2019 IEEE Symposium on Security and Privacy

Measuring and Analyzing Search Engine Poisoning of Linguistic Collisions

Matthew Joslin*, Neng Li†, Shuang Hao*, Minhui Xue‡, Haojin Zhu†
*University of Texas at Dallas †Shanghai Jiao Tong University ‡Macquarie University
{matthew.joslin, shao}@utdallas.edu {ln-fjpt, zhu-hj}@sjtu.edu.cn minhuixue@gmail.com

***Abstract*—Misspelled keywords have become an appealing target in search poisoning, since they are less competitive to promote than the correct queries and account for a considerable amount of search traffic. Search engines have adopted several countermeasure strategies, e.g., Google applies automated corrections on queried keywords and returns search results of the corrected versions directly. However, a sophisticated class of attack, which we term as *linguistic-collision misspelling*, can evade auto-correction and poison search results. Cybercriminals target special queries where the misspelled terms are existent words, even in other languages (e.g., "idobe", a misspelling of the English word "adobe", is a legitimate word in the Nigerian language).**

In this paper, we perform the first large-scale analysis on linguistic-collision search poisoning attacks. In particular, we check 1.77 million misspelled search terms on Google and Baidu and analyze both English and Chinese languages, which are the top two languages used by Internet users [1]. We leverage edit distance operations and linguistic properties to generate misspelling candidates. To more efficiently identify linguistic-collision search terms, we design a deep learning model that can improve collection rate by 2.84x compared to random sampling. Our results show that the abuse is prevalent: around 1.19% of linguistic-collision search terms on Google and Baidu have results on the first page directing to malicious websites. We also find that cybercriminals mainly target categories of gambling, drugs, and adult content. Mobile-device users disproportionately search for misspelled keywords, presumably due to small screen for input. Our work highlights this new class of search engine poisoning and provides insights to help mitigate the threat.

I. Introduction

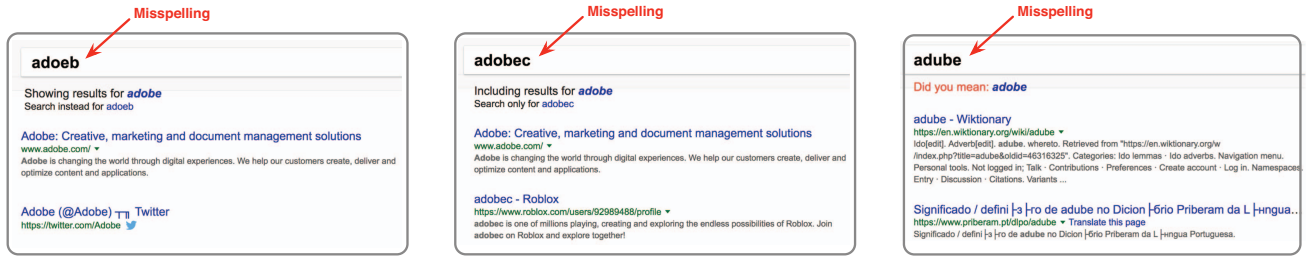
Search engines serve an important role in people's daily lives and drive the majority of web traffic. Indeed, 50%–70% of the traffic to websites come through search engines [2]. Website developers and administrators go to great lengths to improve the rankings of their pages by following benign search engine optimization (SEO) guides [3]. On the other hand, cybercriminals attempt to use search engine poisoning techniques (such as keyword stuffing [4] and link farms [5]) to poison popular search keywords, falsely promote rankings, and divert users to their websites for malicious purposes. Such abuses not only deteriorate users' experience to navigate web content, but also cause substantial loss of visitors and revenue from legitimate businesses.

Misspelled keywords have increasingly become the target in SEO attacks [6], since they are less competitive to poison compared to the correct popular queries and can capture large numbers of users who accidentally make typographical errors. To combat the hassle of abusing misspelled keywords, search engines, including Google and Baidu, have taken multiple actions, ranging from displaying warning messages to bring users' attention when there are potential misspellings in the search queries, to automatically returning search results of the correct versions. As shown in Figure 1(a), suppose a user makes a misspelled search for `adoeb` on Google (misspelling of `adobe`). The search is automatically changed to `adobe` (the correct search term) and the user will not receive any search result for the misspelled input. However, adversaries crave to continue preying on the misspelled query traffic that users generate. Even large vendors attempt to leverage misspelled keywords. For example, Amazon used misspellings to advertise products on their website [7], and Snickers targeted misspelled keywords in the "You are Not You When You're Hungry" advertisement campaign [8]. The rapid adoption of mobile devices, such as smart phones and tablets, exacerbates chances of incorrect inputs, presumably due to typing on small screens. A recent report shows that around 60% of search queries are attributed to mobile devices [9].

To bypass automated corrections of search engines, attackers can employ a new attack scheme, namely *linguistic-collision misspelling*, which abuses the mistyped search queries coinciding with legitimate existent words, even in a different language. For example, "idobe" is a misspelling of the English word "adobe", but also happens to be an existent Nigerian word (meaning "dropping"); "平锅" in Chinese (meaning "frying pan") is a mistake input of "苹果" (meaning "Apple" company). Search engines do not enforce automated corrections on such cases, which introduces exploitation opportunities for cybercriminals to launch search engine poisoning attacks.

In this work, we perform the first large-scale analysis of linguistic-collision search engine poisoning. We focus on both English and Chinese languages, which are the top two languages used by Internet users [1]. We collect target keywords from a variety of categories, such as drugs, gambling, clothing, and food. We also include Alexa top 10,000 names in the English target-keyword corpus. Two main challenges that we face are: (1) how to generate misspelled words, and (2) how to effectively determine whether a particular search term will be auto-corrected/suggested by search engines. For English-word analysis, we first use edit distances to generate potential misspelling candidates. To make the experiment scale (particularly for Alexa top 10,000 names), we adapt a deep learning model–the Recurrent Neural Network framework–to predict how likely a misspelling candidate will not be

© 2019, Matthew Joslin. Under license to IEEE.
DOI 10.1109/SP.2019.00025



(a) Showing-results-for case (high confidence about misspellings), where the returned search results are automatically changed for the corrected search term `adobe`. Users do not receive search results for the misspelled keyword.

(b) Including-results-for case (medium confidence about misspellings), where the top returned results are changed for the corrected search term `adobe` and the rest of the results are for the originally input term.

(c) Did-you-mean case (low confidence about misspellings), where the returned search results are for the misspelled keyword. Meanwhile, users are displayed with a highlighted warning banner to indicate the corrected term.

Figure 1: Examples of Google's auto-correction and auto-suggestion mechanisms on searches with misspelled keywords (original target keyword is `adobe`). Users receive various notifications or corrected results for the misspelled searches.

automatically corrected. Our approach can improve the collection rate by 2.84 times compared to random sampling. For Chinese-word analysis, we use a phonetic approach (pinyin input) to convert Chinese characters to Roman letters and generate misspelling candidates. To reduce online checking, we compare the candidate words against Chinese word dictionaries, since a misspelled Chinese word must still be another valid Chinese word. Finally, we crawl search results showing on the first page from Google and Baidu, and check whether the URLs are blacklisted.

In this work, we have the following key findings.

- We find that linguistic-collision misspellings are widely abused by attackers with 1.19% of non-auto-corrected terms returning malicious results on the first page from both Google and Baidu.
- Cybercriminals primarily target keywords related to drugs, gambling, and adult terms, with searches poisoned at four times the rate of less easily monetized categories (like clothing or food).
- Poisoning activity exhibits a long-tail effect with search results across the Alexa top 10,000 dataset containing around 0.54% poisoning rate on the first page.
- Among various misspelling generation methods, vowel substitution for English produces a 50% higher non-auto-corrected rate compared to average, and the Chinese methods yield a 2.4x improvement for same pronunciation and 2.3x for fuzzy pinyin.
- According to the traffic comparison from Google Adwords and Baidu Index, mobile-device users provide a significant proportion of the traffic to linguistic-collision misspellings presumably through fat-finger errors. The increase in traffic further incentivizes attackers to target this class of search engine poisoning.

To summarize, we make the following contributions in this paper.

- We systematically measure and understand a new threat—linguistic-collision misspellings, which allows attackers to bypass existing auto-correction tools and poison large numbers of search results.
- We design a novel approach using deep learning to collect linguistic-collision misspellings in the wild. Based on our experiment on the Alexa top 10,000 case, we find that our model outperforms random sampling by 2.84x.
- Using our crawling framework, we perform the first large-scale study of linguistic-collision misspellings collecting 1.77 million search results for misspellings generated for 18,234 original keywords across English and Chinese.
- Our results show that linguistic-collision misspellings are widely abused on both Google and Baidu, with around 1.19% results on the first search page directing to malicious websites. We further perform detailed characterization of this class of search poisoning, including the poisoned word categories, effectiveness of misspelling generation approaches, and search volume distribution.

II. Background

A. Chinese Pinyin and Input Approach

Hanyu Pinyin (abbreviated as pinyin) is the phonetic system to represent Chinese characters with Roman letters. Pinyin provides a convenient way to learn Chinese and input Chinese characters on computers. For example, the Chinese character "果" can be encoded as the pinyin symbol `Guo`. Typically each Chinese character is mapped to one pinyin (though there are polyphonic Chinese characters), but one pinyin can represent many different Chinese characters. This can introduce ambiguity when transforming pinyin to Chinese characters. Moreover, pronunciations of pinyin have four tones, which can be indicated by a number following the pinyin. The aforementioned Chinese character "果" (meaning "fruit") maps to pinyin with the third tone `Guo3`. Another Chinese character "锅" (meaning "pan") has the same pinyin spelling but a different tone `Guo1`.

Pinyin input method is the most widely used Chinese-input approach [10] (compared to other input methods, like stroke-based input method). Since the input is based on pronunciations,

it is easy for Chinese speakers to master. Any English keyboard can type pinyin. After users type pinyin of a Chinese character, the input method will display a list of characters corresponded to that pinyin for users to select and use. For convenience, pinyin input system typically does not provide selection of tone marks. The presented possible Chinese characters match the same pinyin spelling and do not distinguish tones. For example, the above "果" and "锅" will be shown simultaneously, once a user types the pinyin `Guo` (since they have the same pinyin spelling).

