Pharmaguard WebApp: an application for the detection of illegal online pharmacies

Matteo Contini, Igino Corona, Alessio Mulas, Giorgio Giacinto, Davide Ariu

DIEE, University of Cagliari, Italy
{name.surname}@diee.unica.it

Abstract. We present a demo for PharmaGuard, a novel system for the automatic discovery of illegal online pharmacies. With its easy to use graphic user interface, a web application architectural approach and leveraging the powers of automatic knowledge discovery, PharmaGuard can assist law enforcement agencies in identifying, blacklisting and shutting-down illegal pharmacies.

Keywords: Detection of Illegal Pharmacies; Search Engines; Pattern Classification; Human-Machine Interaction

1 Introduction

In this paper, we present an actual implementation of PharmaGuard, a system aimed at assist law enforcement operators toward an easier identification and blacklisting of illegal online pharmacies. PharmaGuard theoretical basis has been already presented in the paper *PharmaGuard: Automatic Identification of Illegal Search-Indexed Online Pharmacies* [1], since the time of its publication the described scenario hasn't changed: cybercriminals and law enforcement operators are still trading blows in a fight to either control or dismantle the illicit online drug selling market. An increasing number of websites sells pharmaceutical products violating laws and without any guarantee about the actual composition of the sold merchandise. Law enforcement agencies (from now onwards simply LEA) still don't have a mature set of tools at their disposal to cope with the complex technological and jurisdictional aspects of the phenomenon. The system proposed in this paper combines an automatic approach to illegal pharmacies discovery with a human based check and validation process. These are the core characteristics of the chosen approach:

1. Feature extraction and classification are completely automatic.
2. Human users can change the classification and delete discovered web pages.
3. Detection is based on human validated results stored within the system.

2 Architecture

PharmaGuard has been designed for two kind of users: law enforcement operators and academic researchers. The average user is as a computer lettered person with a simple goal: to check a list of suspected web pages, inspect their content and classify them. Web Pages within the system are classified as belonging to one and only one of these categories:

- **ILLEGAL Pharmacy:** the web page is considered to be a full fledged illegal pharmacy or is it at least involved in the illegal selling of drugs.
- **Legal Pharmacy:** the web page is either a legal online pharmacy or there are no enough "proofs" to actually establish if it's legal or not.
- **Other:** the web page is not involved in any kind of drug selling at all.
- **?:** the web page is hard or impossible to evaluate because of the nature of the content (e.g. content is written in an unknown foreign language).
- **Pharmacy Advertisement:** the web page is not involved in drug selling but rather on advertisement of similar content. Web pages that belong to this category can link to real online pharmacies.

System classifies web pages using only three labels: **ILLEGAL Pharmacy**, **Legal Pharmacy** and **Other**. Human users, while changing the classification of a web page, can use all five labels. System uses also an additional flag called *"useful/not useful"*. System set by default this flag to *"useful"* for every classified web page; users can change flag's value at any time. The next paragraphs will explain this flag role in the actual discovery of potentially illegal online pharmacies. PharmaGuard system can be accessed by users thanks to a Graphic User Interface (from now onwards GUI). System provides two different access levels: Validator and Administrator; the former can classify and delete web pages, the latter can also create new users.

The following list describes a typical scenario of use for a Validator user:

1. A user visualizes the login page and access the system.
2. A list of analyzed web pages is shown.
3. User selects one web page from the list.
4. System shows complete information regarding the selected web page.
5. User can either change or confirm the actual label.
6. User presses the save button (e.g. user presses the save button).
7. User continue from point 2 or leave the system.

It's worth noting that user is shown a complete list of all web pages: new ones with a label suggested by the system, previously labeled ones that has been already validated by a human user (i.e. not necessarily the current user). Once a web page is selected, user can see the following information:

- **Results of automatic classification:** system assigned labels.
- **URLs:** web page's initial and final URL (i.e. in case of redirection).
- **Download time:** a timestamp of web page's download time.
- **Network source:** TTL, IP address and Autonomous System informations.
- **Label:** GUI's element for web page's classification change.
- **Checked by:** list of users that have previously checked the web page.
- **Useful:** GUI's element for web page's *"useful"* flag change.
- **Snapshot:** web page's snapshot.

- **Page:** web page's html.

