P_p$  from the prior one  $P_a$ . For example, the value of 0 indicates 0 relative displacement; i.e.,  $P_p$  and  $P_a$  are equal, and the value of 1 indicates maximum relative displacement; i.e.,  $P_p$  is displaced from  $P_a$  to either ends, 1 or 0. Note that we choose to measure the ratio of displacement instead of the absolute displacement ( $|P_p - P_a|$ ) because the influence of the feature value on the posterior probability is clearer if explained by the ratio instead of the absolute value. For example, the posterior probability of 0 is as distinguishing as the posterior probability of 1; i.e., we can accurately identify

Table II. APCR Values for different Features

Feature	APCR
Anchor Text Ratio	56%
Common Link Ratio (N=1)	38%
Common Link Ratio (N=2)	42%
Common Link Ratio (N=3)	47%
Common Link Ratio (N=4)	46%
Parametrized Link Ratio	90%
Number of Hyper-Link Images	51%
Number of Links	47%
Number of Non-Markup AlphaNum	50%
Frame-Based	24%
Lengthy Link	39%

(non-)ads-portal domains with 100% accuracy. If we use the ratio, the Posterior Change Ratio will be 1 in both cases. But if we use the actual absolute displacement ( $|P_p - P_a|$ ), the absolute displacement will 0.25 in the case of 0 and 0.75 in the case of 1. Therefore, the absolute displacement gives these two cases totally different values even though the posterior probability of 0 is as distinguishing and effective as of 1.

The Average Posterior Change Ratio *APCR* is the average of Posterior Change Ratio's of all the feature value ranges  $I$ 's weighted by the ratio of domains in  $I$ 's. We choose a weighted average instead of an equally-weighted average (arithmetic mean) because the effect of the Posterior Change Ratio, for an interval  $I$ , on the average value should be proportional to the ratio of domains, in  $I$ . Note that the weighted average would be more resistant to noises than the arithmetic mean because a noisy interval would generally have a low ratio of domains falling into it. Again, *APCR* may take a value between 0 and 1. For example, if we have the  $P_p$  value the same as the  $P_a$  value in all  $I$ 's, *APCR* would be equal to 0. One the other hand, if we have the  $P_p$  value equals to either 1 or 0 in all  $I$ 's, the *APCR* would be equal to 1.

As Table II shows, the *APCR* value for the Anchor Text Ratio is 56% which indicates that the Posterior Change Ratio, on average, is 56%. Note that the optimal value, in terms of distinguishability, is 100%. This value is reasonably high and emphasizes on the effectiveness of Anchor Text Ratio on the posterior probability distribution. The *APCR* value shows the strong correlation between the Anchor Text Ratio and event of being an ads-portal domain, and that Anchor Text Ratio highly affects the ads-portal likelihood.

## 4.2 Common Link Ratio

We observe that anchor text of different links embedded in an ads-portal domain tend to share words. That is, we find some degree of coherence and commonality, in terms of word sharing, among anchor text of ads-links shown in an ads domain page. This degree of coherence varies from strongly to loosely coherent. But generally, there is some degree of coherence in many of the ads-portal domains we come across.

An ads-portal domain that is strongly coherent has most of its anchor text of the ads links sharing the same keyword(s) related to some topic. The anchor text in such a domain may promote for a specific merchandise, e.g. cameras. After further investigation in the advertisement syndication (parking) service business, we found that a number of advertisement syndication (parking) services generated contextual ads that were relevant to the domain name of the ads-portal domain or relevant to a set of keywords fixed by the domain name

owner of the ads-portal domain. We believe this is the reason why there are many strongly coherent ads-portal domains. Strongly coherent ads-portal domains are good targets for type-in traffic [Wikipedia 2008]. In type-in traffic, a user types a domain name in the address bar and expects to land on a page that shows ads related to the domain name. Since showing related ads may increase the income of the advertisement syndication (parking) services, probably from type-in traffic, we expect that many ads domains are strongly coherent. Figure 1(a) shows an example of an ads-portal/parked domain page that is strongly coherent. Note that almost all the links in the figure have the word “nail”.

On the other hand, there are loosely coherent ads-portal domains. In a loosely coherent ads domain, the page presents a set of ads links that represent multiple topics. For example, it could show ads links related to finance and ads links related to technology at the same time. In those domains, there exists some common words in different links but at lesser degree than the strongly coherent ones. Figure 1(b) shows a loosely coherent ads-portal domain. It can be observed from the figure that some links, such as “Airline tickets” and “Mortgage”, do not share words with any other link in the page. But there are other links, such as “Free credit report” and “Credit Card Application”, that share some word(s).

To capture the coherence at various degrees (strong, loose) in ads-portal domains, we define a new feature called Common Link Ratio. Common Link Ratio  $CLR_N$  of a domain  $D$  is defined as follows:

$$CLR_N(D) = \frac{\# \text{ links sharing words with other } N \text{ links in } D}{\# \text{ links in } D}$$

In the above equation, “sharing words” means sharing one or more. Words in links are stemmed using the Porter stemmer [Porter 1980] and the stop words are removed. We compute Common Link Ratio for different values of  $N$ : 1, 2, 3, and 4. Note that we compute Common Link Ratio for different  $N$  values to capture different levels of coherence web domains have among their links. Thus,  $N$  represents a correlation factor.

Figures 4, 5, 6 and 7 show the distributions (fraction of domains, posterior and prior) of Common Link Ratio with the  $N$  values of 1, 2, 3, and 4. We can observe the effect of the correlation as we compare the different intervals of the same figure. The four figures show that as the Common Link Ratio value increases, the prevalence of ads-portal domains increases. This increasing trend makes intuitive sense - many advertisement syndication (parking) services feed ads-portal domains with ads that are relevant to their domain names or some sets of keywords. Consequently, the anchor text in the ads links is correlated and the posterior probabilities have increasing trends in the figures.

Also, we can observe the effect of the correlation as we compare the same intervals in the four figures. If we compare the distributions (bar graph and line points) at the right end intervals of Figures 4, 5, 6 and 7, we can observe that as the correlation factor ( $N$ ) increases, from 1 to 4, the bar graphs (fraction of domains) decrease and line points values (posterior probability) increase. For example, if we compare the bar graphs and line points at the interval [0.84-91) of the four figures, we can see that as the correlation factor ( $N$ ) increases, the bar graph decreases and the line point value increases. This implies that for a web domain, the stronger the correlation is, the higher the likelihood of being an ads-portal is.

The four figures show that the posterior distributions largely deviate from the prior ones. To accurately quantify this deviation, we compute  $APCR$  value of Common Link Ratio.

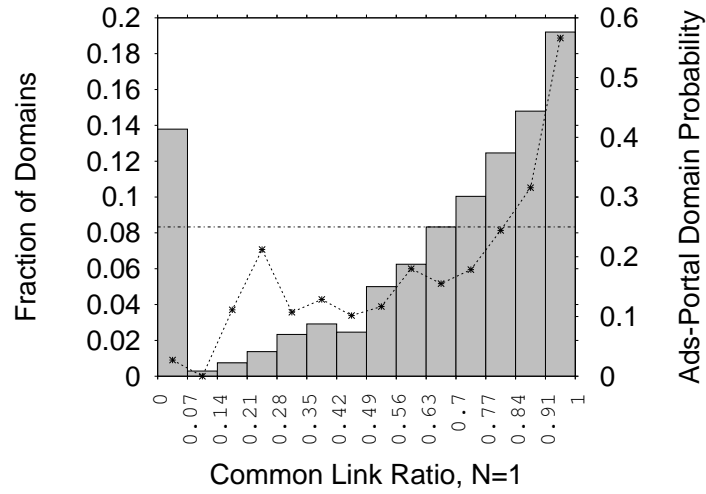


Fig. 4. This figure shows the posterior probability (the line with points and the right y-axis) and the prior probability (the horizontal line and the right y-axis) of being an ads-portal domain. The fractions of domains for different value ranges are also shown by the bar graph and the left y-axis.

