UNIVERSIDAD DE BUENOS AIRES

FACULTADES DE CIENCIAS ECONÓMICAS, CIENCIAS EXACTAS Y NATURALES E INGENIERÍA

CARRERA DE ESPECIALIZACIÓN EN SEGURIDAD INFORMÁTICA

Desafíos sobre las nuevas tecnologías de resolución de CAPTCHA
Características que debería poseer CAPTCHA en el futuro para su mantenimiento

DANIEL ALEJANDRO MALDONADO RUIZ
daniel.a.maldonado@ieee.org

TUTOR:
Juan Alejandro Devincenzi

2013 - 2014

Cohorte 2013

DECLARACIÓN

Por medio de la presente, Yo, Daniel Alejandro Maldonado Ruiz, manifiesto conocer y aceptar el Reglamento de Trabajos Finales vigente y me hago responsable que la totalidad de los contenidos del presente documento son originales y de mi creación exclusiva, o bien pertenecen a terceros u otras fuentes, que han sido adecuadamente referenciados y cuya inclusión no infringe la legislación Nacional e Internacional de Propiedad Intelectual.

_____

Daniel Alejandro Maldonado Ruiz

RESUMEN

El presente proyecto de especialización incluye un análisis teórico-técnico de las razones del uso de CAPTCHA desde el punto de vista de la criptografía, y de como con el paso del tiempo la tecnología ha alcanzado el desarrollo suficiente para poder romper la mayoría de CAPTCHA, ya sea en función de ataques o de análisis al algoritmo utilizado; para finalmente plantear los nuevos avances en inteligencia artificial, sobretodo de redes neuronales avanzadas, que podrían, a mediano plazo, eliminar brechas del análisis humano-máquina de los test de Turing.

El análisis comienza estudiando como CAPTCHA se utilizó para resolver ciertos problemas criptográficos basados en los 'problemas difíciles' de la inteligencia artificial como primitivas criptográficas y de los análisis de probabilidad llevados a cabo para comprobar cómo, sin la mediación de una inteligencia artificial altamente desarrollada, CAPTCHA sirve para resolver problemas de autenticación frente a sistemas, además de comprobar que el usuario es una persona.

El avance de la tecnología permitió el desarrollo de herramientas que podían hacerle frente a CAPTCHA para invalidarlo, estar pruebas iban desde ataques de fuerza bruta hasta más complejas herramientas de análisis basado en la extrapolación de la imagen del CAPTCHA para permitir que los algoritmos de detección óptica determinen el contenido, pero siempre basados en la brecha humano-maquina existente.

Además, se realiza una introducción a las nuevas tecnologías de Inteligencia Artificial que, basadas en nuevas redes neuronales avanzadas, han alcanzado la posibilidad de invalidar CAPTCHA al quitarle la posibilidad de diferenciar si el usuario es o no una máquina.

Como parte de este análisis, se presentan recomendaciones de cómo se debe diseñar y mantener CAPTCHA al corto y mediano plazo fortalecido frente a la evolución de la tecnología y que características debería tener.

Finalmente, se presentan las conclusiones acerca del estudio realizado, poniendo especial énfasis en la necesidad de hacer de CAPTCHA un sistema confiable a mediano plazo.

CONTENIDO

ÍNDICE DE FIGURAS

INTRODUCCIÓN

Desde el desarrollo de los sistemas de computación, se consideró necesario diferenciar las respuestas que provenían tanto de un sistema informático como de una persona, para que uno no pueda suplantar al otro. Alan Turing diseño un test que permitía realizar esta comprobación, mediante preguntas que solo los humanos estábamos en capacidad de responder adecuadamente.