User's manual classification is important for the system and heavily influences its performances. Webpage Finder searches the web for new content according to a criteria of similitude by exploiting search engines capabilities of suggesting content similar to that of a given URL. PharmaGuard keeps stored all previously analyzed web pages; all those labeled as **ILLEGAL Pharmacies** and flagged as *Useful* are used as input for search engine based queries. This is how the system founds new web pages to analyze. System's ability to find illegal pharmacies depends on previous analysis, the more currently classified web pages are samples of actual illegal pharmacy pages the better the system will be at finding potentially illegal new ones. While system's ability to discover new potentially illegal page is tied to the quality of previous analysis, on the other hand feature extraction and classification are not. It's worth noting that users have an impact only on system's performances in discovering new content while automatic classification is unaffected. System's behavior can be seen as a sequence of steps: download a web page, extract information and features, classify, store the result, show the result, create a list of web pages to download, repeat. Although currently not completely divided into separate components, system has been designed to be modular. System's functionalities can be grouped into blocks according to the previous sequence and mapped to the components described in this short list:

- **Scheduler:** this component starts the workflow loop providing the Browser and Metadata Extractor components with a list of web pages to analyze.
- **Browser:** this component downloads a given web page's html and creates a snapshot for a visual inspection. Browser is based on Selenium WebDriver web automation tool and guarantees that inspected web page's content is exactly the same a human user would see with his browser. Downloaded html is the one actually "visible" from the browser and takes in account all changes to the DOM such as javascript injection (i.e. ajax). Downloaded html is sent to the Feature Extraction for further analysis.
- **Metadata Extractor:** given a web page's URL, this component collects additional information such as IP, TTL, Autonomous System Number, etc.
- **Feature Extraction:** this component takes an html as input and provides a list of extracted features as output.
- **Classification:** based on a sequence of two different classifiers, this component is able to classify a given web page as either a legal online pharmacy, an illegal one or as totally unrelated web page that is not a pharmacy at all. As previously stated, these labels are the result of the conjoined work of two classifiers: the first one, called PHARMA vs OTHER, decides if a web page is an online pharmacy or not; the second one, called PHARMA vs PHARMA decides if an online pharmacy is either a legal or illegal one. The final result is sent to the Core component. It's worth noting that the second classifier actually works only if the first one detected an online pharmacy.
- **Webpage Finder:** modern search engines have lot of advanced search functionalities, among these there is "related content search". Webpage Finder

takes all previously analyzed web pages that has been labeled as **ILLEGAL Pharmacies** and are currently flagged as *Useful* then feeds them to search engines in order to find "related" web pages. Webpage Finder's output will be used by Scheduler component.

- **Core:** this component is based on Django framework, uses a MVC architectural pattern and takes care of data persistence, components communication and GUI requirements. Core component uses a relational database to store web page's classification labels, downloaded content and all the necessary information for users login and session handling.

We already described system's purpose and main components, we also presented a typical scenario of use for a human user. We can finally describe a daily "workflow" from the machine point of view, a set of steps required in order to provide the final user with a list of classified web pages:

1. System's Scheduler fetches a list of potentially interesting web pages and submits it to System's Browser and Metadata Extractor components.
2. System's Browser "visits" the web page, creates a snapshot of it and downloads the associated html content.
3. System's Metadata Extractor component collects all required additional information regarding the web page performing additional queries (e.g. whois).
4. Feature Extraction component analyzes web page's html, crawls it and extracts all required features passing them to the next block.
5. Classification component uses a sequence classifiers to label each web page.
6. Core component takes care of saving all the collected data and final analysis result while, at the same time acting as sole interface for human Users.
7. Webpage Finder performs a search for web pages creating in a list of potentially illegal web pages. Scheduler component will then start again.

3 Conclusions

Illegal online pharmacies, despite LEAs effort, are still a problem without solution. Substances sold don't just constitute a source of income for criminals but constitute a real threat for the health of the buyers. In this work, we presented PharmaGuard, a powerful tool for LEAs that can be helpful in many ways:

1. Finding "never-seen-before" illegal pharmacies.
2. Keeping track of all already seen web pages as a reference.
3. Easing the collaboration of multiple agents.

References

1. I. Corona, M. Contini, D. Ariu, G. Giacinto, F. Roli, M. Lund, e G. Marinelli, "PharmaGuard: Automatic Identification of Illegal Search-Indexed Online Pharmacies", IEEE International Conference on Cybernetics - Special session Cybersecurity (CYBERSEC), 2015.