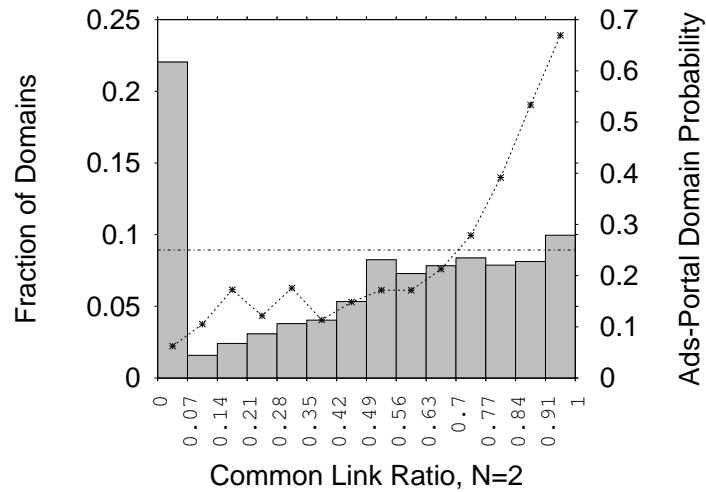


Fig. 5. This figure shows the posterior probability (the line with points and the right y-axis) and the prior probability (the horizontal line and the right y-axis) of being an ads-portal domain. The fractions of domains for different value ranges are also shown by the bar graph and the left y-axis.

Table II shows the *APCR* values of Common Link Ratio, for different  $N$  values, which range from 38% to 47%. These *APCR* values imply that the ads-portal likelihood significantly changes when considering the Common Link Ratio.

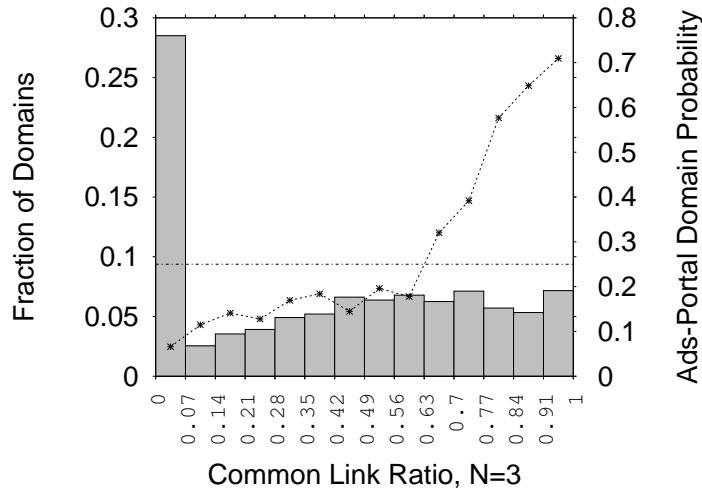


Fig. 6. This figure shows the posterior probability (the line with points and the right y-axis) and the prior probability (the horizontal line and the right y-axis) of being an ads-portal domain. The fractions of domains for different value ranges are also shown by the bar graph and the left y-axis.

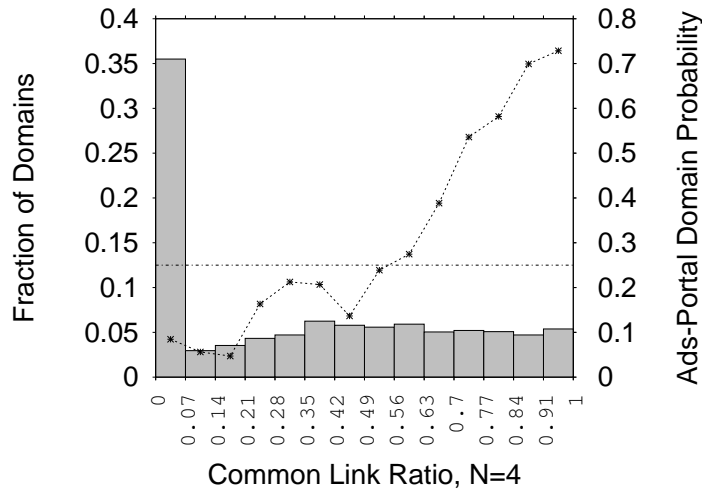


Fig. 7. This figure shows the posterior probability (the line with points and the right y-axis) and the prior probability (the horizontal line and the right y-axis) of being an ads-portal domain. The fractions of domains for different value ranges are also shown by the bar graph and the left y-axis.

### 4.3 Parametrized Link Ratio

We would intuitively expect that many of the links in an ads-portal domain are associated with target URLs<sup>14</sup> that are parametrized (URLs with some parameters). This is because

<sup>14</sup>Target URLs are URLs that are specified in the *href* attribute of the anchor text HTML element  $\langle a \rangle$

ads-portal domains may need to convey some information to the ads syndication server. For example, *ClientID* parameter, which identifies the ads-portal domain from which the link is clicked, needs to be conveyed to the ads-portal syndication server so that the client (ads-portal domain) can be ultimately paid<sup>15</sup>. Another example of a URL parameter is *QueryID* that informs the ads-portal syndication server of the topic of ads-links that are to be generated and sent back to the user. To further explore the prevalence of such parameters in the target URLs, we define ParametrizedLinkRatio *PLR* metric of a domain *D* as follows:

$$PLR(D) = \frac{\# \text{ of Links with Parametrized Target URLs in } D}{\# \text{ of Links in } D}$$

Clearly, the above metric measures the intensity of the links with parametrized target URLs in a domain. We observe from our data set that ads-portal domains, in most cases, attach parameters to the target URLs by one of the following ways:

- (1) **Using “?” Character:** The parameters are added after “?” character in the target URL.
- (2) **Using “id” Attribute:** “id” attribute of the anchor text HTML element points to a script that adds parameters to the target URL.
- (3) **Using “onclick” Attribute:** “onclick” attribute of the anchor text HTML element has a script that adds parameters to the target URL.

In addition, we observe that a significant portion of the ads-portal domains in our data set include a number in the path section of the target URLs. These numbers seem to be automatically generated and differ for different domains (the same in a single domain). These numbers could convey some information to the ads syndication server such as *ClientID*. Thus, we consider these numbers as parameters that convey some information from the client to the server. For every link in a domain that is associated with a target URL that has one of the above properties or a number in its path, we mark it as a parametrized link. Figure 8 shows the distributions related to Parametrized Link Ratio feature.

We can observe from the figure that the posterior probability is highly variant from the prior one in most of the range values. It is clear from the figure that the posterior probability has an increasing trend and the event of being an ads-portal domain is highly correlated with large values of Parametrized Link Ratio. Similar to the previous sections, we compute the APCR value for Parametrized Link Ratio feature. As shown in Table II, the APCR value is 90%; i.e., Posterior Change Ratio, on average, is 90%. That shows how the posterior probability is significantly effected by Parametrized Link Ratio feature and, consequently, this feature is strongly correlated with the event that a domain is an ads-portal.

#### 4.4 Number of Image Links

Since we are trying to identify text-based ads-portal domains, which show most of their ads through anchor text, we expect that many of text-based ads-portal domains have few image links, if any. That might be the case because advertisement syndication (parking) services may prefer to reduce their bandwidth costs by reducing the size of the served content. To

<sup>15</sup>*ClientID* parameter is frequently used in sponsored links.

