Solving CAPTCHAs Automatically for Web Crawling

Swapnil Mane,Mayuri Lokare, Prof. B. B. Gite

Student, Department of Computer Engineering, SAE, Maharashtra, India
Student, Department of Computer Engineering, SAE, Maharashtra, India
HOD, Department of Computer Engineering, SAE, Maharashtra, India

ABSTRACT

Web crawler is an automated script which browses the World Wide Web in methodical, automated manner. This process is called as web crawling or web spidering. Web crawler is also known as web spider or web robot. Crawler needs to catch web pages frequently for updating the data. But by this process, performance and speed may get affected. Crawler can not retrieve data in a great depth. Hence, to reduce load and also for authentication, web server requests web crawler to verify or cross check themselves against CAPTCHAs. Today, more than two billion web pages are there on web server. One human can not enter CAPTCHAs for every time while entering to required web page. Hence, to solve CAPTCHAs automatically we explained a system for text or image recognition from CAPTCHA images. Our focus is on reliable and firm text or image extraction, recognition, putting resolved CAPTCHA to crawler system to continue the web crawling process without involving human directly.

Keyword : *Web Crawler, CAPTCHA, Text recognition.*

1. INTRODUCTION

In 1993, World Wide Web was started with few thousand web pages. Today, more than two billion web pages are there on WWW[1]. Retrieval of information is a challenge because of abundance of data. Thousands of results appear when we search something on internet. So it is the bigger job for search engines to search and sort data according to user’s interest in fraction of seconds. Web crawler is an automated script which browses the World Wide Web in methodical, automated manner. Web crawler makes copy of each web page which is visited by it for further use. This process is frequentative. This process repeats till it comes close to user’s interest area. Search engines use algorithms to continue this process and come close to user’s query. Some algorithms are is in use - Breadth first search, Best first search, Page Rank algorithm, Genetic algorithm, Naïve Bayes classification algorithm[7]. Dynamic page generation and large volume are some of the properties of web search engines. But these make working of web crawler more difficult. Internet browsers and Big Data companies use web crawlers to browse WWW in methodogical and automated way. Web crawlers are used to update data by those companies. For this crawler needs iterative crawling to decrease age and improve freshness of web page. For this, web crawler sends thousands of requests per second to web server. There are many web crawlers working for a company. On the other side, individual users also send requests to web servers. Because of all these requests coming at same time, web servers are overloaded many times. Web servers have something to do to reduce extra load. These web crawlers send requests from same IP address to web server. If someone wants to hack web server or wants to create traffic at the server, then he can send that too much requests from same machine. So sometimes web server thinks that it can be a malicious trap or scene of DOS attack because of these web crawlers. Hence to verify that it is trap or not web servers ask to solve CAPTCHA to those web crawlers to slow down the traffic. But because of this web crawlers have to stop crawling process and solve the CAPTCHA with human involvement. And it can not be possible every time. It slows down the crawling process. It directly affects the performance of web crawling process. One human can not enter CAPTCHAs for every time while entering to required web page. To overcome this problem web crawling process can be made embedded by adding application of resolving CAPTCHA with existing web crawling system. Thus our problem can be defined as to develop a software application to solve the problem of server verification requirement usually through CAPTCHAs after frequently sending requests for access or to download data from same IP address to web server in web crawling process. Hence to reduce the human involvement and making the entire process of web crawling automated with high improved efficiency and less delay.

2. LITERATURE SURVEY

A crawler is sometimes called as spider, bot or agent. It is a software whose purpose is to perform web crawling. A web crawler or spider is an ethical computer program that browses the WWW in sequencing, methodical and automated manner for search engines like Google, Microsoft Bing, yahoo etc. and for many BigData companies. Today the size of the web is millions of web pages that is too high and the growth rate of web pages are increasing exponentially, because of this the main problem for search engine is dealing this amount of the size of the web. The information is continuously being produced and updated anywhere and anytime by means of easy web platforms, and social networks.

The developed web crawler shown in Figure 1. It is multi-threaded which downloads a URL from World Wide Web(WWW) and then stores it in repository. It sticks fast to robot exclusion protocol. It extracts the list of URL from the downloaded web pages and then adds them to the URL Queue. The Duplicate URL Elimination eliminates any repeated URLs in the URL Queue. Then site ordering module ranks the URL according to structural and content similarity as implemented ordering algorithm. Indexer updated ranking in repository. Crawl Scheduler then chooses the new URL to be crawled from the ordered URLs. The user gets the ordering results fetched from the repository[6].