B. Deep Learning and Recurrent Neural Networks

Deep learning has been applied to a wide range of problems as computing power has grown significantly. Neural networks in particular have seen incredible successes in many application domains. A neural network contains layers of neurons, which provide the computation elements to predict future outputs. The parameters of the neurons provide the memory and are adjusted during training.

In this paper, we focus on a particular type of neural network, the Recurrent Neural Network (RNN), which has been shown to work well with sequential data [11, 12]. An RNN accepts an input sequence of vectors and outputs a vector sequence. The input and output symbols are generally converted to a one-hot representation that allows the model to more easily learn the relationships between the input and the output. The output vectors encode the RNN's estimate of the probability that a given symbol should be selected in the output sequence. During training, the correlation between input and output sequences is learned using Long Short-Term Memory (LSTM) [13]. For text input, RNNs are typically used to deal with text at the word level and have proven remarkably successful in generating text. However, character-based RNNs deal with text at the alphabet level and thus can be more robust when dealing with extremely large vocabularies that may be difficult to collect.

III. Search Engine Poisoning of Misspelled Keywords

Misspelled keywords have been extensively exploited to illicitly seize search traffic and gain profit [6, 8]. Recent reports show that 10%–20% of queries on search engines contain misspellings [14, 15]. These alternative keywords are typically less expensive to purchase or less competitive to promote in the search results, making misspellings attractive targets for cybercriminals.

To counteract misspelling abuse and improve users' experience, over the past several years, major search engines, such as Google and Baidu, have taken significant strategy changes to provide auto-suggestion or auto-correction [16, 17]. We use search results from Google to illustrate different levels of correction that search engines offer when a spelling mistake is detected. As an example, for a original keyword `adobe`, misspelled variants result in the following four search return types from Google (sorted from high to low regarding mitigation against misspellings in queries).

Figure 2: Search results of misspelling `cilis` on Google (original target search word is `cialis`). Top results lead to illicit pharmaceutical websites. Our investigation shows that some of these websites are reported at blacklists and they have cloaking or redirection.

1) `Showing-results-for` (high confidence about misspellings). When search engines have high confidence in what the correct keyword should be, results for the corrected term are directly returned. This is the strongest-level mitigation against misspellings in queries, where the results of the suspect misspelled keyword will not be shown at all. Users are notified that search has been modified with the sign "Showing results for". As shown in Figure 1(a), search for `adoeb` (transposition of `b` and `e`) will return all results for `adobe` instead. Users still have the option to modify to search for the previous query by explicitly clicking `adoeb` in the notification "Search instead for".
2) `Including-results-for` (medium confidence about misspellings). If the spelling mistakes are less evident, search engines may include results for the assumed correct keyword as the top results with notification "Including results for". The rest of the returned results are still for the misspelled keyword. The motive is that users are more likely to click on the results of the corrected keyword (which show as the top results). As shown in Figure 1(b), search for `adobec` (appending letter `c`) has the first result of `adobe` and the rest results for `adobec`. By clicking the suggested word `adobe` in "Including result for" or the original misspelled input `adobec` in "Search only for", users can refine which word they indeed hope to search for.

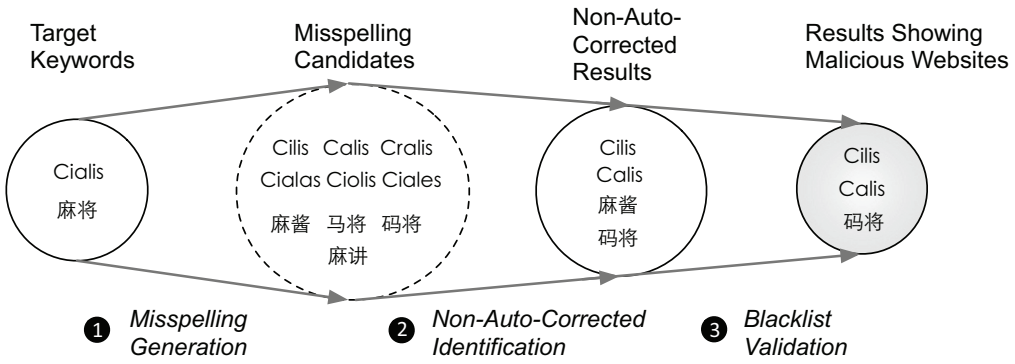


Figure 3: Workflow of finding linguistic-collision keywords for search engine poisoning. Based on a set of selected target keywords, we design algorithms to generate potential misspelling candidates (❶), expanding to a larger word set. Then we reduce the candidate sets to identify the linguistic-collision keywords (❷) and collect the corresponding `non-auto-corrected` results from search engines. Last we check on blacklists to find linguistic-collision keywords associated with malicious websites with high rankings in search results for subsequent analysis (❸).

3) `Did-you-mean` (low confidence about misspellings). When search engines suspect the spelling may contain errors, a warning banner of "Did you mean" with a suggested keyword is displayed to users. However, users receive only search results for the misspelled keyword. Though the notification banner can blend in with search results and be ignored, it raises the chances for users to realize misspellings in the queries and correct them. As shown in Figure 1(c), search for `adube` (misspelling of `adobe` by replacing letter `o` with `u`) on Google leads to search results based on the misspelling. If users click on the suggested query `adobe` in "Did you mean", the search will be re-run for the revised version `adobe` and the warning message will disappear.
4) `Non-auto-corrected` (no detection of misspellings). If search engines have no suspicion of misspellings in the search terms, the query is performed for the keyword that users originally submit. In particular, if a misspelling is coincidentally an existent word, even possibly in a different language, search engines will not modify the original query or display any notification to users. The semantic gap is that search engines have no prior knowledge about the original keywords that users intend to search. For example, search for `idobe` (replacing the first letter `a` with `i`) yields regular search results for the word. The page will show no special notification or hint about potential misspellings. In fact, the word `idobe` (misspelling of `adobe`) is an existent word in a Nigerian language, meaning "dropping".

For the first three cases, users receive notifications or corrected search results automatically, which diminishes chances of attackers to manipulate and monetize the search results of misspellings. However, for the `non-auto-corrected` case, mistyped search queries coincide with legitimate existent words and users receive results of the misspelled input. Therefore, it is more likely that users cannot realize that they make query misspellings and are tricked into clicking on the returned results. Such misspelled keywords remain susceptible to search poisoning attacks, which we coin as linguistic-collision misspellings. In this paper, we focus on the `non-auto-corrected` cases and conduct the first large-scale empirical analysis to characterize linguistic-collision SEO attacks.

Pharmaceutical examples of linguistic-collision SEO. Promoting illicit pharmacy websites is a major target of cybercriminals [18]. We illustrate the scheme with a search on `cilis`, a misspelling of the pharmaceutical drug `cialis` (missing one letter `a` in the middle). The misspelled variant exists in the language of Esperanto and means "chilis". Figure 2 shows the Google search results. We note that obviously the top search results contain links to pharmacy websites. In particular, there are three interesting observations. (1) The paid ads on the top refers to a website selling pharmaceutical drugs. Vendors intentionally purchase misspelled keywords for advertising on search engines to gain traffic and profit. (2) The first returned result is a website under `terrypaulson.com`, flagged as malicious by VirusTotal [19]. The website deploys cloaking mechanisms to hide the true intention. If users directly visit the URL, the website shows a page full of text. If users click through the Google search result, the website turns to make online pharmacy sales (as shown in Figure 2). (3) The third search result shows a URL under `oversand.es`. Clicking the link will follow redirection to reach a website `online-pharmacyrx-canada.com`, which sells illicit drugs. The entry page is hosted at Spain, while the landing page locates at Lithuania. The above findings show that through linguistic-collision SEO, it is comparatively easier for cybercriminals to achieve high rankings on search engines and evade filtering from authorities.

Another interesting example of linguistic-collision SEO is `clalis` (replacing the first `i` with `l` in `cialis`), which does not trigger auto-correction on Google search. Similarly, the returned results have a purchased ads linking to an online pharmacy website `goodrx.com`. Moreover, U.S. Food & Drug Administration (FDA) has advised consumers not to fall victim to clalis scams [20] (which is not cialis). Abuse of linguistic-collision keywords causes negative impact to users and degrades the results' quality for search engines.

IV. Methodology

In this section, we describe how we generate linguistic-collision misspellings and establish ground truth data. We select English and Chinese as our analyzed languages, since they are the top two languages used by Internet users [1]. The experiments are performed for Google and Baidu respectively, which represent the largest search engine market share [21]. Figure 3 outlines the overall design of our methodology. The workflow applies to both the English and Chinese experiments. The circles represent the data sets that we generate during the process. The descriptions about the data are shown above each circle, and in the circles we show word examples. In Figure 3, the English word example is `cialis`, referring to a classic pharmaceutical drug. The Chinese word example is "麻将" (Pinyin as `Ma2Jiang4`), meaning a traditional Chinese gambling game. The sizes of the circles simulate whether the data size will increase or shrink compared to the data at the previous step. In Section VI, we investigate details of the change ratios of data sizes along the process.

The process has three main steps. Given a set of target keywords, we develop mechanisms to transform them into misspelling candidates (❶). Note that the generated candidates are not necessarily linguistic-collision misspellings, and may cause auto-suggestion/correction on search engines. Typically one target keyword will correspond to multiple misspelling candidates, therefore the dataset at this step will expand considerably. Next we filter to obtain the candidates that produce non-auto-corrected search results (❷), which will shrink the keyword set. We collect the search results and the corresponding URLs showing on the first search page, typically around 10 results. Previous studies show that 70%–90% of user clicks happen at the first page of search results [22, 23]. We then examine whether the URLs of the first-page search results are flagged as malicious by public blacklists (❸). Correspondingly, we discern which misspelled keywords are abused for search poisoning attacks and further characterize various facets of the attacks.

A. English-language Design

Since English and Chinese languages have distinct lingual properties, we use different design strategies, in particular for the first two steps. We introduce our design of English language for misspelling generation and non-auto-corrected identification.