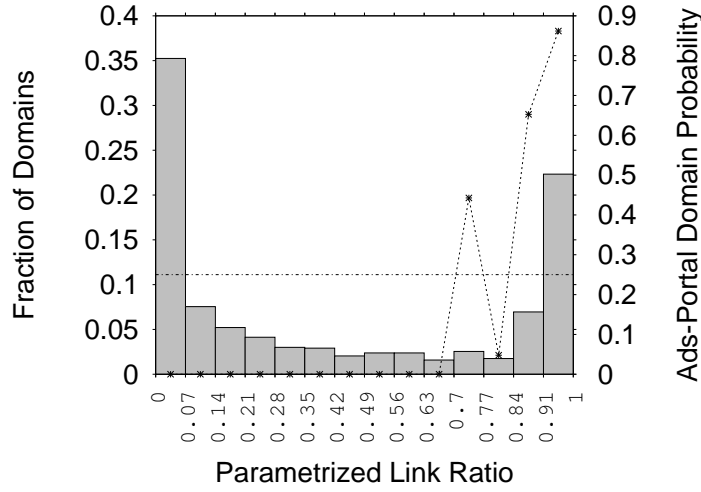


Fig. 8. This figure shows the posterior probability (the line with points and the right y-axis) and the prior probability (the horizontal line and the right y-axis) of being an ads-portal domain. The fractions of domains for different value ranges are also shown by the bar graph and the left y-axis.

further investigate the effect of the number of image links, we plot the distributions (bar graph and line points) of the number of image links as shown in Figure 9.

Figure 9 shows, with few exceptions, that the posterior probability has a descending trend, i.e., as the number of image links increases, the posterior  $P_p$  probability decreases. The posterior  $P_p$  probability peaks when the number of image links is 0. This decreasing trend complies with the nature of those text-based ads-portal domains as they show their ads in the form of text-based ads listing with few images. From Figure 9, we can conclude that few number of image links might be a good indicator that a given domain is an ads-portal. Note that in Figure 9, there is one exception (when the number of images is 11) where the posterior  $P_p$  probability unexpectedly spikes. The reason for that is related to the ads-portal pattern produced by one parking service, where it uses many image links as its ads-links.

It can also be observed from the figure that the posterior probability is highly different from the prior one at many of the intervals. Similar to the previous sections, we compute the  $APCR$  value of Number of Image Links to quantify the change in the posterior probability. Table II shows that  $APCR$  value of Number of Image Links feature is 51%; i.e., Posterior Change Ratio, on average, is 51%. This shows that the ads-portal likelihood is largely effected when we consider the number of image links.

#### 4.5 Number of Links

An ads-portal domain mostly shows ads links and no real content. Thus, the number of links in a domain could be useful in identifying (non-)ads-portal domains. To further explore the usefulness of Number of Links property, in terms of identifying ads-portal domains, we plot the distributions as shown in Figure 10. We can observe from the figure that the ratio of ads-portal domains increases when the number of links is moderate (between 21 and 70). That is, very few/large number of links is an indicator that a given domain is

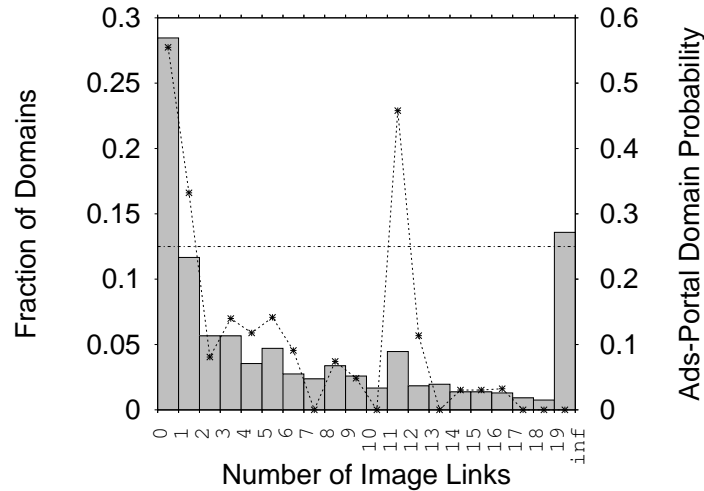


Fig. 9. This figure shows the posterior probability (the line with points and the right y-axis) and the prior probability (the horizontal line and the right y-axis) of being an ads-portal domain. The fractions of domains for different value ranges are also shown by the bar graph and the left y-axis.

not an ads-portal.

We can observe from the figure that the posterior probability  $P_P$  is highly variant from the prior one in most of the range values. Similar to the previous sections, we compute the *APCR* value of Number of Links. As shown in Table II, the *APCR* value is 47%; i.e., Posterior Change Ratio, on average, is 47%. That shows how the posterior probability is significantly effected by Number of Links and, consequently, Number of Links is strongly correlated with the event that a domain is an ads-portal.

#### 4.6 Number of Non-Markup Alphanumeric Characters

We observe that ads-portal domains tend to have small content in terms of the amount of alpha-numeric text shown in their pages. One possible justification is that an advertisement syndication (parking) service does not want to lose the visitor attention by showing him/her so much textual content in the page. To further investigate this property, we plot the distributions as shown in Figure 11. Mostly, the posterior probability decreases as the Number of Non-Markup Alphanumeric Characters increases. Specifically, the range of values between 300 and 1500 shows a major increase in  $P_p$ . Thus, limited number of non-markup Alphanumeric Characters might be a good indicator that a domain is an ads-portal.

Also, Figure 11 shows that posterior probability  $P_P$  drastically varies from the prior one at many intervals. Similar to the previous sections, we compute *APCR* value of this feature. Table II shows that the *APCR* value of Number of Non-Markup Alphanumeric Characters is 50%, which indicates that Posterior Change Ratio, on average, is 50%. The *APCR* value suggests that this property significantly changes the ads-portal likelihood.

#### 4.7 Frame-Based Domains

We observe that a number of ads-portal domains are frame-based; i.e., they use the frame HTML structure to fetch ads content from advertisement syndication (parking) services.

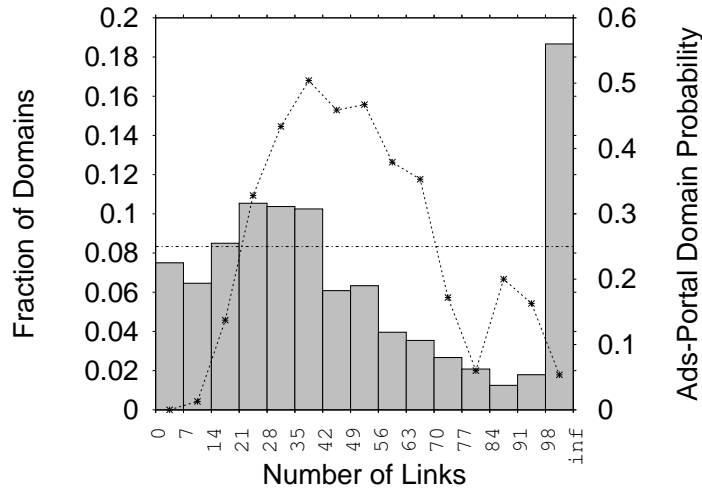


Fig. 10. This figure shows the posterior probability (the line with points and the right y-axis) and the prior probability (the horizontal line and the right y-axis) of being an ads-portal domain. The fractions of domains for different value ranges are also shown by the bar graph and the left y-axis.

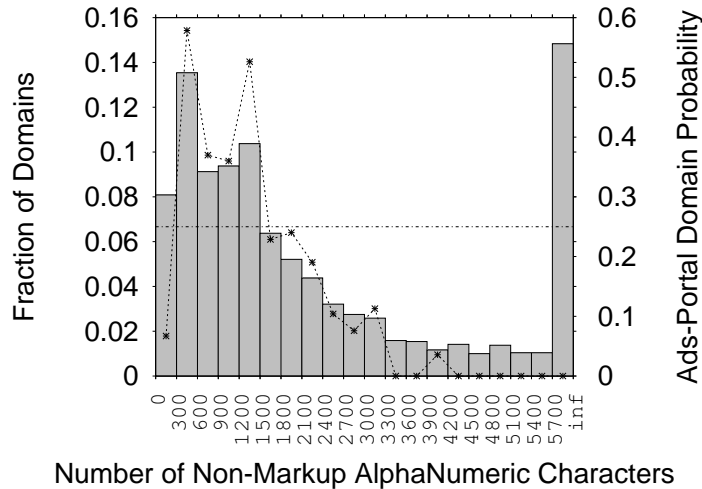


Fig. 11. This figure shows the posterior probability (the line with points and the right y-axis) and the prior probability (the horizontal line and the right y-axis) of being an ads-portal domain. The fractions of domains for different value ranges are also shown by the bar graph and the left y-axis.