Con la masificación de servicios en Internet surgió la necesidad inversa, es decir, la de comprobar que el usuario que estaba accediendo a estos servicios sea en efecto una persona y no un *bot* programado para saturar un servicio o una red. Por este motivo, se desarrolló CAPTCHA, o Prueba de Turing Automática y pública para diferenciar humanos de máquinas, por sus siglas en inglés. La idea era mantener a los programadores maliciosos fuera de servicios masivos de información; pero con el tiempo su uso se extendió hasta ser utilizado como prueba de identificación y autenticación de usuarios, facilitando tareas a criptógrafos y encargados de seguridad.

Conforme el avance de la tecnología, también comenzaron a aparecer métodos cada vez más poderosos para burlar CAPTCHA a través de algoritmos de reconocimiento grafico; sin embargo y a pesar de sus desarrollos, no habían logrado alcanzar el nivel de poder batir CAPTCHA completamente, hasta el desarrollo de la nueva inteligencia artificial. Muy recientes avances en este campo claman haber podido finalmente vencer CAPTCHA mediante una red neuronal avanzada.

El presente trabajo pretende dar una visión de cuál es la fortaleza de CAPTCHA desde el punto de vista criptográfico y de seguridad, de cómo los avances en reconocimiento grafico han permitido vulnerar de alguna manera este sistema y de que características debería poseer CAPTCHA en el futuro para poder seguir siendo seguro frente a cualquier ataque, sea este

algorítmico o de inteligencia artificial; es decir, que aun solo pueda ser resuelto por humanos.

Es importante comprender que aun cuando un sistema pueda ser visto como infalible al corto plazo; la tecnología de inteligencia artificial crece cada vez más y es un desafío mantener los sistemas seguros en un mundo donde las máquinas están tomando control de comunicaciones y servicios con los que nos relacionamos día a día.

SISTEMA VISUAL DE AUTENTICACIÓN HUMANO-MAQUINA O CAPTCHA

CAPTCHA COMO MODELO DE SEGURIDAD [2][3][5][23]

La necesidad de crear textos que sean ilegibles a las computadoras ha formado parte de los estudios de los desarrolladores desde los primeros días de Internet. Ya en 1997 Altavista diseñó un sistema para evitar que *bots* ingresaran *url's* a su motor de búsquedas [1]. Pero no sería hasta 2000 cuando investigadores de la Universidad Carnegie-Mellon e IBM desarrollaron un test para evitar que *bots* desde Internet pudieran acceder y monopolizar servicios para usuarios. Este test se denominó CAPTCHA, o Prueba de Turing Automática y pública para diferenciar humanos de máquinas, por sus siglas en inglés; y fue descrito como tal en una publicación de 2003, tiempo desde el cual se han desarrollado distintos tipos de CAPTCHAS en función a las necesidades de los sistemas que poco a poco iban a invadir la Internet.

CAPTCHA fue diseñado para generar y verificar test que una persona podría fácilmente resolver, pero que a un sistema computacional le sea muy difícil. De acuerdo al diseño original, ni siquiera el generador del test en sí mismo está capacitado para pasarlo. Esta particularidad del diseño ha hecho que efectivamente sirva para crear una brecha entre humanos e inteligencia programable. Esta capacidad de diferenciación ha posibilitado que CAHTCHA sea utilizado como:

- Verificador de votantes en encuestas
- Comprobador de identidades para la creación de correos electrónicos y demás servicios sociales en la Red
- Implementar restricciones de acceso frente a motores de búsqueda.
- Defensas frente al envío masivo de correo basura y malware a través de listas automáticas

- Defensa frente a ataques de diccionario en contraseñas de sistemas informáticos [4]

El concepto público de CAPTCHA implica que el algoritmo de creación e implementación debe ser público, con excepción de la porción de código que maneja la aleatoriedad para su utilización. Este concepto hace que mientras las vulnerabilidades de otros sistemas de seguridad están basadas en la ruptura de un algoritmo especifico, la ruptura de CAPTCHA es un problema específico de Inteligencia Artificial, en donde encontrar un sistema que pueda batir a CAPTCHA eficientemente es encontrar un sistema de Inteligencia artificial altamente eficiente y confiable.

