Contents

Chapter 1

Introduction

A CAPTCHA or 'Completely Automated Public Turing test to tell Computers and Humans Apart' is a test which is used, as the name suggests, to determine whether or not a user is human. The main application of a CAPTCHA is to ensure that malicious bots are not being used in order to access a website for any given reason, such as spam. A commonly used form of CAPTCHAs are text-based CAPTCHAs, where an image is provided to the user, containing letters, numbers, or a combination of the two. These characters are then distorted and overlapped, making them difficult to read, in order to ensure that bots cannot access the page that has been secured using the text-based CAPTCHA.

1.1 Problem Statement

The problem statement for this project is to create an application that can solve text-based CAPTCHAs using Machine Learning techniques. The reason text-based CAPTCHAs have been chosen for this project is because, despite being widely used in sectors such as banking or e-ticket booking, they are often inaccessible for people with visual impairments, reading disabilities, or people who do not speak English[1]. Hence, there is a need to create an application that can be used in order to help these people access sites that use CAPTCHAs for security.

1.2 Practical Implementation

The aim is to implement this project for practical use in one of the following ways:

- Desktop Application
- Mobile Application
- Web Application

1.3 Approach

1.3.1 Application

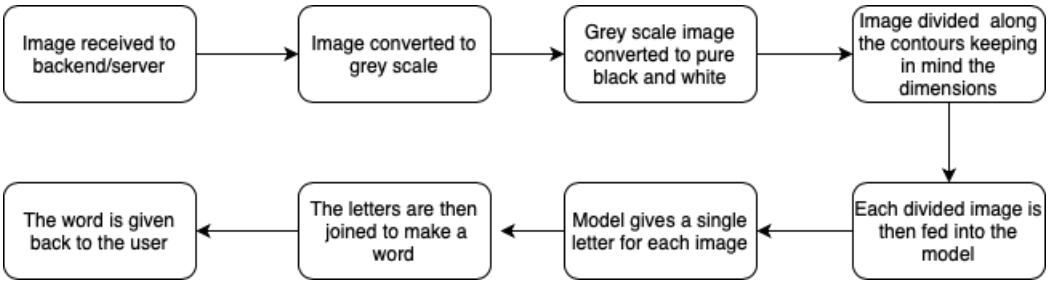
In order to use the application to solve a given text-based CAPTCHA, the user needs to scan the area of the screen that contains the CAPTCHA. The app then pre-processes the scanned image removing all the unnecessary pixels, which is essentially the parts of the image that do not contain the CAPTCHA. This processed image is then sent to the server where it is fed to the pre-trained Machine Learning model. The model returns the deciphered CAPTCHA text, which is then sent back to the user, so that they can input it where required.

1.3.2 Machine Learning Model

Prior to the image being fed into the Machine Learning model, the labels, that is the range of possible characters, are converted to one hot encoded form, so that they can be read by the model. Then, the image scanned by the user is passed through a function that does the following:

- Converts the image to grey scale
- Converts this new image to pure black and white form
- Splits the image based on its contours, and the width and height of each contour
- Inputs each image to the Machine Learning model and receives a single letter or number in return

The returned characters are compiled in order to decipher the complete CAPTCHA. The following flowchart visualises the above process.



Chapter 2

Use Cases

On average, a person takes approximately 10 seconds to solve a text-based CAPTCHA. However, many people are unable to do this for various reasons, and are thus dependent on others in order to perform sensitive tasks such as online banking. This includes people with visual and learning disabilities[2], as well as non-English speakers. These situations have been explored below.

2.1 Visually Impaired People

Text-based CAPTCHAs require the person to be able to see the distorted characters on their screen, interpret and decipher these characters, and return the characters within a given amount of time in order to verify that they are human. However, a person with visual impairments cannot perform this task exclusively on their own. While they may be capable of reading plain text, it can be greatly difficult for them to read distorted text. In this case, an application that solves the text-based CAPTCHA for them would assist them while also ensuring their independence.

2.2 People with Reading Disabilities

People who suffer from dyslexia, for example, find it difficult to read text as it is. For them to read distorted characters on their screen is a virtually impossible task. However, if they had an application that returned the correct

text to them, they could simply copy that text, paste it where they need to enter it, and be able to access the desired website.

2.3 Non-English Speakers

Most of the time, the characters in text-based CAPTCHAs tend to be letters of the English alphabet. A non-English speaker would not be able to differentiate these characters easily. In this case, an application to solve the CAPTCHA for them would be of great use.

Chapter 3

Novelty

The use of Machine Learning in order to solve CAPTCHAs is a heavily explored domain. Despite this, there are certain areas that previous projects have been unable to cover, which this project aims to tackle.

3.1 Competition

As mentioned previously, there have been several attempts in the past to solve CAPTCHAs using Machine Learning. There have also been products created specifically to assist the visually impaired access websites. Some of the prominent competitors have been discussed below.

3.1.1 CAPTCHA Be Gone

CAPTCHA Be Gone is a series of web extensions available as a service in order to solve different types of CAPTCHAs, including text-based CAPTCHAs. According to their website, they detect CAPTCHAs on web pages, solve them, and copy the result to the user's clipboard with the press of a "single keystroke". However, CAPTCHA Be Gone appears to have been inactive since approximately 2017. It is also worth noting that their code is not open source, so they do not elaborate on how the user's privacy is maintained, despite clearly mentioning that on their website. The service is also currently restricted exclusively to the Firefox, Chrome, and Internet Explorer browsers[3].

3.1.2 CAPTCHAsolutions

CAPTCHAsolutions is a Chrome Browser Extension used in order to solve reCAPTCHA v2. However, it was last updated in 2018, and their website appears to have security issues, as the Mozilla Firefox Browser throws a 'Potential Security Risk Ahead' warning while attempting to access the website[4].

3.1.3 Others

While other services to solve CAPTCHAs do exist, these are either limited to a specific platform, such as Chrome or Firefox, or they do not deal with text-based CAPTCHAs.

3.2 Project Novelty

The following sections explain why this project is unique.

3.2.1 Not Platform Specific

As the implementation of the project is in the form of either a desktop or mobile application, there is no need for the user to ensure that they are using a specific browser to access the desired website. They can make use of any platform they like and simply feed the CAPTCHA data into the application, externally.

3.2.2 Variable Text Lengths

So far, text-based CAPTCHA solvers have restricted themselves to a specified number of characters when solving the CAPTCHA. With this project, the aim is to be able to incorporate anywhere between four and six characters when attempting to find a solution for a given text-based CAPTCHA.

Chapter 4

Future Work

This project deals exclusively with text-based CAPTCHAs, and is targeted towards those with visual impairments, reading disabilities, and non-English speakers. For future implementation, these functionalities could be extended to those with hearing disabilities, in order to help them use audio CAPTCHAs. This will continue to be helpful for people who do not speak English as non-English audio CAPTCHAs are extremely rare.

Bibliography

[1] J. W. M. C. Scott Hollier, Janina Sajka, "Inaccessibility of captcha," 2019.

[2] L. Moreno, M. González-García, and P. Martinez, "Captcha and accessibility. is this the best we can do?," 2014.

[3] "Captcha be gone," 2016.

[4] "Captchasolutions," 2018.