That naturally comes from how some of the advertisement syndication (parking) services provide their ads-content: an ads-portal domain obtains the dynamic ads content from the parking/syndication service using an HTML frame that refers to the advertisement syndication (parking) service.

We compute the ratio of domains that are frame-based and the resulting posterior probability from our Positive-Negative-Samples data set. A ratio of 10% of the web domains

in our data set are frame-based and 70% of them are ads-portal domains. This means that the ads-portal likelihood increases from 25% (prior) to 70% (posterior). On the other hand, 90% of the web domains are not frame-based and only 20% of them are ads-portal domains. This shows how Frame-Based feature would significantly change the ads-portal likelihood if the domains are frame-based.

The *APCR* value of Frame-Based feature, as shown in Table II, is 24%; i.e., the Posterior Change Ratio, on average, is 24%. Even though the *APCR* value of Frame-Based is less than the previous features, it is still influential on the posterior distribution. Note that from Frame-Based feature, the Posterior Change Ratio is high when the feature is true, but Posterior Change Ratio is low when the feature is false. The low value of Posterior Change Ratio, when the feature is false, reduces the *APCR* value to 24%.

#### 4.8 Lengthy Link Domains

We observe that a number of ads-portal domains tend to have the size of their anchor text small, in terms of the number of alpha-numeric characters, per link. One possible justification is that showing many links with lots of textual content may distract the visitors and encourage them to leave the domains without clicking on any sponsored link. The prevalence of short textual links in ads-portal domains is clearly more observable in ads-portal domains that show their ads at two levels: one level serves as an index page and the other level shows the actual sponsored-links.

To further investigate the effect of the size of the links, we define new boolean feature called Lengthy Link Domains that is set to *True* if the domain has lengthy links and *False* otherwise. Specifically, the feature is set to 1 whenever the page has 3 or more links each of which with anchor text of 30 or more alpha-numeric characters.

A ratio of 65% of the web domains in our Positive-Negative-Samples data set have this feature set to 0 and 36% of them are ads-portal domains. That is, the likelihood of being an ads-portal increases from 25% (prior) to 36% (posterior). On the other hand, 35% of the web domains have their feature values set to 1 and only 96% of them are non-ads-portal domains. That is, the likelihood of not being an ads-portal increases from 75% (prior) to 96% (posterior). The *APCR* value of Lengthy Link feature, as shown in Table II, is 39%; i.e., the Posterior Change Ratio, on average, is 39%. Clearly, these measures indicate a strong correlation between Lengthy Link domains and the posterior probability. The thresholds used to set Lengthy Link domains feature are subjectively selected and there might be room for improvement, but we leave that as part of our future work.

### 5. USING CLASSIFIERS TO IDENTIFY ADS-PORTAL DOMAINS

In the previous section, we analyze several properties/features of ads-portal domains and show how specific value ranges of those features could highly increase/decrease the ads-portal likelihood. However, using each feature individually may not be helpful or may lead to a large range of false positives. Thus, it will be more effective to combine multiple features and leverage their discriminative capabilities.

One way to combine the features is to consider them as one feature vector and feed them to a machine learning classifier. Thus, we combine the features described in Section 4 into one feature vector of 11 features (for Common Link Ratio, we consider  $N$  values of 1, 2, 3 and 4) and compare several classification algorithms. Particularly, we try the following classifiers: Random Forest [Breiman 2001] (with 20 trees and 10 random features), Decision Tree (C.45) [Quinlan 1993], Support Vector Machines (SVM) [Drucker

Table III. Performance Metrics Values(%) of Several Classifiers - Feature vector consists of only the 11 content-based features

Classifier	Positive Recall	Positive Precision	Negative Recall	Negative Precision
Random Forest	99	94	98	99
SVM	96	92	97	99
Nearest Neighbor	97	93	97	99
K-Nearest Neighbor (K=10)	98	89	96	99
Unpruned C4.5 Decision Tree	96	94	98	99
Pruned C4.5 Decision Tree	98	93	98	99
Boosting C4.5 Decision Tree	98	90	96	99
Bagging C4.5 Decision Tree	99	93	98	99.7
RIPPER	97	91	97	99
Bayes Net	90	94	98	97
Neural Net	98	93	98	99
Decision Table	95	94	98	98
Boosting - Decision Stump	99.8	76	89	99

et al. 1999; Joachims 2001], K-Nearest Neighbor [Mitchell 1997], RIPPER [Cohen 1995], Bayesian Networks [Mitchell 1997], Neural Networks [Mitchell 1997], Decision Table [Kohavi 1995]. In addition, we try some optimization techniques to classifiers; namely, we try boosting(with 10 iterations) of Decision Tree [Freund and Schapire 1995; Quinlan 1996], bagging(with 20 trees) [Breiman 1996; Quinlan 1996] of Decision Tree, and boosting(with 10 iterations) of Decision Stump [Iba and Langley 1992; KAWAKITA et al. 2005].

We train the classifiers with Positive-Negative-Samples data set, which is explained in Section 3.3. We use the WEKA java classes [Witten and Frank 2005] to run and test different machine learning classification algorithms. One reliable way of evaluating the accuracy of different classifiers is to use the ten-fold cross validation method [Mitchell 1997]. In the ten-fold cross validation, we randomly divide the data set into 10 sets of equal size, perform 10 different training/testing steps in which each step consists of training a classifier on 9 sets, and then testing it on the remaining set. We take the average of the results as the accuracy of the classifier.

We use the positive/negative precision and recall values to summarize the performance of a classifier. The positive (negative) recall value shows the fraction of the ads-portal domains (non-ads-portal domains) that are correctly identified by the classifier among all ads-portal (non-ads-portal) domains fed to the classifier. The positive (negative) precision shows the fraction of ads-portal (non-ads-portal) domains among the set of domains classified as ads-portal (non-ads-portal) domains by the classifier.

Table III shows the performance results of the all of the above classifiers. We can see from the table that the Random Forest classifier shows the most superior performance in all of the four metrics: Positive Recall, Positive Precision, Negative Recall, and Negative Precision. Note that the bagging classifier performance is really close to the Random Forest one. The reason is that both classifiers are almost the same except that in Random Forest, there is some randomness involved in the selection the features. The exact description of bagging, Random Forest and the other classifiers is out of the scope of this paper. For more information about the mentioned classifiers and how text classification works, the reader is advised to refer to the above references and [Mitchell 1997; N. Cristianini and J. Shawe-Taylor 2000].

Even though the evaluation measures of the Random Forest classifier are good, we need

Table IV. Performance Metrics Values (%) of Several Classifiers - Feature vector consists of the 11 content-based features and the word-based features

Classifier	Positive Recall	Positive Precision	Negative Recall	Negative Precision
Random Forest	99	96	99	99.7
SVM	99	96	99	99.5
Nearest Neighbor	96	95	98	99
K-Nearest Neighbor (K=10)	98	94	98	99
Unpruned C4.5 Decision Tree	97	96	99	99
Pruned C4.5 Decision Tree	98	96	99	99
Boosting C4.5 Decision Tree	96	96	99	99
Bagging C4.5 Decision Tree	99	96	99	99.7
RIPPER	98	96	99	99
Bayes Net	97	95	98	99
Neural Net	98	96	99	99
Decision Table	96	96	99	99
Boosting - Decision Stump	99.8	76	89	99.9

to enhance the performance, especially for the ads-portal Precision. As a step to enhance the performance, we add another set of features that boosts the discriminative strength of all the classifiers. These added features are words and phrases (“for sale”, “sponsored listings”, etc.) that commonly/uncommonly exist in ads-portal domains. We add these words/phrases to the feature vector as boolean features corresponding to the occurrence of these words/phrases. In selecting these words/phrases, we first identify an initial set of words/phrases that commonly or uncommonly exist in ads-portal domains based on our personal observation. Then, we eliminate from this set the ones that do not affect the ads-portal likelihood. We end up with 56 different words and phrases. The average *APCR* of these words/phrases is 15% and that shows their influence on the ads-portal likelihood.