Misspelling generation (❶). To generate misspellings from the English keywords, we use a modified version of the Damerau-Levenshtein edit operations [24, 25]. The Damerau-Levenshtein edit operations can (1) insert a character, (2) replace a character, (3) transpose two adjacent characters, or (4) delete a character. To restrict the number of the generated candidates, we use the approach proposed by Moore and Edelman [26], which limits the character replacement operation to characters that are adjacent to the original key on a QWERTY keyboard (i.e., fat-finger errors). In addition, we allow replacement of any English alphabet vowels, including letters `a`, `e`, `i`, `o`, `u` and `y`. We focus on edit distances with one, as previous work has suggested that the Damerau-Levenshtein edit operations with distance one contain about 80% of all single mistake misspellings [24].

Non-auto-corrected identification (❷). We first introduce two straw-man approaches to identify linguistic-collision words for English misspellings. (1) Mapping to explicit vocabulary in dictionaries. The approach has two main limitations. One is that linguistic-collision misspellings may be legitimate words in non-English languages, which requires to include numerous multi-language dictionaries. Another issue is that users keep inventing plausible words to describe new phenomena. For instance, "Linsanity" follows most English spelling rules, but was not in popular use until 2012. As we will show in Section V-B, strict dictionary checking results in poor coverage of confirmed linguistic-collision misspellings. (2) Brute-force checking on search engines. The approach is to perform online checking for all misspelling candidates on search engines. For a selected set of keywords (Alexa top 1K and manually selected categories), we conduct exhaustive checking to obtain comprehensive analysis (see Section V). However, the approach cannot scale for large-scale experiments (Alexa top 10K). For example, enumerating all possible insertions (one of the Damerau-Levenshtein edit operations) requires performing 26 queries per input character. Such a high-level of overhead cannot be supported for web-scale datasets, and we need to develop a method for eliminating auto-corrected candidates more efficiently.

We adapt a Recurrent Neural Network (RNN) framework to estimate how likely a word will not be auto-corrected by search engines. RNNs have been widely applied to natural language processing (as described in Section II) and used to predict sequential text outputs. Our primary insight is that a formally recognized word should display character-level patterns similar to the rest of dictionary vocabulary for users to adopt it. RNNs can generate high-quality language models for character-level representations [27, 28]. Our developed approach effectively addresses the challenges of recognizing new words (not covered in dictionaries) and linguistic-collision words in non-English languages.

Figure 4 demonstrates our framework for training an adapted RNN and generating confidence estimates on misspelling candidates. The system consists of two phases, training phase and prediction phase. (1) In the training phase, we adapt to train with individual words from dictionaries. We use dictionaries to learn from a large corpus of words and capture the general English lexical patterns. We append a null character to the beginning and end of the word to allow the RNN to learn about word boundaries. With the popular Tensorflow library [29], we train a character-based RNN to recognize the typical structure of legitimate words. After randomly initializing the model weights, we use the Adam optimization algorithm [30] with gradient clipping to reduce the cross-entropy during training. (2) In the prediction phase, our goal is not to generate arbitrary text content, but to predict whether particular misspellings that we have generated will not be auto-corrected by search engines (i.e., coincidentally legitimate words). Given an input prefix $\vec{x}$ (e.g., `goog` in Figure 4), an RNN outputs a probability distribution $\vec{p}$ for the alphabet on which character is most likely (in the example

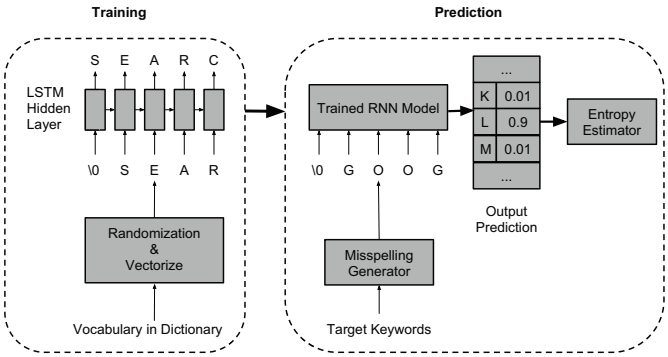


Figure 4: RNN framework to predict how likely misspelling candidates for English original keywords will cause non-auto-corrected results on search engines.

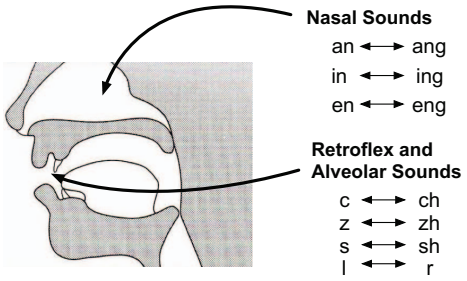


Figure 5: Fuzzy pinyin and anatomical parts to produce the sounds. We include pinyin strings that are easy to confuse with each other.

letter `l` has the highest probability). We adapt to calculate the average entropy of the RNN's prediction over each output character. Suppose the candidate word has n letters, the size of the character set is l, and the distribution output of the RNN at letter position k ($1 \leq k \leq n$) is $\vec{p}_k = (p_{k1}, p_{k2}, \ldots, p_{kl})$. The entropy at the position k is $H(\vec{p}_k) = \sum_{i=1}^{l} p_{ki} \log_2(p_{ki})$. The average entropy for a given prediction can be calculated as $\sum_{j=1}^{n} H(\vec{p}_j)/n$. Intuitively, the average entropy is a normalized estimate of the RNN's confidence that the misspelling could plausibly be used as an existent word. Low entropy values indicate misspellings which should be more likely to be non-corrected.

B. Chinese-language Design

The linguistic properties of Chinese words require different strategies to generate misspelling candidates and identify non-auto-corrected search keywords.

Misspelling generation (❶). For each target keyword, we first convert the Chinese characters into pinyin, which is composed of English letters. Then we apply same edit distance operations (as for English misspelling generation) to spawn new pinyin strings. According to pinyin's lexical rules, some generated pinyin strings may not be valid (we still count them as candidates to match existent pinyin). We transform pinyin strings to all possible Chinese characters with that pronunciation. In particular, there exist two phenomena. (1) Same pinyin. As introduced in Section II, many different Chinese characters map to the same pinyin. When we transform back from pinyin to Chinese characters, the number will increase considerably. Different tones further exaggerate the phenomenon, given that most pinyin input methods do not provide tone selection to users. (2) Fuzzy pinyin. Some pinyin have close pronunciations, including nasal, retroflex, and alveolar sounds. Figure 5 shows the anatomical parts to make the pronunciations and the confusing pinyin strings. Many people cannot distinguish the differences. Pinyin input methods also automatically include Chinese characters that match fuzzy pinyin for users to select. More analysis on misspelling generation comparison will be shown in Section VI.

Non-auto-corrected identification (❷). In contrast to the English case, linguistic-collision Chinese words will still be Chinese words. Therefore, we directly check whether a misspelling candidate exists in Chinese dictionaries. For valid Chinese words, search engines will not apply auto-correction/suggestion. As the examples in Figure 3 demonstrate, even if all Chinese characters are valid, the combination may not form meaningful Chinese words. The identification procedure can be performed offline. We collect commonly used Chinese words from four popular word dictionaries of Sogou pinyin input method [31]. In total, the dataset contains 1,166,765 Chinese words.

C. Crawling Tasks

To perform the experiment at a large enough scale, we designed a framework to collect search results, search volumes, translation data, and blacklist information. Figure 6 gives a high-level view of these tasks and how they relate to each other. We begin by collecting the search results for input keywords, and then check the search volumes, Google Translate API, and blacklist for search terms. Together, these datasets provide a comprehensive view of linguistic-collision misspellings. To ensure that the search engine servers would not be overloaded, we rate-limited our crawlers.

1) Search results. To determine whether or not the search results were auto-corrected, we checked the returned page for the notices described in Section III. If the keyword was not corrected by the search provider, we parsed the search result page and collected the first 10 search result entries in a database for later analysis. In particular, we saved the title, description, and URL for each entry. We used the URL to check if the result was blacklisted and the title and description proved invaluable to understanding the SEO techniques used with linguistic-collision misspellings. In addition, we captured the estimated number of search results to understand how difficult the SEO is for particular keywords. Because the search results can change quickly for pages with malicious entries, we also captured the raw HTML to allow for later manual inspection.
2) Search volumes. To analyze how users are exposed to non-auto-corrected misspellings we queried Baidu Index [32] and Google Adwords [33]. To estimate search volume for Chinese terms, we used Baidu Index to collect daily search volumes for the previous week and month. While Baidu Index allows

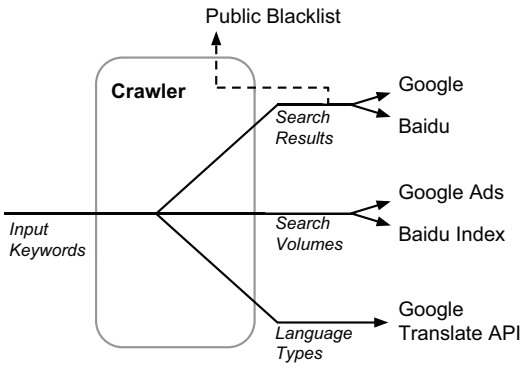


Figure 6: Crawling framework that contains four tasks, collecting search results, search volumes, language types, and public blacklist.

users free access to search volumes, Google Adwords has recently restricted search volume data to paid customers. As a result, we only use Google Adwords data to investigate questions that only require comparing the predictions, such as from what types of devices users are searching. Using relative Google Adwords data allows us to compare mobile and desktop searches, but not exact volumes for large lists of words.

3) Language types. Because we are interested in what percentage of English linguistic-collision misspellings are coexistent within the same language vs. other languages, we decided to use the Google Translate API to detect the language of the misspellings [34]. Knowing the language of a misspelling allows us to determine whether the misspelling is between two languages or within the same language. In addition to returning the detected language, the Google Translate API returns a confidence score which allows us to understand why Google would fail to correct the misspelling.
4) Public blacklist. Finally, we scanned all of the URLs returned for the uncorrected misspellings found during task 1). To determine whether a URL is malicious, we checked VirusTotal [19]. VirusTotal currently aggregates 68 antivirus scanning engines to identify malicious URLs, including Google Safebrowsing [35], Yandex Safebrowsing [36], Spamhaus [37], and Baidu-International [38]. To avoid introducing high false positive rates, we also implemented manual spot checking to ensure that the accuracy remained high.