Hablando desde un punto de vista sistemático, no se puede afirmar que CAPTCHA sea imbatible programáticamente dado que un programa, como el funcionamiento del cerebro humano, puede resolverlo. Sin embargo, los creadores presentaron evidencia matemática en la cual se afirma que escribir un programa que pueda romper CAPTCHA es altamente complicado, afirmando que:

> *"Un CAPTCHA es un protocolo criptográfico cuya dificultad asumida está basada en un problema de Inteligencia Artificial". [2]*

Moni Naor, del Instituto Weismann de ciencias de Israel, en 1997 desarrollo la idea de los Test automáticos de Turing [8], que dio pie a los primeros desarrollos del sistema presentados por Altavista; y en general este sistema fue útil hasta el desarrollo de los sistemas de reconocimiento óptico de caracteres (Optical Character Recognition u OCR), fruto de lo cual se desarrolló CAPTCHA como un Test de Turing que no esté basado en la dificultad presentada por los OCR's, que es lo que está activo actualmente.

1.1 DEFINICIONES PRINCIPALES

Para definir a CAPTCHA[1], se asume que existe una distribución de probabilidad definida como C, y su soporte modular definido como [C]. Si $P(\cdot)$ es un programa probabilístico, se escribirá $P_r(\cdot)$ al programa determinístico resultante cuando P utilice entradas aleatorias r.

Sea (P,V) son un par de programas probabilísticos interactuando entre sí, si se obtiene una salida de V luego de la interacción entre P y V con entradas aleatorias u_1 y u_2, se asume que la interacción final se describe como $\langle P_{u_1}, V_{u_2} \rangle$. Un programa V es llamado *test* si por cada P y u_1, u_2 la interacción entre P_{u1} y V_{u2} finaliza y $\langle P_{u_1}, V_{u_2} \rangle \in \{aceptar, rechazar\}$. Se llama V al verificador y *tester* y a cualquier P que interactúa con V la prueba.

Definición 1: el éxito de un programa A sobre un test V como:

$$Exito_A^V = Pr_{r,r'}[\langle A_r, V_{r'} \rangle = aceptar]$$

Se asume que A tiene conocimientos precisos de cómo funciona V, exceptuando r', el factor de aleatoriedad.

A partir de esta definición resulta claro que CAPTCHA es un test V que los humanos pueden resolver casi al 100%, y es la causa por la cual es difícil escribir un programa que lo resuelva teniendo tanto éxito sobre V.

Definición 2: un test V se dice que es $(\alpha, \beta) - humano\ ejecutable$, si al menos una porción α de la población humana tiene un éxito sobre V mayor que β.

[1] Conceptos matemáticos y probabilísticos expuestos en la publicación original de 2003 [2] sobre la que se basó CAPTCHA como un modelo de primitiva de seguridad basado en Inteligencia Artificial.

Aunque la condición "$(\alpha, \beta) - humano\ ejecutable$" solo puede ser probada empíricamente, y depende de las habilidades linguisticas y sensoriales de los involucrados.

Definición 3: Un problema de Inteligencia Artificial se considera como un triplo $P = (S, D, f)$, donde:

. S es un conjunto de posibles instancias

. D es una distribución de probabilidad sobre el problema del conjunto S

. $f: S \rightarrow \{0,1\} *$ como las respuestas instanciadas.

Siendo $\delta \in (0,1)$, se requiere que para un $\alpha > 0$ fraccion de humanos H, dando como resultado:

$$Pr_{x \leftarrow D,r}[H(x) = f(x)] > \delta.$$

Definición 4: Un problema de Inteligencia artificial se dice que es $(\delta, \tau) - resuelto$ si existiese un programa A, corriendo durante al menos un tiempo τ en cada entrada de S, tal que:

$$Pr_{x \leftarrow D,r}[A_r(x) = f(x)] \geq \delta$$

Con A siendo una solución (δ, τ) de P, donde P es considerado como un problema difícil de Inteligencia artificial.