Table IV shows the evaluation results after adding these word-based features. Clearly, adding them improves the performance of nearly all the classifiers, especially in ads-portal precision and recall values. Again, Random Forest (along with Bagging) shows the most superior performance after the addition of the word-based features and thus, it is the classifier that we use in our measurement in Section 6. In Section 6.7, we further validate the accuracy of this classifier on different data sets. The accuracy results are close to the one shown in Table IV. Note that many of the other classifiers have comparable performance results.

For further comparison among classifiers with different feature sets, we train the above classifiers on feature vectors that consist of only the word-based features, which are described above. Table V shows the performance results of all the classifiers. We can see from the table that there is no clear winner. But, if we consider the total sum of all the four performance metrics as a way to compare the classifiers, Random Forest is among the best. This Random Forest classifier is clearly worse than Random Forest classifier shown in Table IV and Table III in almost all the four performance metrics.

## 6. MEASUREMENT RESULT

Several studies [Wang et al. 2006; McAfee 2007] show the existence of ads-portal and typo-squatting domains in the Internet. However, we do not know how many of ads-portal domains are in the Internet. Do they represent a trivial or a major ratio of the web domains? What percentage of ads domains are typo domains? Do Internet users access ads-portal domains? Are typographical errors the main reason why users access ads domains? This

Table V. Performance Metrics Values(%) of Several Classifiers - Feature vector consists of only the word-based features

Classifier	Positive Recall	Positive Precision	Negative Recall	Negative Precision
Random Forest	88	92	98	96
SVM	84	96	99	95
Nearest Neighbor	86	91	97	95
K-Nearest Neighbor (K=10)	73	95	99	92
Unpruned C4.5 Decision Tree	86	88	96	95
Pruned C4.5 Decision Tree	84	93	98	95
Boosting C4.5 Decision Tree	65	96	99	90
Bagging C4.5 Decision Tree	82	92	98	94
RIPPER	83	91	97	95
Bayes Net	81	93	98	94
Neural Net	87	92	97	96
Decision Table	76	96	99	92
Boosting - Decision Stump	65	96	99	90

Section explains several experiments that address the above questions. Knowing the ratio of ads-portal domains helps in better characterizing the Internet documents and may indicate the degree of success of this type of business. If the business is successful, more effort to better monetize the traffic is worthwhile. In addition, finding the ratios of typo-squatting ads domains is important in measuring the extent of the problem and knowing if some counter measures are needed to discourage the spread of typo-squatting practice.

### 6.1 Number and Ratio of Ads-Portal Domains in the Internet

Table VI shows the number of ads-portal domains and their corresponding ratios, relative to the data set size, in the zone data sets. Apparently, the number of ads-portal domains and their ratios are high. The ads-portal domains represent 28.3/25% of the COM-Zone/NET-Zone data sets<sup>16</sup>. This indicates that approximately quarter of the web domains that fall into the (two-level) \*.net and \*.com domains are ads-portals. These ratios show a large prevalence of ads-portal domains in the Internet. They also indicate that this type of business and monetization of traffic is to some extent successful as ads-portal domains represent a major ratio of the Internet domains. Note that the \*.net and \*.com represent a major portion of the Internet web domains, if not the largest.

### 6.2 Number and Ratio of Accessed Ads-Portal Domains

Table VII shows the numbers and ratios of ads-portal domains found in the trace sets: COM-Trace, NET-Trace, ORG-Trace, BIZ-Trace and INFO-Trace. Although these numbers are not as large as the ones in the zone sets, they represent considerable ratios of the accessed domains. Since ads domains only show ads listings, we initially expected that their access ratios would be extremely small. Unexpectedly, the computed ratios of ads-portal domains represent around 3-19.4% of the trace sets. These ratios show that ads-portal domains are successful, with respect to UCI traces, in attracting many visits.

At this point, it would be helpful to understand what are the factors that lead to this high access ratio. We think that there are three possible reasons leading to the high access ratio of ads-portal domains: Direct Search, Search Engines, and Typo-squatting. First,

<sup>16</sup>We sampled the TLD zone files in March of 2009 and the results were comparable. That is, 25.9/24.2% of the (two-level) \*.net and \*.com web domains were ads-portal domains.

Table VI. Numbers and ratios of ads-portal domains in the zone data sets

Data Set	#Ads Domains	%Ads Domains
COM-Zone	47,700	28.3%
NET-Zone	40,207	25%

Table VII. Numbers and ratios of ads-portal domains in the DNS trace data sets

Data Set	#Ads Domains	%Ads Domains
COM-Trace	3,627	4.1%
NET-Trace	4,257	4.9%
ORG-Trace	2,762	3%
BIZ-Trace	312	7.3%
INFO-Trace	2,241	19.4%

we believe that some users access ads-portal domains because they are performing Direct Navigation [Wikipedia 2008]; i.e., they are searching for some topics by bypassing search engines and directly typing in the address bars of their browsers some generic names related to their topics, hoping to find related content. An example of such a generic name ads-portal domain is *www.nail.com* as shown in Figure 1(a). Second, we think that search engines are contributing to the high access ratio of ads-portal domains. Wang et al. [2007] show how web-spammers trick the search engines to promote for URLs that redirect to ads-portal domains. In addition, there might be some ethical, or white-hat, SEO (Search Engine Optimization) techniques that ads-portal domains are using. Finally, we believe that some users access ads-portal domains because the domains happen to be typos. For example, *hotmail.com* is an ads-portal domain that is one error away from *hotmail.com*. In Section 6.6, we study how much the typo-factor is contributing to the high access of ads-portal domains. For the other two factors, we leave their analyses as part of our future work.

### 6.3 Parking Service Distribution in the Internet

An ads-portal domain fetches its ads content from a syndication/parking service. To understand how different syndication/parking services are contributing to the large number of ads-portal domains in the Internet, we collect the signatures of fourteen different well-known parking services. The signatures are basically the same as the one we use in Section 3.3. Given those signatures, we match each of the ads-portal domains, detected in our COM-Zone and NET-Zone data sets, to the corresponding parking service.

Table VIII shows the top ten parking services in terms of their relative shares in the set of detected ads domains in both COM-Zone and NET-Zone. Apparently, the shares of different parking services are not equal. *GoDaddy.com* has the largest share, followed by *DomainSponsor.com* parking service. In fact, these two parking services are responsible of  $\sim 58\%$  of the ads-portal domains<sup>17</sup>. Many of the remaining parking services are of comparable ratios and few of them have small ratios. Note that *Google.com* refers to Google's parking service (AdSense for Domains), not Google's search engine or Google

<sup>17</sup>We sampled the TLD zone files in March of 2009 and equivalent results were found. That is, *GoDaddy.com* has the largest share (48.7%), followed by *DomainSponsor.com* (15.4%). These two parking services have a combined share of  $\sim 64\%$  of the total detected ads-portal domains in *\*.com* and *\*.net* during March, 2009.