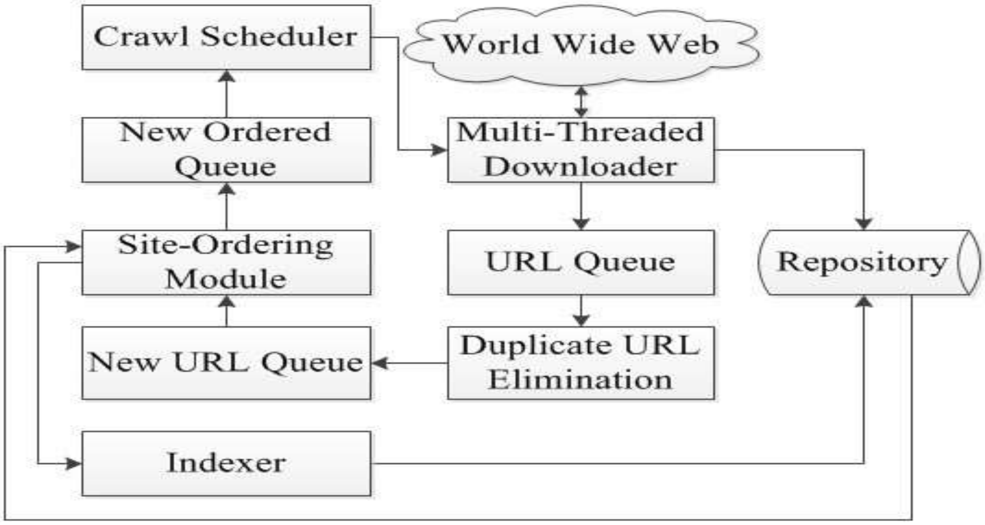

Fig 1. Architecture of Web Crawler

The high rate of change implies that by the time the crawler is downloading the last pages from a site, it is very likely that new pages have been added to the site, or that pages have already been updated or even deleted. Performance of web crawler based on freshness and age. When the same copy exists in both local and remote sources, then it is considered to be the “fresh” page. Cho and Garcia [7] calculated the freshness of a page as shown in figure 1,

$$F(e_i;t) = \begin{cases} 1 & \text{if } e_i \text{ is up-to-date at time } t \\ 0 & \text{otherwise.} \end{cases}$$

Fig 2

where ei is the element of database. Freshness focus on whether or not the local copy is the current copy of the resource. And the age of a page can be given as shown in figure 2,

$$A(e_i;t) = \begin{cases} 0 & \text{if } e_i \text{ is up-to-date at time } t \\ t - t_m(e_i) & \text{otherwise.} \end{cases}$$

Fig 3

Where tm (ei) is the time of first modification of ei after the most recent synchronization [6]. Age focus on how long ago the local copy was updated. The freshness drops to zero when the real-world element changes and the age increase linearly from that point on. When the local element is synchronized to the real-world element, its freshness recovers to one, and its age drops to zero[7].

New tools and techniques are crucial for intelligently searching for useful information on the web. But more frequent access to any web page results in CAPTCHA. CAPTCHA is a technique to tell computer and human apart. In this case CAPTCHAs are preventing crawlers from caching data from web pages to search engine's databases. Human entering CAPTCHAs in crawling process will significantly increase the delay in process because there remains a large gap in ability between human and machine vision systems, even when reading printed text. Also it is not convenient for human to solve thousands of millions of CAPTCHAs.

There have been a different methods dealing with text detection and reorganization in images. CAPTCHAs are nothing but image embedded with text thus in order to solve these CAPTCHAs one need to consider text detection and recognition using image processing.

To detect text regions embedded in images, Miss. Poonam B. Kadam, Mrs. Latika R. Desai have propose a Hybrid Approach framework which is more concentrating to give input to OCR having less false positive which result into efficient and accurate text recognition. For robust text detection and recognition right now particular type of EZ Gimpy CAPTCHA images as input and by applying different preprocessing method to remove complex background. Detecting connected components approach developing learning based methods for text extraction from complex backgrounds and text normalization for OCR recognition[5].

Julinda Gllavata1, Ralph Ewerth1 and Bernd Freisleben proposed robust algorithm is based on employing a color reduction technique, a method for edge detection and region segmentation, and selecting text regions based on their horizontal projection and geometrical properties. Their software is completely written in JAVA to be able to easily run the code in parallel on possibly heterogeneous networked computing platforms. Experimental results on a set of images have demonstrated the performance of this approach, achieving an overall recall of 88.7w and a precision of 83.9w[6].