V. Experiment

In this section, we describe our experiment settings, keyword selection, and statistics of the collected data. We also demonstrate the performance of the adapted RNN approach to generate eligible search keywords (i.e., those that are not auto-corrected by search engines).

A. Data Collection and Validation

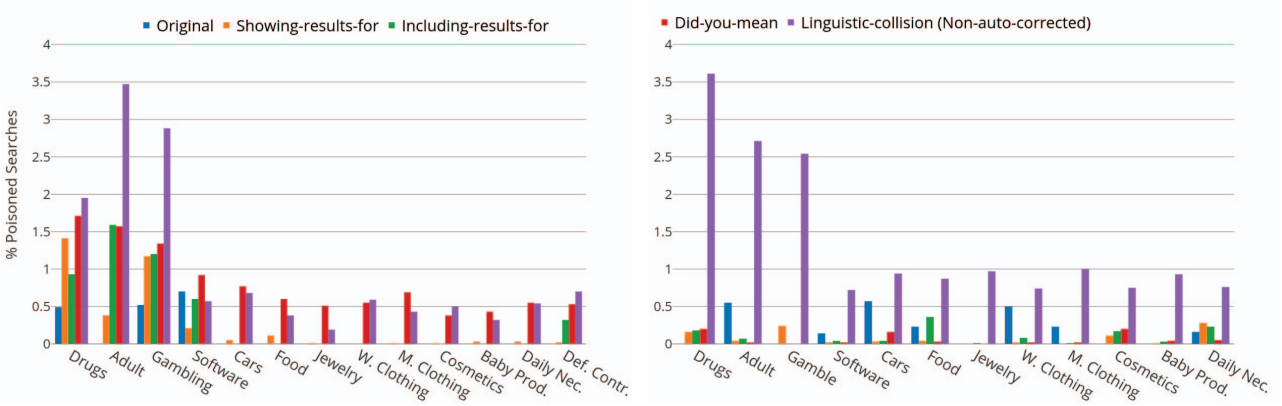
To understand the characteristics of linguistic-collision misspelling SEO, we perform a large scale data collection and analysis. We ran the experiment on a cluster of 26 servers with 2 CPUs and 4 GB of RAM from December 2017 to July 2018. Specifically, we conducted two parallel studies targeting Chinese and English terms. We follow the approach in Section IV to generate candidate keywords and fetch search results from Google and Baidu respectively. For the English study, we generated misspellings from 11,520 original keywords and collected 1,044,711 searches using the Google search service. For the Chinese study, we generated misspellings from 6,714 original keywords and collected 724,865 searches from Baidu. We use two strategies to select original target keywords: (1) manually collected categories, and (2) Alexa list of popular websites, for which we will describe details below.

Keyword collection per category. Miscreants intend to target specific sets of keywords to gain illicit profit, so we manually select 13 different categories in English and 12 different categories in Chinese for analysis. Previous work indicates that cybercriminals target more on prescription drugs, gambling terms, adult terms, and software categories [18, 39] (results in Section VI confirm the conjecture). We collect terms in such categories for analysis. We also include general consumer product categories, such as food, cards, clothing, cosmetics, and jewelry, to allow for a comprehensive comparison. For English analysis, we collected the terms from the user-ranked forums [40], and other lists curated for specific topics [41–43]. In addition, the discovery of a parked domain using the misspelling of a major US defense company led to the inclusion of defense contractor's names as this type of more targeted misspelling could be used by more sophisticated attackers for phishing. In total, the English per-category keywords contain 1,520 terms, and lead to 563,555 misspelling candidates. For Chinese analysis, we mainly obtain the target keywords from the website `china-10.com`, which contains terms for various categories. We totally collect 6,714 Chinese target keywords, and generate 718,151 misspelling candidates. A detailed breakdown of the per-category statistics is shown in Table I. The first column is the names of the categories, the second column shows the numbers of the collected target keywords of English, and the sixth column shows the counts of the target terms of Chinese. We will describe the other columns of the table in Section VI.

Keyword collection based on Alexa top list. In domain typosquatting attacks, cybercriminals target names of popular websites [44, 45]. Similarly, we include the top names of Alexa domain list [46] in our analysis. Because it is difficult to find a counterpart list for Chinese, we only collected the Alexa top list for English analysis. Table II shows the statistics of Alexa top 100, 1,000, and 10,000 names respectively. The second column represents the numbers of the generated misspelling candidates that we search on Google. For Alexa top 1,000 terms, we use brute-force search results of misspelling candidates for comprehensive analysis and evaluation of RNN performance (Section V-B). To examine the long-tail effect [47], we also consider the Alexa top 10,000 domains, which lead to 2,105,218 misspelling candidates. However, it is inefficient to exhaustively crawl all these keywords. Instead, we deploy the RNN approach that we design in Section IV to identify keywords likely to cause linguistic collision and not to be auto-corrected by Google.

	English				Chinese			
Category	*# Target*	*# Misspell Candidates*	*% Non-Auto-Corrected*	*% Poisoning*	*# Target*	*# Misspell Candidates*	*% Non-Auto-Corrected*	*% Poisoning*
Drugs	205	57,255	4.59% (2.6K)	1.95% (51)	46	3,738	11.85% (443)	3.61% (16)
Adult Terms	214	73,089	37.57% (27.5K)	3.47% (950)	181	32,047	11.41% (3.7K)	2.71% (99)
Gambling	192	79,464	7.33% (5.8K)	2.88% (168)	42	1,951	18.14% (354)	2.54% (9)
Software	288	126,622	6.96% (8.8K)	0.57% (50)	700	84,008	6.29% (5.3K)	0.72% (38)
Cars	68	16,675	11.40% (1.9K)	0.68% (13)	1,767	218,697	4.74% (10.4K)	0.94% (97)
Food	98	43,668	8.49% (3.7K)	0.38% (14)	1,738	159,825	6.62% (10.6K)	0.87% (92)
Jewelry	49	16,613	9.53% (1.6K)	0.19% (3)	148	24,956	6.17% (1.5K)	0.97% (15)
Women's Clothing	43	14,235	8.33% (1.2K)	0.59% (7)	199	25,365	10.18% (2.6K)	0.74% (19)
Men's Clothing	55	18,781	9.99% (1.9K)	0.43% (8)	440	40,903	8.85% (3.6K)	1.00% (36)
Cosmetics	47	17,706	5.72% (1.0K)	0.50% (5)	439	75,844	6.86% (5.2K)	0.75% (39)
Baby Products	46	15,484	14.09% (2.2K)	0.32% (7)	394	51,935	6.62% (3.4K)	0.93% (32)
Daily Necessities	126	42,638	6.10% (2.6K)	0.54% (14)	620	68,176	8.92% (6.1K)	0.76% (46)
Defense Contractors	89	40,984	6.65% (2.7K)	0.70% (19)	—	—	—	—

Table I: Detailed breakdown of per-category collection statistics. "# Target" is the number of original terms used to generate misspellings for that category, "# Misspell Candidates" is the number of generated misspelling variants of the target keywords. "% Non-Auto-Corrected" is calculated as the number of queries for which the search engine does not offer auto-correction either automatically or as a suggestion, and "% Poisoning" is calculated as the percentage of non-auto-corrected queries which contain malicious URLs on the first page of search results. For the "% Non-Auto-Corrected" and "% Poisoning", we also show the raw numbers of searches in parentheses.



(a) English experiment (on Google).

(b) Chinese experiment (on Baidu).

Figure 7: Comparison of search poisoning rates among different misspelling types per keyword category. The y-axis indicates the percentage of searches that contained malicious URLs on the first page of search results (for a given keyword category and misspelling protection type). From left to right for each category, `Original` refers to searches made for the correctly spelled terms, while `Showing-results-for`, `Including-results-for`, `Did-you-mean`, and `Linguistic-collision (Non-auto-corrected)` refer to types of auto-correction offered for the searches as described in Section III. The different categories are described in Section V-A, note that "Defense Contractors" is only present in the English experiment. The search poisoning rates of `Linguistic-collision (Non-auto-corrected)` are the same values as "% Poisoning" columns in Table I.

Auxiliary information collection. In addition to the search results collected from Google and Baidu, we also collected information from VirusTotal, Google Adwords, Google Translate, and Baidu Index. We used VirusTotal to identify URLs with suspicious activity and then investigated further into the flagged results. In total, we collected scans for 2.06M URLs of which 1.18% (24.4k) had been detected by at least one scanner. To improve the accuracy, we manually spot-checked the flagged URLs for malicious activity using a virtual machine which eventually obtained 5,256 malicious URLs under 2,743 domains. For the English search results, we checked the device breakdown estimates for 117,791 uncorrected misspellings and 12,943 original keywords using the Google Adwords Keyword Planner tool [48]. Using the Google Detect Language API we collected 105,978 predictions for the uncorrected misspellings in an attempt to understand the distribution of how the language distribution varies across different categories. The details for our language results can be seen in Table III.

B. Results of RNN

The final model used 150 hidden layers with a sequence length of 5 characters. The vocabulary consisted of lower-case alphanumerics and a null character for a total vocabulary size of 37 characters. To train the RNN model for different parameters, we used 4 servers with 24 GB RAM and 16 CPU cores each. The training set we used was a wordlist with 675,903 unique words taken from several wordlists [49–52]. To select optimal parameters, we checked each setting on completely separate

Category (Alexa Top)	*# Misspell Candidates*	*% Non-Auto-Corrected*	*% Poisoning*
1–100	20,192	16.29% (3.2K)	0.85% (28)
101–1,000	216,157	13.28% (28.7K)	0.78% (221)
(RNN) 1,001–10,000	61,088	38.04% (23.2K)	0.50% (116)

Table II: Data collection statistics based on Alexa top list (similar header meanings as in Table I). Note that the results for the Alexa top 1,001–10,000 are collected using the RNN model's predictions.

validation data taken from the ground truth data on the Alexa top 1,000 misspellings.