Table VIII. Top ten parking services in terms of their shares of ads-portal domains in the Internet - \*.com and \*.net

Parking Service	Ratio-Zone(%)
GoDaddy.com	44.3
DomainSponsor.com	13.5
RevenueDirect.com	3.9
Hitfarm.com	3.2
Fabulous.com	2.4
Sedo.com	2.4
NameDrive.com	1.9
TrafficZ.com	1.4
Parked.com	1.1
Google.com	0.4

AdSense (AdSense for content) [Google 2008]<sup>18</sup>.

We find that 26% of the ads-portal domains are of unknown parking service signatures. This shows the generality of our identification methodology; i.e., we train our classifier on fourteen ads-portal templates, and then we are able to detect many ads-portal domains served from unknown syndication/parking services. Table VIII suggests that *GoDaddy.com* is probably the most dominant parking service. Note that *GoDaddy.com* is also a domain name registrar and it parks all of the domains that customers register if the registered domains do not have any name servers. This automatic domain parking of the newly registered domains might be the reason why *GoDaddy.com* is the most dominant parking service.

#### 6.4 Parking Service Distribution in the Accessed Domains

Similar to Section 6.3, we find the distribution of parking services over the set of accessed ads domain, from UCI campus. Table IX shows the top ten parking services in terms of their shares in the set of accessed ads-portal domains found in the trace sets. Unlike Table VIII, *DomainSponsor.com* has the largest share of accessed ads domains with 25.6%, followed by *GoDaddy.com* with 17.5%, *Sedo.com* with 10.4% and then *NameDrive.com* with 6.9%. Many of the remaining are of comparable ratios and few has small ratios. The measures in Table IX shows that some parking services are successful in attracting many visits. For example, *DomainSponsor.com* (*GoDaddy.com*) has 1.2% (0.8%) share of the web domains in the trace sets.

#### 6.5 Typo-Squatting Domains in the Internet

To know the ratio of typo-domains, we must first know how we can identify typos. If we have a comprehensive list of target domains, we can use distance functions such as edit distance[Gusfield 1998] to identify typo domains. Unfortunately, we do not have a comprehensive list of all possible target domains. Even if we have it, we need a very efficient implementation of distance functions so we can efficiently run them over a large number of target domains. To avoid such complications, we resort to third-party typo identification services. Specifically, we use the well-known typo correction (spelling suggestion) services that are provided by Google [Google 2006] and Yahoo [Yahoo 2008]. Therefore, if a domain is corrected by either Google or Yahoo typo correctors, we consider the domain as

<sup>18</sup>More information can be found at [www.google.com/domainpark](http://www.google.com/domainpark).

Table IX. Top ten parking services in terms of their shares of accessed ads-portal domains - \*.com, \*.net, \*.org, \*.biz and \*.info

Parking Service	Ratio-Trace(%)
DomainSponsor.com	25.6%
GoDaddy.com	17.5%
Sedo.com	10.4%
NameDrive.com	6.9%
Parked.com	3.17%
Hitfarm.com	2.2%
RevenueDirect.com	1.3%
Google.com	1.1%
TrafficZ.com	0.8%
ParkingSpa.com	0.6%

a typo.

Since the typo correction services impose limitations on the number of typo correction queries per day, we take 5000 random samples of ads-portal domains found in each of the zone sets, COM-Zone and NET-Zone, and that results in two random sets of ads-portal domains: COM-Zone-5000 and NET-Zone-5000. Table X shows the number of typos and their corresponding ratios in COM-Zone-5000 and NET-Zone-5000. A ratio of 41%/39.8% of ads-portal domains in COM-Zone-5000/NET-Zone-5000 are typo domains (according to our definition of typo)<sup>19</sup>. In fact, the measures in Table VI and Table X imply that 12/10% of the (two-level) \*.com/\*.net web domains are typo-squatting domains<sup>20</sup>. The results show that typo domains represent a major portion of the ads-portal domains in \*.com and \*.net and, potentially, in the Internet. In other words, typo-squatting as a practice is highly contributing to the large number of ads-portal domains in the Internet and highly abusing the advertisement syndication business. Note that our finding about typo-squatting is different from the ones in [McAfee 2007; Banerjee et al. 2008; Wang et al. 2006] which show that many typo-squatting domains exist but do not show what ratio of ads-domains they represent.

For each of the fourteen parking services  $ParkService_i$ , we find the ads-portal domains in COM-Zone and NET-Zone parking with  $ParkService_i$  and how many of those domains are typo domains (using our typo definition). The ratios of these typos are shown in Figure 12. As shown in the figure, the typo ratios range from 16% to 64%. The mean of the typo ratios is 45.2% and the standard deviation is 12.2%. Apparently, for most of the parking services, typo-squatting domains represent considerable ratios of the ads-portal domains parking with them. In the figure, Hitfarm.com has the highest typo ratio (64%) and Fabulous.com has the lowest one (16%).

## 6.6 Typo-Squatting Domains in the Traces

Similar to Section 6.5, we find the typos in the trace data sets: COM-Trace, NET-Trace, ORG-Trace, BIZ-Trace and INFO-Trace. Table XI shows the number of typo ads-portal domains and their ratios, relative to the number of ads-portal domains in the sets. The table

<sup>19</sup>We sampled the TLD zone files in March of 2009 and the results were comparable. That is, a ratio of 41.4/39.3% of the detected ads-portal domains were typos.

<sup>20</sup>From Table VI and Table X,  $41\% \times 28.3\% = 12\%$  and  $39.8\% \times 2.5\% = 10\%$ .

Table X. Numbers and ratios of typos found in COM-Zone-5000 and NET-Zone-5000

Data Set	#Ads Typo Domains	% Ads Typo Domains
COM-Zone-5000	2,052	41%
NET-Zone-5000	1,988	39.8%

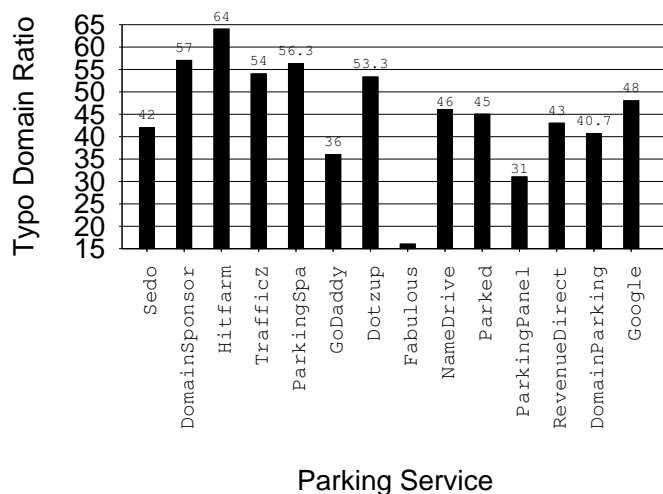


Fig. 12. Parking Service Typo Ratios (Zone) - In this figure, the typo ratio of a parking service  $ParkService_i$  is the number of typo ads-portal domains in COM-Zone and NET-Zone parking with  $ParkService_i$  over the total number of ads-portal domains in COM-Zone and NET-Zone parking with  $ParkService_i$

shows that 31.1-60.8% of ads-portal domains in the trace data sets are typos (according to our typo definition in Section 6.5). Apparently, these ratios are high and suggest that typo-squatting, with respect to UCI campus, highly contributes to the number of accessed ads-portal domains.

In fact, the measures in Table VII and Table XI imply that 2.2%, 2.9%, 1.8%, 3.5% and 6% of the (two-level) \*.com, \*.net, \*.org, \*.biz and \*.info UCI web accessed domains, respectively, are typo-squatting<sup>21</sup>. These ratios suggest that typo-squatting, with respect to UCI, is successful in attracting many visits.