Definición 5: Un $(\alpha, \beta, \eta) - CAPTCHA$ es un test V del tipo $(\alpha, \beta) - humano\ ejecutable$ que cumple con la propiedad de existir un $(\delta, \tau) - dificil$ problema de Inteligencia Artificial P y un programa A, tal que si B tiene un éxito mayor que η sobre V entonces AB es una solución (δ, τ) de P, donde V tiene que ser necesariamente un programa de código abierto.

Cabe recalcar que los problemas del tipo (S, D, f) no abarcan todos los problemas de Inteligencia Artificial, sino una aproximación que facilita el estudio de los problemas difíciles de Inteligencia Artificial como tal, al

comprobar que CAPTCHA es uno de ellos. Estos problemas están definidos si una porción de la población puede resolverlos, aun si el tiempo que les tome sea elevado. Lo que hace que CAPTCHA sea considerado un problema difícil adicionalmente es que su tiempo de resolución debe ser muy corto, caso contrario sería inútil para todo fin práctico.

Estas definiciones ayudan a construir CAPTCHA como una primitiva de seguridad a los problemas de Inteligencia Artificial, lo que hace que se deba diferenciar entre estas y las primitivas criptográficas habituales, ya que la Inteligencia Artificial, como se ha definido, no tiene asíntotas temporales con las cuales trabajar. Existe una brecha entre las capacidades humanas y las computacionales que con cada descubrimiento nuevo se acorta, acercando a la Inteligencia Artificial a tener las mismas características cognitivas y de razonamientos que la humana. Esta brecha es la que potencialmente se usa como una primitiva de seguridad. No todos los problemas difíciles de Inteligencia Artificial pueden ser usados para construir CAPTCHA, ya que se necesita que estos sean automatizados para generar el problema de su resolución.

Las definiciones [2] implican que el atacante debe conocer exactamente el funcionamiento de CAPTCHA con excepción de su generador de aleatoriedad. Esto hace que la generación de CAPTCHA mas desafiantes se vuelva complicada, ya que no se pueden permitir CAPTCHAS que basen su seguridad en una base de datos o una porción de código.

A partir de una aproximación por repeticiones seriales, los creadores definieron una brecha arbitrariamente cercana al 100% entre el éxito de los humanos frente a los programas informáticos contra un CAPTCHA, la cual afirma que siendo V un $(\alpha, \beta, \eta) - CAPTCHA$, y que V_k^m es el test que resulta de repetir V m veces en series, cada una con un patrón de aleatoriedad diferente y aceptando las respuestas solo si el testeante pasa V mas de k veces. Entonces para cada ε > 0 existe un m y un k donde $0 \leq k \leq m$ tal que V_k^m es un $(\alpha, 1 - \varepsilon, \varepsilon) - CAPTCHA$.

Generalizando, tenemos que $m = O\left({}^{1}/_{(\beta-\eta)^2}\, ln({}^{1}/_{\epsilon})\right)$, y habitualmente mucho mas pequeño. Desde que CAPTCHA esta desarrollado para uso humano, es necesario que se encuentre un m lo más pequeño posible, lo que se logra mediante esta optimización de la aproximación:

$$min_m\left\{\exists k: \sum_{i=K+1}^{m}\binom{m}{i}\beta^i(1-\beta)^{m-i} \geq 1-\epsilon \& \sum_{i=0}^{k}\binom{m}{i}\eta^i(1-\eta)^{m-i} \leq \epsilon\right\}$$

Mientras más se amplifique la brecha se puede aumentar el parámetro de seguridad de CAPTCHA: Si el mejor programa computacional tiene un éxito de 0.10 contra un CAPTCHA, el probar al mismo programa dos veces reducirá sus probabilidades en 0.01

Las suposiciones realizadas para construir estos modelos matemáticos están basadas en las mismas suposiciones sobre las cuales se basan los protocolos criptográficos para garantizar la dificultad al construir CAPTCHA, lo cual genera dos grupos de problemas de Inteligencia Artificial.