To evaluate the RNN's performance and investigate misspellings affecting less popular domains, we used the trained RNN with the best performance on the Alexa 1,000 misspellings to generate predictions for the 2.4 million misspellings from the Alexa 10,000. From these predictions, we selected the keywords with the lowest entropy from the predictions and used the crawling framework to collect search results. The ground truth data collected for the Alexa top 1,000 indicates that randomly sampling the misspellings would yield a hit rate of about 13.28%. Dictionary checking exhibited even lower performance on the Alexa top 1,000 ground truth set with a 2.6% hit rate. The poor performance of dictionary checking vs. random sampling can be explained by the fact that many of the words are new, obscure, or only in use as slang. Our RNN approach also outperforms the naive Bayes and random forest algorithms. Due to space limitation, more details are shown in Appendix A. Crawling the 61,088 highest confidence predictions from the RNN gave a non-auto-corrected rate of 38.04% with 23,236 uncorrected misspellings. Compared to random sampling, the RNN gave a performance improvement of 2.84x.

VI. Measurement and Discoveries

In this section, we present findings from our study, including landscape of the abuses, characteristics of the linguistic-collision misspellings, and estimates of search volumes for cybercriminals. We also provide deep analysis of two interesting cases.

A. Landscape and Comparison of Misspelling Search Results

First, we examine how pervasive the linguistic-collision misspelling SEO is. In fact, we find linguistic collisions are widely existent: 15.16% of the English misspelling keywords that we generate using edit distance 1 are not auto-corrected, and 7.69% of the Chinese misspelling terms based on the fat-finger, fuzzy pinyin, and same pronunciation generation methods are not auto-corrected. Because users primarily click search results returned on the first page [53], we only checked to see whether the first page of search results has been poisoned.

Blacklist statistics. To determine whether or not a URL was potentially malicious, we checked VirusTotal for reports of malicious activity from that URL. In total, we determine that 1,511 URLs from first-page results (10 results per first page) of non-auto-corrected searches are malicious. Correspondingly, 0.98% (1,872) of English linguistic-collision search terms on Google result in first-page blacklisted URLs, and 1.39% (538) of Chinese linguistic-collision terms show poisoned results on the first pages on Baidu. The observation indicates that linguistic-collision misspelling SEO has widespread impact, and cybercriminals can comparatively easily manipulate rankings and promote their pages index by linguistic-collision misspellings.

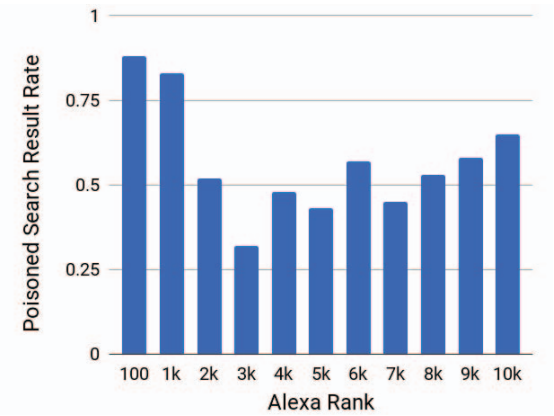


Figure 8: Longitudinal view of the poisoned non-auto-corrected search result rate over Alexa terms (1,001–10,000 using the RNN predictions). The results are binned by the original term's Alexa rank with the x-axis labels denoting the bucket lower and upper bounds, e.g., 2k covers the range of 1,001–2,000.

Per-category results. As mentioned in Section V, the English misspellings were split into two major sets, per-category keywords and Alexa domains. Table I describes the per-category datasets for Chinese and English. The first column shows the category names. We have 13 categories, and 12 of them are present in both Chinese and English ("Defense" category only has keywords in English and contains the names of the 100 largest defense contractors around the world). The fourth and eighth columns "% Non-Auto-Corrected" represent the proportion of misspelling queries not auto-corrected by search engines, regarding English and Chinese respectively. The fifth and last columns "% Poisoning" indicate the percentage of non-auto-corrected queries containing VirusTotal blacklisted URLs on the first-page search results, regarding English and Chinese respectively. We also include raw numbers of searches in parentheses in Table I. There are two observations: (1) A considerable portion of misspellings ($>$ 4.5% for all categories) result in linguistic collisions that will not be auto-corrected by search engines, and (2) many linguistic-collision misspelling searches lead to malicious websites appearing on the first pages of search results.

To compare linguistic-collision misspelling to other types misspelling searches, we queried all misspell candidates that we generated (column "# Misspell Candidates" in Table I) and the original target keywords (column "# Target" in Table I) from the search engines. Figure 7 shows the poisoning rates for English and Chinese by category and level of correction from the search engines. We find that indeed attackers more successfully target linguistic-collision (`Non-auto-corrected`) misspellings than misspellings that are protected by the different types of auto-correction discussed in Section III. On average linguistic-collision misspellings are poisoned at a rate of 1.19% across English and Chinese categories as compared to

All Results		*Alexa top 1K*		*Drugs*		*Software*		*Gambling*		*Adult Terms*	
English	57.44%	English	40.67%	English	49.28%	English	74.04%	English	66.44%	English	81.67%
Arabic	2.76%	Arabic	5.42%	Latin	3.69%	Italian	1.91%	Spanish	2.69%	French	1.96%
Spanish	1.66%	Hindi	2.19%	Spanish	2.82%	Arabic	1.44%	Norwegian	2.14%	Spanish	1.30%
Hindi	1.56%	Welsh	2.18%	Italian	2.47%	Spanish	1.33%	Italian	1.78%	Indonesia	1.05%
Italian	1.53%	Danish	1.68%	Romanian	2.25%	Hindi	1.01%	French	1.68%	Polish	0.79%

Table III: Per-category breakdown of language statistics.

0.16% for `Original`, 0.18% for `Showing-results-for`, 0.23% for `Including-results-for`, and 0.47% for `Did-you-mean` terms.

We observe that the "Drugs", "Gambling", and "Adult Terms" categories exhibit higher rates of poisoned non-auto-corrected searches at 2.86% on average than other categories which exhibit average rates of 0.66%. These terms are more easily monetized than searches for more benign terms such as "Food" or "Cosmetic" products, as the attackers can easily enroll in affiliate ad programs [54]. Additionally, malicious attackers (as opposed to those simply looking for ad revenue) may rationalize that users performing these searches may be more willing to ignore suspicious patterns in URLs or even explicit warning messages by browsers to access the advertised content. Finally, other search engine products such as Google Autocomplete have avoided optimizing and maintaining "inappropriate" predictions for search queries such as adult terms [55]. In contrast to the aforementioned three categories, "Software" linguistic-collision misspellings do not result in high poisoning rates. The comparatively lower exploitation is presumably due to current success of traditional SEO methods for these keywords (note the high poisoning rates for `Original` terms in the English "Software" category). However, because cybercriminals have historically targeted software terms [18, 39], we continue to include "Software" in our analyzed categories in Section VI-B.

While the English "Drugs", "Gambling", and "Adult Terms" categories include poisoned searches for misspellings with every type of correction, the corresponding Chinese categories contain poisoned searches almost exclusively for linguistic-collision misspellings. The disparity between the two is conjectured as an artifact of Baidu's ranking algorithm to prioritize URLs under reputed domains. We find that on Baidu 91.3% of search results for the `Original`, `Showing-results-for`, `Including-results-for`, and `Did-you-mean` terms are under only 1,000 domains (with `baidu.com` alone accounting for 42.7% of results). In contrast, these 1,000 domains account for 83.3% of the results in linguistic-collision misspelling searches. The observations indicate that Baidu exercises less caution on linguistic-collision misspelling searches and is likely to include malicious results.

Alexa top list results. Table II describes the results from the Alexa misspellings (with similar header meanings as in Table I). To investigate the trends and long-tail effect, we use the Alexa top 100, 1,000, and 10,000 website names as target keywords respectively. As mentioned in Section V, the results for the Alexa domains ranked between 1,000 and 10,000 are selected using the RNN described in Section IV. In particular, we crawled 61,088 misspellings which received the lowest entropy from the RNN's entropy estimator. The Alexa 1,000 ground truth dataset blacklist rate is 0.78% with 221 poisoned searches. Interestingly, we see the rate of blacklisted results remains fairly constant based on the RNN results with an average of 0.50% in the Alexa top 1,000–10,000 (116 poisoned searches). Figure 8 shows the longitudinal distribution of attacker activity. On average, 0.54% of the non-auto-corrected results in the Alexa dataset are poisoned. Longitudinally, we find that the level maliciousness is high for the Alexa 100 and 1K, indicating cybercriminals target more on popular domains. After reaching the lowest for the 3K domains, the poison rate slowly increases over the long-tail. Szurdi et al. observed similar long-tail effect on domain typosquatting [47]. Lower popularity domains may have fewer resources to check for poisoned search results, less risk of litigation, and less competition from other cybercriminals.

B. Characteristics of Linguistic-collision Search Results

Next we investigate the detailed properties of misspelling search results that lead to malicious websites.

Comparison of misspelling generation. Intuitively, we would expect users to generate some types of misspellings more frequently than others either through mistyping or confusing the spelling of the original term. For the English results, we compare the non-auto-corrected rate for the wrong vowel substitution method to the average for all misspelling generation, while for Chinese we compare the same pronunciation terms and fuzzy pinyin method to the rest of the misspellings. Because these methods produce misspellings that are closer to the original keyword than the edit-distance 1 heuristics, we would expect these methods to produce more linguistic-collision misspellings. Indeed, we find that for English the wrong vowel method produces a non-auto-corrected rate of 22.85% as compared to the edit-distance 1 misspellings which showed a non-auto-corrected rate of 15.16%. Similarly, for Chinese the more realistic methods outperform the fat-finger misspellings with same pronunciation keywords uncorrected 18.21% of the time and fuzzy pinyin escaping auto-correction for 17.63% of misspellings. Meanwhile, for Chinese the edit distance 1 data set resulted in a non-auto-corrected rate of 7.69%.