For each of the fourteen parking services  $ParkService_i$ , we find the ads-portal domains in data trace sets parking with  $ParkService_i$  and how many of those domains are typo domains (using our typo definition). The ratios of these typo are shown in Figure 13. As shown in the figure, the typo ratios range from 41.7% to 71.9%. The average of the typo ratios is 53.3% and the standard deviation is 16.9%. Note that there is not any ads-portal domain found in the trace that has ParkingPanel.com's signature. Apparently, parking services, with respect to UCI campus, are making an advantage of users typographical errors, with Hitfarm.com having the highest typo ratio (71.9%) and GoDaddy.com having the lowest one (41.7%).

<sup>21</sup>From Table VII and Table XI,  $4.1\% \times 54.8\% = 2.2\%$ ,  $4.9\% \times 58.4\% = 2.9\%$ ,  $3\% \times 60.8\% = 1.8\%$ ,  $7.3\% \times 48.1\% = 3.5\%$  and  $19.4\% \times 31.1\% = 6\%$ .

Table XI. Numbers and Ratios of typos in ads-portal domains found in the trace data sets

Data Set	#Ads Typo Domains	%Ads Typo Domains
COM-Trace	1,986	54.8%
NET-Trace	2,488	58.4%
ORG-Trace	1,678	60.8%
BIZ-Trace	150	48.1%
INFO-Trace	697	31.1%

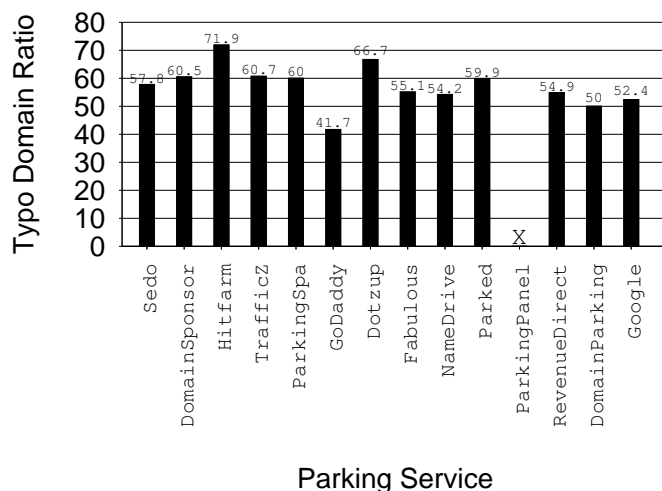


Fig. 13. Parking Service Typo Ratios (Trace) - In this figure, the typo ratio of a parking service  $ParkService_i$  is the number of typo ads-portal domains in COM-Trace and NET-Trace parking with  $ParkService_i$  over the total number of ads-portal domains in COM-Trace and NET-Trace parking with  $ParkService_i$

## 6.7 Accuracy Verification

To verify the classifier accuracy of the detected ads-portal domains in the zone sets (Com-Zone and Net-Zone) and the trace sets (COM-Trace, NET-Trace, ORG-Trace, BIZ-Trace and INFO-Trace), we performed manual verification. For each of these sets, we random sampled (around 40 samples) the set of its detected ads-portal domains and manually inspected the accuracy. The accuracy results of the detected ads-portal domains in the zone and trace data sets are shown in Table XII. The accuracy is high and close (or higher for the zone sets) to the one shown in Section 5. This shows that with different data sets, the accuracy of our classifier is high. Even though we design our classifier to work for English domains, the accuracy results (shown in Table XII) are high in data sets that probably include non-English domains.

## 7. FUTURE WORK

In Section 6, we show how often users access (typo) ads-portal domains. This opens the question about what could increase/decrease the profit and the incoming traffic of a parking service and a parked domain. As part of our future work, we try to answer this question in details. We want to answer this question from two different perspectives: the parking service perspective and the domainer perspective. That is, we want to know what the park-

Table XII. Accuracy results of the ads-portal domains detected in the zone and trace data sets

Data Set	Accuracy(%)
COM-Zone	100
NET-Zone	100
COM-Trace	95
NET-Trace	95
ORG-Trace	95
BIZ-Trace	100
INFO-Trace	100

ing service can legitimately do to increase its traffic and profit, and what the domainers can legitimately do to increase the traffic and profit (click-through ratios) of their parked domains. Also, in Section 6, we study the typo factor and show that typos represent a considerable ratio of the accessed ads-portal domains. As part of our future work, we also want to study how the direct search (type-in traffic) and search engines are increasing/decreasing the access to ads-portal domains.

Even though many of the typo-squatting domains map to ads-portal domains, we observe other forms of typo-squatting: typo-squatting domains forwarding to other non-ads domains and typo-squatting domains serving malicious content. In fact, There has been an incident that a typo-squatting domain of *www.google.com* was providing malware to its visitors [F-Secure 2005]. We would like to work on a framework that discourages access to the typo-squatting domains once and for all.

In spite of the high accuracy of our classifier, we believe that there is room for improvement. First, we need to minimize the number of word-based features. Second, we need to find other features, possibly network-based properties, that make the classifier more resilient when facing confusing non-ads-portal domains.

## 8. RELATED WORK

There is an enormous amount of work done in the area of web classification and mining. One of the most relevant work is the one in [Ntoulas et al. 2006] for web spam detection. Ntoulas et al. [2006] develop a binary classifier that identifies web spam pages from their content. A number of content-based heuristics for recognizing web spam pages are identified. Then, these heuristics are combined into a machine learning decision tree [Quinlan 1993] classifier. The classifier reaches 86.2/91.1% recall/accuracy values after boosting [Freund and Schapire 1995] it. Even though this work solves different problem, we develop our classifier in similar fashion. That is, we identify heuristics, verify the effectiveness of these heuristics by drawing their distributions, and finally, combine them into a machine learning classifier.

Esfandiari and Nock [2005] study advertisement filtering. They propose a methodology to filter out ads-related URLs from a web page through weighted majority algorithm. Their methodology works at the links level- i.e, identifying if a URL is an ads URL. Whereas, our methodology works at the page level and identifies if the whole page is an ads-portal page. Similarly, Kushmerick [1999] proposes a methodology based on inductive learning that automatically removes advertisement images from pages before they are downloaded. However, their methodology is dedicated to removing ads images but in our case the ads are mostly textual links. In addition to that, the methodology treats each image independently

whereas we treat the whole page as one unit. In terms of existing components, there is a Mozilla Firefox extension called AdsBlock [Mozdev 2008] that blocks unwanted ads content based on filters set by the user. Two types of filters are offered: simple (simple string of text) and regular expression. There is no automatic way of detection, so user intervention is required. Moreover, it treats each link independently but our system treats the whole page as one unit.

In [Wang et al. 2007], one type of web spam - redirection spam - is studied and analyzed. Redirection spam refers to web spam URLs that redirect the URLs to spammer-controlled domains, which are mostly in the form of an ads-portal domain. The authors propose a five layer double funnel model that shows how the web redirection spam works. The authors show important domains at each layer and their related characteristics. Those findings could be helpful for search engine ranking algorithms to be more robust against spam. This work shows one way of abusing the ads-portal domains using the redirection spam. Our work along with the findings of [Wang et al. 2007] can be used by search engines to help in degrading the web spam URL ranking in the search results.

Typo-squatting is studied in [Wang et al. 2006; Banerjee et al. 2008; McAfee 2007]. These studies reach the conclusion that many typo-squatting domains are registered and exist in the Web. Unlike our study, these studies do not show how often typo-squatting domains are accessed and how much of the ads-portal domains are typo-squatting. Wang et al. [Wang et al. 2006] show that a large number of typo-squatting domains exist and a large number of these typo-squatting domains are parked with few parking services that serve ads on these domains. The authors identify parked domains by checking if the third-party URL refers to a parking service, essentially similar to the signature-based identification we use in our data set collection in Section 3. But for us, we identify ads-portal domains through a machine learning classifier that enables us to detect more parked domains regardless of the parking service. Wang et al. [2006] implement a tool called “Strider URL Tracer” that displays the third-party URLs and helps the trademark owners to check if there are typo-squatting domains of their domains by automatically generating and scanning typo domains. However, it does not automatically detect which of the generated and scanned typo domains are typo-squatting domains. A manual examination is needed.