Max Jaderberg, Karen Simonyan, Andrea Vedaldi, Andrew Zisserman have introduced a new synthetic data and artificial neural networks framework for scalable, state-of-the-art word recognition – synthetic data generation followed by whole word input CNNs. They introduced a new synthetic word dataset, orders of magnitude larger than any released before[8].

Another approach called scene text detection, in which layout analysis of color decomposition and horizontal alignment is performed to search for image regions of text strings. Proposed character descriptor is effective to extract representative and discriminative text features for both recognition schemes. To model text character structure for text retrieval scheme, we have designed a novel feature representation, stroke configuration map, based on boundary and skeleton. It also proves that the assumptions of color uniformity and aligned arrangement are suitable for the captured text information from natural scene. A dictionary search based method for recognition error correction is also proposed. Actually, the modification they have made to the Random Forest classifier for the purpose of simultaneous text classification and character recognition is quite general and thus can be applied to other domains and problems, in which tasks are carried out at different levels and features can be shared by these tasks. This direction is worthy of further exploration.

Alessandro Bissacco, Mark Cummins, Yuval Netzer, Hartmut Neven have describe PhotoOCR. Focus is reliable text extraction from smartphone imagery, with the goal of text recognition as a user input modality similar to speech recognition. Commercially available OCR performs poorly on this task. Recent progress in machine learning has substantially improved isolated character classification; they build on this progress by demonstrating a complete OCR system which also incorporate modern datacenter-scale distributed language modeling. This approach is capable of recognizing text in a variety of challenging imaging conditions where traditional OCR systems fail, notably in the presence of substantial blur, low resolution, low contrast, high image noise and other distortions. It also operates with low latency; mean processing time is 600 ms/image. The system is currently in use in many applications at Google, and is available as a user input modality in Google Translate for Android[9].

Traditionally there have been a different methods dealing with text detection and reorganization in images. Some approaches to text detection classified into three categories: texture-based methods, region based methods, and hybrid methods. But these methods do not yields significant performance. Texture-based methods involve texture properties of text such as style, orientation, and wavelet coefficients, the construction of gray-level co-occurrence matrix. These methods are computation demanding as all locations and scales are exhaustively scanned. Moreover, these algorithms mostly concentrate to detect horizontal texts. Region based methods use the properties of the color or gray scale or alignment in a text region or their differences in properties of the background. First extract candidate text regions through segmentation or clustering and then remove non-text regions. The third category, hybrid methods, is a fusion of region-based and texture-based methods. Different document or web ,e-mail images, in which text characters are normalized and proper resolutions, natural scene images, embed text can be in size, shapes and orientations into complex background, difficult to find text. Also there exists one system called OCR (Optical Character Recognition).

In this we present a unified framework for text detection and recognition in natural images. It consist of four stages text detection, text localization, text extraction, text recognition. We can use these stages text detection, localization, and extraction interchangeably. Text detection consists of determination of the occurrence of text in images. Text localization is the process of determining the location of text. In text extraction stage the text components in images are segmented from background. After, the extracted text images can be converted into plain text using OCR technology. Through Text detection and recognition in images, which coupling of text-based searching technologies and optical character recognition (OCR), is now recognized as a key component which are present in the images. Unfortunately, text characters contained in images can be multi-colour or any gray-scale value, variable size, low resolution, and embedded in noisy backgrounds. Many experiments done on text recognition by applying conventional OCR technology directly it leads to decrease rates of recognition. Some text images having complex background. Detecting texts in complex images has received lot of attentions and remains a challenge for most practical systems. In this work, an effort is made to build an effective and convenient detection system for texts having variation is style, font, color in images.

Text detection from any kind of images like document, digital camera based and web, email is challenging due to the random text appearances and complex backgrounds. To detect text regions embedded in those images, we propose a new framework which is more concentrating to give input to OCR having less false positive which result into efficient and accurate text recognition. Because more accurate detection is helps to make character recognition more effective and accurate. CAPTCHA images as input and by applying different preprocessing method to remove complex background. First we detect connected components in image and then character grouping is performed to detect text characters, Then, text recognition is performed to identify text in input image. This approach developing learning based methods for text extraction from complex backgrounds and text normalization for OCR recognition. It prove that CAPTCHA can be breakable using this approach.

As an additional contribution, a novel image database with texts of different scales, colors, fonts, and orientations in diverse real world scenarios, is generated and released. Extensive experiments on standard benchmarks as well as the proposed database demonstrate that the proposed system achieves highly competitive performance, especially on multi-oriented texts.

To address these problems, we are describing OCR technology that recognises and captures alphanumeric characters on a computer at high speed. It works in flow of text detection, text localization, and text recognition.