En general, una imagen no es más que una matriz de [h,w] pixeles, donde un pixel está definido como un triplo de enteros (R,G,B), donde $0 \leq R, G, B \leq M$ para una constante M definida; asi como una transformación de imagen es una función que toma la matriz original de la imagen y nos devuelve una matriz diferente según el tipo de función aplicada.

Siendo I una distribución de imágenes, y T una distribución de transformadas y asimuendo que $i, i' \in [I] \forall\, i \neq i'$ se tiene que $T(i) \neq T'(i')\ \forall\ T, T' \in [T]$, lo que define a:

Grupo de problemas ($\boldsymbol{P1}$)**:** Está definido por el siguiente experimento:

Para una imagen $i \leftarrow I$ y su correspondiente transformada $t \leftarrow T$, es decir $t(i)$, se define $P1_{I,T}$ que consiste en escribir un programa tomando $t(i)$ como entrada e i como salidas. Formalmente se lo puede definir como $S_{I,T} = \{t(i): t \in [T]\ \&i \in [I]\}$, $D_{I,T}$ es la distribución correspondiente a $S_{I,T}$ que resulta de desarrollar el experimento y $f_{I,T}: S_{I,T} \rightarrow [I]$ seria escrito de mejor manera como $f_{I,T}\big(t(i)\big) = i$. Entonces $P1_{I,T} = \big(S_{I,T}, D_{I,T}, f_{I,T}\big)$

Grupo de problemas ($\boldsymbol{P2}$)**:** Además de las distribuciones anteriores I y T, definimos L como un conjunto de etiquetas. Sea $\lambda: [I] \rightarrow L$ definido como la etiqueta de una imagen, entonces para el grupo de problemas $S_{I,T} = \{t(i): t \in [T]\ \&i \in [I]\}$ y la distribución de instancias $D_{I,T}$ asignada por escoger $i \leftarrow I$ y $t \leftarrow T$ existe entonces $g_{I,T,\lambda}$ tal que $g_{I,T,\lambda(t(i))} = \lambda(i)$. Por lo tanto $P2_{I,T,\lambda} = \big(S_{I,T}, D_{I,T}, g_{I,T,\lambda}\big)$ consiste en escribir un programa que tenga por entradas $t(i)$ y por salidas $\lambda(i)$.

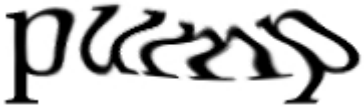
Estas definiciones permiten relacionar ambos grupos de problemas desde el punto de vista de cada uno, ya que una solución (δ, τ) en A para una instancia de $P1$ tambien se obtiene una solución $(\delta', \tau + \tau')$ para una instancia de $P2$ donde $\delta' \geq \delta$ y $\tau' \leq log|[I]|$, que es el tiempo que tomaría computar λ. Es relativamente inútil para δ muy pequeños, por lo que se considera más bien un conjunto muy limitado de etiquetas. Al revés, $P1$ puede ser visto como una solución especial de $P2$ mantiendo λ y asumiendo que $L = [I]$; lo que mostraría que $P1$ y $P2$ son familias isomorficas, lo cual facilita la distinción en aplicaciones que no reconocen o no manejan etiquetas.