Banerjee et al. [2008] study the extent of typo-squatting. For 900 well-known domains, they generate around 3 millions similar URL variations and then investigate to see which of them are phony/typo-squatting domains. They find that typo-squatting domains exist at a large extent. Also, they find that most of the typo-squatting domains are of one character variation of the original target domains. McAfee [2007] studies the prevalence of typo-squatting. For 2,771 target domains, 1.9 million different single error typos have been generated. In the typo set, 127,381 suspected typo-squatting domains have been identified. Unlike our machine learning way of identifying parked domains, McAfee uses the existence of a parking service signature (URLs, pieces of text) in the content of the domain as a way to identify typo-squatting domains. Also, McAfee has equipped its extension site advisor [McAfee 2008] with the capabilities of identifying typo-squatting domains. The extension will show a yellow color if the site is a typo-squatting site with no risk. If the site is risky, a red color will be shown.

## 9. CONCLUSION

A text-based ads-portal domain refers to a web domain that shows advertisements in the form of ads listing and no real content. The ads content in an ads-portal domain is served by a third-party advertisement syncation service. Ads-portal domains are useful in showing related ads content to users performing direct search. However, ads-portal domains are misused in at least two ways: typo-squatting and web spamming.

In this paper, we develop a machine-learning-based classifier to identify ads-portal domains. The features of the classifier are extracted from the web content of the domain. In developing the classifier, we first create negative and positive samples set. Then, we identify a set of features that are effective in distinguishing ads-portal domains from other domains. Finally, we combine these features along with other keyword-based features into a machine learning classifier. The resulting classifier has 96% accuracy in identifying ads-portal domains. Our identification methodology represents a step towards better mining and categorizing the web domains. Also, it can be helpful to search engines ranking algorithms, helpful in identifying web spams that redirects to ads-portal domains, and used to discourage access to typo-squatting domains.

We use this classifier along with the Internet zone files for *.com* and *.net* to measure the prevalence of ads-portal domains in the Internet and to find the ratio of ads-portal domains that are typo-squatting. We find that 28.3/25% of (two-level) *\*.com/\*.net* web domains are ads-portal domains and 41/39.8% of the ads-portal domains in *\*.com/\*.net* are typos. These numbers show the prevalence of both ads-portal and typo-squatting domains in the Internet. In addition, we use the classifier along with DNS trace files to estimate how often Internet users visit ads-portal domains and typo ads-portal domains. It turns out that  $\sim 5\%$  of the (two-level) *\*.com, \*.net, \*.org, \*.biz and \*.info* web domains found in the trace files are ads-portal domains and  $\sim 50\%$  of these ads-portal domains are typos. These numbers show that ads-portal domains and typo-squatting domains are successful, with respect to our traces, in attracting many visits.

## REFERENCES

- BANERJEE, A., BARMAN, D., FALOUTSOS, M., AND BHUYAN, L. N. 2008. Cyber-Fraud is One Typo Away. In *Infocom 2008 mini-conference*.
- BREIMAN, L. 1996. Bagging predictors. *Machine Learning* 24, 2, 123–140.
- BREIMAN, L. 2001. Random Forests. *Machine Learning* 45, 1, 5–32.
- COHEN, W. 1995. Fast Effective Rule Induction. In *Proceedings of the 12th International Conference on Machine Learning*.
- DRUCKER, H., VAPNIK, V., AND WU, D. 1999. Support vector machines for spam categorization. *IEEE Transactions on Neural Networks* 10, 5, 1048–1054.
- ESFANDIARI, B. AND NOCK, R. 2005. Adaptive Filtering of Advertisements on Web Pages. In *Proceedings of International World Wide Web Conference (WWW)*.
- F-SECURE. 2005. Google.com installed malware by exploiting browser vulnerabilities. <http://www.f-secure.com/v-descs/google.shtml>.
- FIELDING, R., GETTYS, J., MOGUL, J., FRYSTYK, H., MASINTER, L., LEACH, P., AND BERNERS-LEE, T. 1999. *Hypertext Transfer Protocol – HTTP/1.1*. RFC 2616.
- FREUND, Y. AND SCHAPIRE, R. 1995. A decision-theoretic generalization of on-line learning and an application to boosting. In *European Conference on Computational Learning Theory*.
- GOOGLE. 2006. Google SOAP Search API. <http://code.google.com/apis/soapsearch/>.
- GOOGLE. 2008. Google adsense. <http://www.google.com/adsense>.
- GUSFIELD, D. 1998. *Algorithms on Strings, Trees and Sequences*. Cambridge University Press.

- IBA, W. AND LANGLEY, P. 1992. Induction of one-level decision trees. In *Proceedings of the 9th International Conference on Machine Learning*.
- JOACHIMS, T. 2001. A statistical learning model of text classification with support vector machines. In *Proceedings of the 24th ACM International Conference on Research and Development in Information Retrieval*.
- KAWAKITA, M., MINAMI, M., EGUCHI, S., AND LENNERT-CODY, C. E. 2005. An introduction to the predictive technique AdaBoost with a comparison to generalized additive models. In *Fisheries research*.
- KOHAVI, R. 1995. The power of decision tables. In *Proceedings of the European Conference on Machine Learning*.
- KUSHMERICK, N. 1999. Learning to remove Internet advertisements. In *The 3rd International Conference on Autonomous Agents*.
- MCAFEE. 2007. McAfee's Study of Typosquatting. [www.mcafee.com/typosquatters](http://www.mcafee.com/typosquatters).
- MCAFEE. 2008. McAfee SiteAdvisor. <http://www.siteadvisor.com/>.
- MITCHELL, T. 1997. *Machine Learning*. McGraw Hill.
- MOCKAPETRIS, P. 1987. *DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION*. RFC 1035.
- MOZDEV. 2008. Adblock. <http://adblock.mozdev.org/>.
- N. CRISTIANINI AND J. SHAWE-TAYLOR. 2000. *An introduction to support Vector Machines: and other kernel-based learning methods*. Cambridge University Press, New York, NY, USA.
- NTOULAS, A., NAJORK, M., MANASSE, M., AND FETTERLY, D. 2006. Detecting Spam Web Pages through Content Analysis. In *Proceedings of International World Wide Web Conference (WWW)*.
- PORTER, M. 1980. An algorithm for suffix stripping. *Program 14*, 3, 130–137.
- QUINLAN, J. 1993. *c4.5: Programs for Machine Learning*. Morgan Kaufmann.
- QUINLAN, J. R. 1996. Bagging, boosting, and c4.5. In *13th National Conference on Artificial Intelligence and 8th Innovative Applications of Artificial Intelligence Conference*.
- RAGGETT, D., HORS, A. L., AND JACOBS, I. 1998. *HTML 4.0 Specification*. <http://www.w3.org/TR/1998/REC-html40-19980424>.
- WANG, Y.-M., BECK, D., WANG, J., VERBOWSKI, C., AND DANIELS, B. 2006. Strider Typo-Patrol: Discovery and Analysis of Systematic Typo-Squatting. In *Proceedings Usenix SRUTI Workshop*.
- WANG, Y.-M., MA, M., NIU, Y., AND CHEN, H. 2007. Spam Double-Funnel: Connecting Web Spammers with Advertisers. In *Proceedings of International World Wide Web Conference (WWW)*.
- WIKIPEDIA. 2008. Type-in traffic. [http://en.wikipedia.org/wiki/Type-in\\_traffic](http://en.wikipedia.org/wiki/Type-in_traffic).
- WITTEN, I. H. AND FRANK, E. 2005. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.
- YAHOO. 2007. Yahoo! directory. <http://dir.yahoo.com/>.
- YAHOO. 2008. Yahoo Search Web Services. <http://developer.yahoo.com/search/web/V1/spellingSuggestion.html>.