3. PROPOSED SYSTEM

In our system or application, we are linking web crawling process with this CAPTCHA resolver. If number of web crawlers related to a company are sending thousands of requests per second using same IP address to web server, then web server thinks that it can be a malicious trap or some chance of DOS attack. So web server asks to solve CAPTCHA in between the process to confirm that it is not robot or a bot for disturbing the process. But, periodically this can be happened so everytime it is needed to stop the whole process and enter that security CAPTCHA. One human is always required there to do this job. Hence our application will detect this CAPCTHA and then solve it and put forward. So by this, it need not to be stopped and no human involvement is needed in the whole process of web crawling. For this, we are using text recognition and optical character recognition(OCR). Embedding web crawlers with CAPTCHA resolver will improve efficient and speedy performance for uninterrupted web crawling.

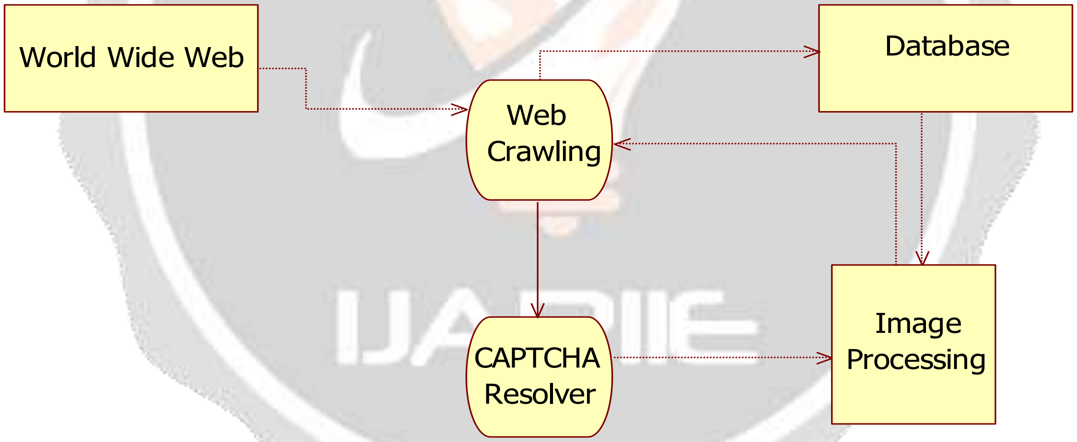

Fig 4 : Proposed System Architecture

Architecture of CAPTCHA resolving system for web crawling is as shown in above figure. Web crawling process will be crawling internet with the URLs stored in database. Database will also be contains crawled data for end users. Whole process will need image processing system when they clash with CAPTCHA for authentication.

4. CONCLUSION

Optical character recognition technique will automate whole process by reducing human interaction. Text recognition method along with OCR technique which includes character detection, image segmentation, and character recognition can be combined with web crawling application to improve performance.

The text resolving techniques can be efficiently utilized in resolving CAPTCHAs for web crawling. Ultimately results in reducing the crawling process time.

5. REFERENCES

[1] Shalini Sharma (Department of Computer Science Shoolini University). "Web Crawler", April 2014 volume 4 IJARCSSE.

[2] Cong Yao, Xiang Bai, Wenyu Liu (Member, IEEE). " A Unified Framework for Multioriented Text Detection and Recognition", IEEE Transactions on image processing, volume 23, no. 11, November 2014.

[3] Miss. Poonam B. Kadam, Mrs. LatikaR. Desai (Computer Department, D.Y.P.I.E.T)."A Hybrid Approach to Detect and Recognize Texts in Images"July-2013Volume 2IJARCSSE.

[4] Julinda Gllavata, Ralph Ewerth and Bernd Freisleben, "A Robust Algorithm for Text Detection in Images", IEEE 2003.

[5] Pavalam S M, S V Kashmir Raja, Felix K Akorl3 and Jawahar M (National University of Rwanda)," A Survey of Web Crawler Algorithms", International Journal of Computer Science Issues, Vol. 8, Issue 6, No 1, November 2011.

[6] Max Jaderberg Karen Simonyan Andrea Vedaldi Andrew Zisserman (University of Oxford)." Synthetic Data and Artificial Neural Networks for Natural Scene Text Recognition", cs.CV 9 December 2014

[7] Alessandro Bissacco, Mark Cummins, Yuval Netzer, Hartmut Neven, "PhotoOCR: Reading Text in Uncontrolled Conditions", Volume 2 IJARCSSE 2013.