Language distribution of linguistic collisions. To determine why Google would fail to correct so many misspellings, we used the Google Translate API to detect the language which returned the detected language and the prediction confidence. The Google Translate API reported that the uncorrected misspellings contained words from 74 languages, while many of the non-English predictions had lower confidence manual spot-checking shows that many of these misspellings are actually valid words in other languages. To better understand the breakdown, we

Domain name	# of Poisoned Searches	# of URLs	Traffic monetization
*.0catch.com	732	109	malvertising
*.atspace.name	63	17	malvertising
hdvidzpro.me	58	58	malvertising
wannajizz.com	49	48	malvertising
theunderweardrawer.co.uk	40	38	malvertising

Table IV: The top five malicious domains using non-auto-corrected misspellings to poison English search terms. The websites typically contain malicious software download or collect personal information. While domains 0catch.com and atspace.name themselves are not intended for malicious activities, cybercriminals utilize the sites' free hosting to promote malicious content through misspelled keywords.

present the top five languages in Table III for the whole dataset, the Alexa domains, and the categories with higher malicious activity. The international flavor of the Alexa domain dataset probably explains the low percentage of English predictions for the Alexa misspellings as many of the top sites serve non-English speakers. Similarly, the lower prevalence of English predictions for the drug's misspellings likely stems from the many unusual drug product names.

Domains (with blacklisted URLs) indexed by multiple misspelled keywords. To better understand how attackers apply linguistic-collision misspelling SEO, we analyze the mapping between misspelled keywords and domains containing blacklisted URLs. Figure 9 displays the CDF of the number of non-auto-corrected misspellings poisoned by the same domains.

In total, for English we saw 1,872 poisoned searches and 538 for Chinese. We observed a distinct difference in SEO tactics with Chinese attackers carefully using paid infrastructure (e.g., xinnet.com) and English search poisoners utilizing free hosting services (e.g., atspace.name). While only 14.1% of the English domains appeared for more than one misspelling, 38.6% of Chinese domains appeared more than once. For English we observed 1,404 malicious domains that together used 2,394 unique blacklisted URLs indicating that some search results contained several blacklisted URLs. While some URLs were optimized to rank for several misspellings, the majority of URLs were targeted at a single misspelling. Rather than attempt to build content with many misspellings, which might cause search engines and users to conclude the content is low quality, the attackers create over 100 webpages, each targeting different misspellings. The Chinese dataset contained 179 domains that deployed 264 URLs. In contrast to the English attacker's reliance on free hosting services to create many highly targeted pages, the Chinese domains tend to be paid and optimized for a wider variety of search terms.

In addition to considering the high level statistics, we also examined the five most successful second-level domains in the English dataset, which are shown in Table IV. Examining how these sites achieve such effectiveness, we find that wannajizz.com, hdvidzpro.me, and theunderweardrawer.co.uk use misspelled URLs and page titles to appear in the first page. On the other hand, the *.0catch.com and *.atspace.name campaigns each used pages targeted at a single original term

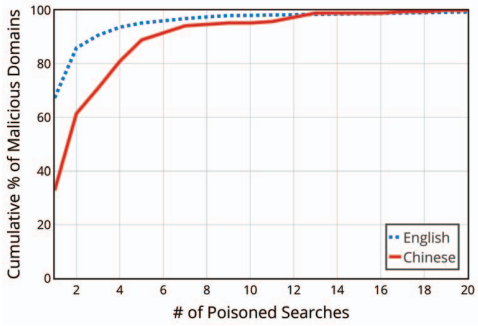


Figure 9: Cumulative distribution function of the number of indexed misspelled keywords that were poisoned by the same domain. Note that 38.6% of Chinese domains poisoned more than one misspelling search result, while only 14.1% of the English domains appeared for multiple misspelling searches. The disparity between the English and Chinese results indicates that the English attackers target individual terms, while the Chinese domains contain a wider variety of misspellings.

	English		Chinese	
Device Type	Original Keywords	Misspellings Targeted by Attackers	Original Keywords	Misspellings Targeted by Attackers
Desktop	36.05 %	11.96 %	39.74 %	21.22 %
Mobile	56.56 %	84.56 %	60.26 %	78.78 %
Tablet	7.40 %	3.48 %	—-	—-

Table V: Device breakdown estimates obtained from the Google Adwords Keyword Planner (we only use the relative numbers returned by Google Adwords as most of the data is imprecise) and Baidu Index. "Original Keywords" estimates market segmentation for all original English and Chinese terms, while "Misspellings Targeted by Attackers" estimates device usage for user searching for the linguistic-collision English and Chinese misspellings in the gambling, drugs, software, and adult term categories.

by enumerating hundreds of misspellings. While the resulting text does not appear coherent to a human, the content is obviously sophisticated enough to convince the search algorithms. Together, these sites provide an interesting view into how the truly successful attackers achieve SEO for linguistic-collision misspellings and also how they monetize their traffic.

C. Search Volume Analysis

To understand how attackers are able to achieve profitability with the linguistic-collision technique, we used the Google Adwords [33] toolsuite for the English dataset and Baidu Index [32] for the Chinese dataset.

Mobile and desktop traffic breakdown. The device breakdown provides insight into how users arrive at the linguistic-collision misspelling results. While in general the device breakdown has similar characteristics between the original and misspelled keywords, Table V shows that keywords from the traditional spam categories (gambling, drugs, software, and adult terms) attract a much higher percentage of mobile users. These results indicate that attackers may tend to target mobile users who are much more likely to misspell words by fat-fingering.

Average search volume. To estimate how many users are exposed to blacklisted search results, we collected search volume for the Chinese non-auto-corrected misspellings from Baidu

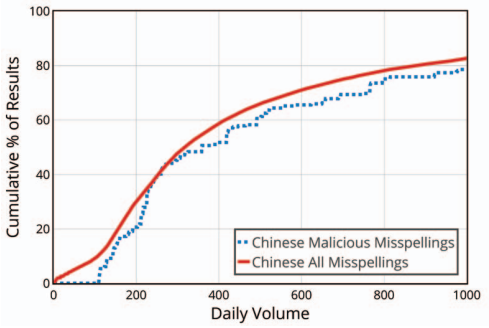


Figure 10: Traffic volume estimates obtained from Baidu Index tool-suite for the Chinese results. The x-axis is the estimated number of searches per day and the y-axis is the cumulative distribution function of individual category. From top to bottom, the curves represent all of the Chinese uncorrected misspellings and the Chinese poisoned misspellings. Note that poisoned misspellings actually receive higher traffic than the other cases indicating that the attackers carefully choose the optimum misspellings.

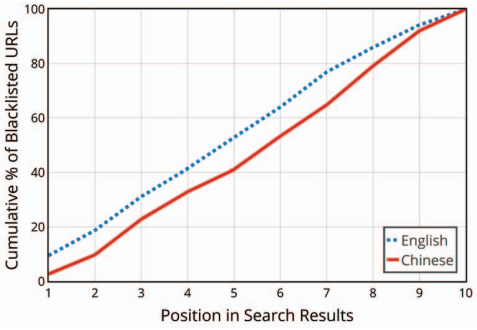


Figure 11: Cumulative percentage of blacklisted URLs in search results for decreasing search result position. Note that some URLs appeared in several search pages so we treat each appearance separately when calculating the CDF.

Index (unfortunately Google Adwords no longer offers API access to traffic volumes). Figure 10 displays the average daily search volume for all of the uncorrected misspellings and the poisoned misspellings. Although many of the poisoned search terms receive little traffic, some may achieve profitability as 21.5% of the poisoned terms receive over 1,000 searches a day. The respectable search volumes per misspelling coupled with the fact that many of these attackers can appear for many misspellings could allow attackers to accumulate significant traffic volumes. Even more worrisome, the search volume results suggest that the attackers are now incentivized to increase their attacks and that the remaining attack surface is actually rather large.

Rankings of search results. One might hope that the blacklisted URLs would be relegated to the bottom of the search results. However, we find that the attackers have managed to be ranked first for 9.5% of the English results. The Chinese blacklisted URLs were less successful with only 2.7% as the first result. As shown in Figure 11, the positions of blacklisted search results for the English URLs appear to follow a uniform distribution, while the Chinese results show comparatively lower ranking. The disparity between the English and Chinese again seems to indicate that the Baidu ranking algorithm prioritizes reputed content sources (see Section VI-A).

D. Case Studies

To further explain how the attackers use linguistic-collision misspelling, we investigate two interesting cases that highlight both attacker incentives and methods.

"Gambling siti" and "hayday loans online". A campaign (involving 89 URLs) mixes content in several languages (with an emphasis on Germanic languages such as English, Finish, and German) to promote advertisements. For example, `raswearsh.890m.com` appears as the fourth result of the search"gambling siti" which is a misspelling of "gambling site" where "siti" is Italian for site. The webpage uses "Siti Gambling" as the title.

By searching small snippets of text from collected attacks, we easily find over 100 other attack URLs using the same snippets to promote a variety of products. Because the resulting pages have valid words (albeit in different languages), the attackers are able to rank in the top 10 search results of misspellings for adult sites, payday loans, gambling, writing services, and options trading kits. To monetize the traffic, each site uses affiliate marketing programs that lead to malicious downloads or phishing pages. For instance, a search for "hayday loans online" (originally "payday loans online") returns `gin.890m.com`, where "hay" is a Spanish word meaning "there are". The website hosts a sign-up form from `leadapi.net` which asks users for social security numbers, date of birth, and bank account information. We find the campaign contains at least 20 websites similar to `gin.890m.com`.

XieHe media ("协和影视"). A malicious website `sds.ccbkr.com` has the title "协和影视". The website induces users to install malicious software with free movies, and also displays various advertisements related with gambling and adult content. However, the title "协和" is the same as the name of a large, well-known hospital in China. If a user directly searches for "协和" on Baidu, most of the returned results are related to that hospital. Indeed, the website `sds.ccbkr.com` will be positioned as the 93rd in the search results (far away from the first page) and it is unlikely that users will reach and click the search result. On the other hand, if a user searches the misspelled keyword "谐和" (which has the same pronunciation as "协和"), the malicious website will show as the first in the search results. Cybercriminals abuse the Chinese misspelling with the same pinyin to achieve higher rank in the search engine. In addition, we find `ccbkr.com` sets wildcard DNS records to display the illicit content on arbitrary subdomains.

VII. Mitigation Discussion

Based on our findings, we propose several potential mitigation strategies. Although affiliate networks should hold their affiliates responsible for participating in linguistic-collision misspelling SEO, the affiliate programs may lack the incentive to enforce such a policy. Realistically, the search engine providers are probably in the best position to defend against linguistic-collision

misspelling by proactively correcting search variants to better protect users from attackers. While auto-correction services have improved significantly, the services could potentially benefit from other data sources. For example, Google Translate data could be used to identify illogical word combinations, words that are outside of the user's normal language, and words that are existent within the language but very rarely used. In addition, search engine providers, such as Google and Baidu, could put forward a more restrictive policy to limit users from purchasing misspelled search keywords and further disincentive affiliate networks caught using linguistic-collision misspellings.

Finally, free hosting services should more strictly enforce the terms and conditions of use for attackers that are utilizing these services to obtain free infrastructure. While we only mentioned `0-catch.com` and `atspace.name` previously, we observed several other hosting sites (`uol.com.br` was another repeat offender) that were allowing attackers to promote dangerous or misleading ads (including at least one pyramid scheme). Enforcing the terms and conditions for these hosting sites could make linguistic-collision misspelling SEO less profitable for the attackers and associating attacker activity to payment details should make the miscreants think twice.

VIII. Related Work

Search engine poisoning. A number of studies examine search engine poisoning where cybercriminals illicitly manipulate search engine results. deSEO [56] generated URL signatures to detect malicious pages that are hosted on compromised legitimate web servers for SEO attacks. SURF [57] designed a browser plugin to detect redirection chains and poisoned search results. Leontiadis et al. [58] conducted a measurement based study on search redirection attacks for online illicit products and found that the conversion rate was higher than email spam. Extending the initial work, Leontiadis et al. [39] performed a four-year longitudinal study to examine the evolution of search engine poisoning, which highlighted a set of traffic redirectors and showed that the overall scale of search poisoning attacks had increased steadily. Liao et al. [59] focused on long-tail search-result manipulation that uses cloud hosting platforms. Wang et al. [60] studied the problem of exploiting autocomplete of suggested queries on search engines to promote illicit content. Our research differs from previous search poisoning work in that we focus on *linguistic-collision misspellings*, a sophisticated class of attacks, which evade current auto-correction defenses to poison search results. We conduct the first large-scale analysis to understand and characterize the abuse of linguistic-collision misspellings to spread malicious content via search results.

Domain typosquatting. In domain typosquatting, attackers register domain names that are purposefully similar to reputed domains. Szurdi et al. [47] investigated long-tail typosquatting registrations, by combining both passive and active domain features to categorize typosquatting domains. Agten et al. [44] focused on a sizeable set of typosquatting targets by using crawled data over a seven-month monitoring period. They found that typosquatting versions of popular domains appear to change owners more frequently and few trademark owners protect themselves by registering typosquatting domains. Nikiforakis et al. [61] studied bit flips in DNS requests (i.e., bitsquatting), where random bit-errors occurring in the memory of commodity hardware can redirect Internet traffic to compromised domains. Khan et al. [45] quantified the harm of typosquatting and found that a typical user loses a second when visiting a typosquatting domain. Kintis et al. [62] studied a specific type of domain squatting, termed "combosquatting," where attackers register domains that combine a popular trademark with one or more phrases. They found that combosquatting is used to perform a spectrum of different types of abuse including phishing, social engineering, affiliate abuse, trademark abuse, and even advanced persistent threats. In addition, several studies have suggested domain squatters often use domain parking services to monetize their holdings [63–65]. Though the attack that we study has a similar incentive to monetize on misspelled user inputs, unlike traditional domain typosquatting, linguistic-collision misspellings circumvent current auto-correction defenses by using legitimate words in other languages.

Security analysis using deep learning. Recently, recurrent neural networks (RNNs) were used as a tool for generating fake Yelp reviews that are able to evade detection by humans and existing algorithms [12]. Long Short-Term Memory (LSTM) networks are a special type of RNN that have the ability to remember long-term dependencies over sequences. LSTM networks have been applied to solve various security problems, such as vulnerability detection [66], website fingerprinting [67], and system logs anomaly identification [11]. In our work, we adapt an RNN architecture to predict misspellings that are likely to avoid auto-correction, to more efficiently identify linguistic-collision search terms.

IX. Conclusion

In this paper, we conduct the first large-scale measurement analysis of search engine poisoning, evaluating over 1.77 million searches on Google and Baidu. By using linguistics and measurement techniques, we systematically analyze the linguistic-collision misspelling attack for English and Chinese. We further develop a deep learning model to more efficiently select non-auto-corrected misspelled keywords.

Our findings reveal that linguistic-collision misspellings widely exist in search engines with 1.19% of search results on the first page directing to blacklisted websites. We also discover the primary target is drug, gambling, and adult terms. In addition, we observe that mobile users disproportionately search for misspellings. Although search engine providers have already reduced the attack surface of typosquatting by adding auto-correction, linguistic-collision misspellings present a vulnerability that attackers can exploit to promote malicious links. Our study sheds light on this new threat and provides insights to ultimately mitigate the problem.

Acknowledgments

We thank the anonymous reviewers for their valuable comments to improve the paper. We thank Christian Kreibich

and the International Computer Science Institute for providing Spamhaus data. Minhui Xue is supported by the Optus Macquarie University Cyber Security Hub.

REFERENCES

[1] Internet World Stats. *Number of Internet Users by Language*. http://www.internetworldstats.com/stats7.htm. June 2017.

[2] Amy Gesenhues. *Organic Search Drives 51% Of Traffic, Social Only 5%*. http://searchengineland.com/study-organic-search-drives-51-traffic-social-5-202063. Aug. 2014.

[3] Google. *Search Engine Optimization Starter Guide*. https://www.google.com/webmasters/docs/search-engine-optimization-starter-guide.pdf. Sept. 2017.

[4] Alexandros Ntoulas, Marc Najork, Mark Manasse, and Dennis Fetterly. "Detecting Spam Web Pages through Content Analysis". In: *15th International Conference on World Wide Web (WWW)*. May 2006.

[5] Baoning Wu and Brian D Davison. "Identifying Link Farm Spam Pages". In: *14th International World Wide Web Conference (WWW)*. May 2005.

[6] Jennifer Slegg. *Targeting Keyword Variations for Increased Search & Pay per Click Traffic*. http://www.jenniferslegg.com/2007/04/06/targeting-keyword-variations-for-increased-search-pay-per-click-traffic/. Apr. 2007.

[7] David Z. Morris. *German Court Orders Amazon to Stop 'Typo-Targeting' Ads for Birkenstocks*. http://fortune.com/2017/12/30/amazon-typo-targeting-birkenstock-advertising/. Dec. 2017.

[8] Shubham Grover. *Snickers Misspelling Search Keyword Campaign Reached 50K People In 3 Days*. http://www.digitalvidya.com/blog/snickers-misspelling-search-keyword-campaign-reached-50k-people-in-3-days/. Oct. 2015.

[9] Greg Sterling. *Nearly 60 Percent of Searches Now from Mobile Devices*. http://searchengineland.com/report-nearly-60-percent-searches-now-mobile-devices-255025. Aug. 2016.

[10] Chen Yuan. *Chinese Language Processing*. Shanghai Education Publishing Company, 1997.

[11] Min Du, Feifei Li, Guineng Zheng, and Vivek Srikumar. "DeepLog: Anomaly Detection and Diagnosis from System Logs through Deep Learning". In: *24th ACM Conference on Computer and Communications Security (CCS)*. Oct. 2017.

[12] Yuanshun Yao, Bimal Viswanath, Jenna Cryan, Haitao Zheng, and Ben Y. Zhao. "Automated Crowdturfing Attacks and Defenses in Online Review Systems". In: *24th ACM Conference on Computer and Communications Security (CCS)*. Oct. 2017.

[13] Sepp Hochreiter and Jurgen Schmidhuber. "Long Short-Term Memory". In: *Neural Computation* 9.8 (Nov. 1997).

[14] Jennifer Valentino-DeVries. *What Words Get Misspelled in Web Searches?* https://blogs.wsj.com/digits/2010/06/04/what-words-get-misspelled-in-web-searches/. June 2010.

[15] Christopher Mele. *Is Wisconsin Really That Hard to Spell?* https://www.nytimes.com/2017/05/31/us/misspelled-words-states.html. May 2017.

[16] Marjory Meechan. *Google's Algorithm Update for Misspelled Words: A Big Change for SEO*. https://www.morevisibility.com/blogs/seo/googles-algorithm-update-for-misspelled-words-a-big-change-for-seo.html. Dec. 2008.

[17] Xiaoqing Hu. "The Examples Analysis of Chinese-Error Correction Function in Search Engines". In: *Library and Information Service Online* (2008).

[18] Kirill Levchenko, Neha Chachra, Brandon Enright, Mark Felegyhazi, Chris Grier, Tristan Halvorson, Chris Kanich, Christian Kreibich, He Liu, Damon McCoy, Andreas Pitsillidis, Nicholas Weaver, Vern Paxson, Geoffrey M. Voelker, and Stefan Savage. "Click Trajectories: End-to-End Analysis of the Spam Value Chain". In: *32nd IEEE Symposium on Security and Privacy*. May 2011.

[19] VirusTotal. *VirusTotal*. https://www.virustotal.com. Mar. 2018.

[20] FDA. *Public Notification: "Clalis" Contains Hidden Drug Ingredient*. https://www.fda.gov/Drugs/ResourcesForYou/Consumers/BuyingUsingMedicineSafely/MedicationHealthFraud/ucm359070.htm. 2015.

[21] *Search Engine Market Share*. https://netmarketshare.com/search-engine-market-share.aspx. 2018.

[22] Philip Petrescu. *Google Organic Click-Through Rates in 2014*. https://moz.com/blog/google-organic-click-through-rates-in-2014. 2014.

[23] Eric Sharp. *The First Page of Google's Search Results Is the Holy Grail for Marketers*. https://www.protofuse.com/blog/details/first-page-of-google-by-the-numbers/. Apr. 2014.

[24] Fred J. Damerau. "A Technique for Computer Detection and Correction of Spelling Errors". In: *Communications of the ACM* 7.3 (Mar. 1964).

[25] V. I. Levenshtein. "Binary Codes Capable of Correcting Deletions, Insertions and Reversals". In: *Soviet Physics Doklady* 10 (Feb. 1966).

[26] Tyler Moore and Benjamin Edelman. "Measuring the Perpetrators and Funders of Typosquatting". In: *14th International Conference on Financial Cryptography and Data Security*. Feb. 2010.

[27] Kazuya Kawakami, Chris Dyer, and Phil Blunsom. "Learning to Create and Reuse Words in Open-Vocabulary Neural Language Modeling". In: *Annual Meeting of the Association for Computational Linguistics (ACL)*. July 2017.

[28] Yoon Kim, Yacine Jernite, David Sontag, and Alexander M. Rush. "Character-Aware Neural Language Models". In: *13th AAAI Conference on Artificial Intelligence (AAAI)*. Feb. 2016.

[29] Martin Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mane, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viegas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. 2015. URL: https://www.tensorflow.org/.

[30] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *CoRR* abs/1412.6980 (2014). URL: http://arxiv.org/abs/1412.6980.

[31] *Sogou Pinyin Input Dictionaries*. https://pinyin.sogou.com/dict/. 2018.

[32] *Baidu Index*. https://zhishu.baidu.com/. Jan. 2018.

[33] Google. *Google Adwords*. https://adwords.google.com/home/. Jan. 2018.

[34] Google. *Google Translate API*. https://cloud.google.com/translate. Mar. 2018.

[35] Google. *Google Safe Browsing API*. https://safebrowsing.google.com/. Mar. 2018.

[36] Yandex. *Safe Browsing API*. https://tech.yandex.com/safebrowsing/. Mar. 2018.

[37] Spamhaus. *Spamhaus*. http://www.spamhaus.org/. Mar. 2018.

[38] Baidu. *Baidu-International Antivirus*. http://antivirus.baidu.com/en/. Mar. 2018.

[39] Nektarios Leontiadis, Tyler Moore, and Nicolas Christin. "A Nearly Four-Year Longitudinal Study of Search-Engine Poisoning". In: *21st ACM Conference on Computer and Communications Security (CCS)*. Oct. 2014.

[40] *Ranker*. https://www.ranker.com. Mar. 2018.

[41] *Defense News*. http://people.defensenews.com/top-100/. Nov. 2017.

[42] *Pharmaceutical Spam Keywords*. http://www.localseoguide.com/the-ultimate-list-of-pharmaceutical-spam-keywords/. Nov. 2017.

[43] Kaggle. *Kaggle Datasets*. https://www.kaggle.com/datasets. Nov. 2017.

[44] Pieter Agten, Wouter Joosen, Frank Piessens, and Nick Nikiforakis. "Seven Months' Worth of Mistakes: A Longitudinal Study of Typosquatting Abuse". In: *22nd Annual Network & Distributed System Security Symposium (NDSS)*. Feb. 2015.

[45] Mohammad Taha Khan, Xiang Huo, Zhou Li, and Chris Kanich. "Every Second Counts: Quantifying the Negative Externalities of Cybercrime via Typosquatting". In: *36th IEEE Symposium on Security and Privacy*. May 2015.
[46] Alexa. *Alexa List*. https://www.alexa.com/topsites. Nov. 2017.
[47] Janos Szurdi, Balazs Kocso, Gabor Cseh, Jonathan Spring, Mark Felegyhazi, and Chris Kanich. "The Long "Taile" of Typosquatting Domain Names". In: *23rd USENIX Security Symposium*. Aug. 2014.
[48] Google. *Google Adwords Keyword Planner*. https://support.google.com/adwords/answer/2999770?hl=en. Jan. 2018.
[49] Peter Norvig. *Peter Norvig N-grams Dataset*. http://norvig.com/ngrams/. Jan. 2018.
[50] *Open Office Dictionary*. https://extensions.openoffice.org/en/project/us-english-spell-checking-dictionary. Jan. 2018.
[51] *Assorted English Words List*. https://github.com/dwyl/english-words. Jan. 2018.
[52] John Lawler. *An English Word List*. http://www-personal.umich.edu/~jlawler/wordlist.html. Mar. 1999.
[53] Michael Hodgdon. *Value of Organic First-Page Results*. https://www.infront.com/blog/the-infront-blog/2015/06/17/value-of-first-page-google-results. June 2015.
[54] Damon McCoy, Andreas Pitsillidis, Jordan Grant, Nicholas Weaver, Christian Kreibich, Brian Krebs, Geoffrey Voelker, Stefan Savage, and Kirill Levchenko. "PharmaLeaks: Understanding the Business of Online Pharmaceutical Affiliate Program". In: *21st USENIX Security Symposium*. Aug. 2012.
[55] Google. *Google Autocomplete Policies*. https://support.google.com/websearch/answer/7368877. Mar. 2018.
[56] John P John, Fang Yu, Yinglian Xie, Arvind Krishnamurthy, and Martin Abadi. "deSEO: Combating Search-Result Poisoning". In: *20th USENIX Security Symposium*. Aug. 2011.
[57] Long Lu, Roberto Perdisci, and Wenke Lee. "SURF: Detecting and Measuring Search Poisoning". In: *18th ACM Conference on Computer and Communications Security (CCS)*. Oct. 2011.
[58] Nektarios Leontiadis, Tyler Moore, and Nicolas Christin. "Measuring and Analyzing Search-Redirection Attacks in the Illicit Online Prescription Drug Trade". In: *20th USENIX Security Symposium*. Aug. 2011.
[59] Xiaojing Liao, Chang Liu, Damon McCoy, Elaine Shi, Shuang Hao, and Raheem Beyah. "Characterizing Long-tail SEO Spam on Cloud Web Hosting Services". In: *25th International Conference on World Wide Web (WWW)*. May 2016.
[60] Peng Wang, Xianghang Mi, Xiaojing Liao, XiaoFeng Wang, Kan Yuan, Feng Qian, and Raheem Beyah. "Game of Missuggestions: Semantic Analysis of Search-Autocomplete Manipulations". In: *25th Annual Network & Distributed System Security Symposium (NDSS)*. Feb. 2018.
[61] Nick Nikiforakis, Steven Van Acker, Wannes Meert, Lieven Desmet, Frank Piessens, and Wouter Joosen. "Bitsquatting: Exploiting Bit-flips for Fun, or Profit?" In: *22nd International Conference on World Wide Web (WWW)*. May 2013.
[62] Panagiotis Kintis, Najmeh Miramirkhani, Charles Lever, Yizheng Chen, Rosa Romero-Gomez, Nikolaos Pitropakis, Nick Nikiforakis, and Manos Antonakakis. "Hiding in Plain Sight: A Longitudinal Study of Combosquatting Abuse". In: *24th ACM Conference on Computer and Communications Security (CCS)*. Oct. 2017.
[63] Sumayah Alrwais, Kan Yuan, Eihal Alowaisheq, Zhou Li, and XiaoFeng Wang. "Understanding the Dark Side of Domain Parking". In: *23rd USENIX Security Symposium*. Aug. 2014.
[64] Thomas Vissers, Wouter Joosen, and Nick Nikiforakis. "Parking Sensors: Analyzing and Detecting Parked Domains". In: *22nd Annual Network & Distributed System Security Symposium (NDSS)*. Feb. 2015.
[65] Najmeh Miramirkhani, Oleksii Starov, and Nick Nikiforakis. "Dial One for Scam: A Large-Scale Analysis of Technical Support Scams". In: *24th Annual Network & Distributed System Security Symposium (NDSS)*. Feb. 2017.
[66] Zhen Li, Deqing Zou, Shouhuai Xu, Xinyu Ou, Hai Jin, Sujuan Wang, Zhijun Deng, and Yuyi Zhong. "VulDeePecker: A Deep Learning-Based System for Vulnerability Detection". In: *25th Annual Network & Distributed System Security Symposium (NDSS)*. Feb. 2018.
[67] Vera Rimmer, Davy Preuveneers, Marc Juarez, Tom Van Goethem, and Wouter Joosen. "Automated Website Fingerprinting through Deep Learning". In: *25th Annual Network & Distributed System Security Symposium (NDSS)*. Feb. 2018.

APPENDIX

A. RNN Comparison to Random Forest and Naive Bayes

We compare the accuracy of non-auto-corrected predictions of our RNN model with random forest and naive Bayes algorithms using two approaches for each algorithm.

Approach 1. The first approach directly classifies whether a misspelling string is likely to be non-auto-corrected by Google. The brute-force search results of manually selected categories contain both positive and negative cases, which we use as the training dataset. Because both of the classification algorithms require fixed length input vectors, we pad the variable length words with null values. After training, the algorithms estimate the probability that a given misspelling will be autocorrected. However, because the ground truth data is generated from relatively few original terms (compared to all possible words in use on the Internet), the algorithms struggle to generalize for misspellings generated from other original terms.

Approach 2. The second approach is similar to the one that we developed in Section IV. In this approach, we generate a training dataset from dictionary words. The classifier learns the future character distribution based on the prefixes. The entropy of a prediction estimate the likelihood whether a misspelling candidate will be automatically corrected.

For misspellings from Alexa top 1,001–10,000 terms, our RNN approach achieves a hitting rate of 38.04% (as shown in Table II). At the same hitting rate on the Alexa top 1K ground truth, we need to collect 127,438 searches with the best predictions from the RNN. When crawling the same number of searches, the naive Bayes model with approach 1 yields a hit rate of 13.6% . We hypothesize that the naive Bayes model's poor performance stems from the strong dependency between adjacent characters. For approach 2, naive Bayes achieves a hit rate of 15.2% (most likely due to the reduced input size). Since random forests can capture dependencies between input features, the random forest classifier outperforms naive Bayes for both approach 1 and approach 2. For approach 1, random forest exhibits a hit rate of 29.9%, and for approach 2 the hit rate is 22.8%, both of which are less efficient than the RNN predictions.