A partir de, es posible construir una solución δ_T, τ_T para $P1_{I,T}$, donde $\delta_T = max\big\{Pr_{j \leftarrow I}[j = i]: i \in [I]\big\}$ y τ_I como el tiempo que le toma para describir un elemento de $[I]$, siempre suponiendo una imagen con la mas alta probabilidad en I. De igual manera, se puede construir una solución $\delta_{T,\lambda}, \tau_{T,\lambda}$

para $P2_{I,T,\lambda}$, donde $\delta_{I,\lambda} = max\{Pr_{j\leftarrow I}[\lambda(j) = \lambda(i)]: i \in [I]\}$ y $\tau_{I,\lambda}$ como el tiempo que le toma para describir una etiqueta de $[L]$. Aunque se tienen todas las posibilidades, para la construcción de sistemas se restringen las soluciones de $P1_{I,T}$ donde $\delta > \delta_I$; y soluciones de $P2_{I,T,\lambda}$ donde $\delta > \delta_{I,\lambda}$. [2]

Las variables consideradas en ambas familias proporcionan la seguridad de que los humanos van a ser capaces de resolverlos en un determinado tiempo, sean estas imágenes u otro tipo de objetos reconocibles bajo determinada transformación, como audios y pequeñas animaciones distorsionadas.

La mayoría de CAPTCHAs diseñados son distorsiones de letras, tal como los considerados para este estudio, a partir de los problemas contemplados de Inteligencia Artificial y pueden ser derrotados a partir de las soluciones de $P2$. Para demostrarlo, se considera como W a un conjunto de imágenes de palabras con diferentes tipografías. Siendo I_W una distribución de W, T_W la correspondiente distribución de transformadas y λ_W como el mapa de bits de la imagen que contiene el texto. Con estos datos, una solución $P2_{I_W,T_W,\lambda_W}$ es un programa que puede vencer un CAPTCHA de letras distorsionadas como el de la imagen 1. Entonces obtener el texto de la imagen en cuestión será una instancia de $P2$, aunque fácilmente derivable a una de $P1$, como vimos anteriormente. La lectura de un texto ligeramente distorcionado es uno de los problemas a los que se siguen enfrentando los OCRs.



1.1. CAPTCHA de Yahoo!

Sin embargo, $P1$ y $P2$ son problemas bastante generales, y textos ligeramente distorsionados son de alguna manera instancias sencillas de estos problemas. Lo que no es el caso de textos que se reducen a emparejar

26*2+10 caracteres. La dificultad de $P1$ y $P2$ radica en T. Particularmente, debería ser computacionalmente inviable para enumerar todos los elementos de $[T]$, dado que I será normalmente tal que la enumeración de $[I]$ es factible. Entonces interesan mas soluciones de (δ, τ) donde $\tau \ll |[T]|$ mientras que $\tau > |[I]|$ puede algunas veces ser aceptable. Adicionalmente al tamaño del conjunto de transformadas, el tipo de las transformadas debe poder sortear comprobaciones simples, como comparaciones de color, dominios de frecuencia, etc.

A partir de las instancias de $P1$ y $P2$ como parámetros, se han intentado buscar soluciones basadas en mayor o menor manera en Inteligencia Artificial, con determinado éxito según como los proveedores hayan implementado CAPTCHA en sus sitios y sistemas de seguridad, dado que, a pesar de que los parámetros descritos se definieron con CAPTCHA, el avance real de la Inteligencia Artificial y de los OCR solo despunto en los últimos años, a través de algoritmos de redundancia de análisis de patrones visuales, consiguiendo resultados medibles para cierto tipo de CAPTCHA.

En Internet el tipo de CAPTCHA más utilizado se conoce como PIX, y que está directamente basado en una instancia de $P2_{I,T,\lambda}$. Por definición, una instancia de PIX es un tuplo $X = (I, T, L, \lambda, \tau)$, y se verifica si V consigue que $i \leftarrow I$ y $t \leftarrow T$. V entonces envía a P el mensaje $(t(i), L)$, y asigna el temporizador para τ. P responde con una etiqueta $l \in L$, que será aceptada por V si $l = \lambda(i)$ y el temporizador no ha caducado; dando lugar al teorema sobre el que se basa la mayoría de CAPTCHA en Internet:

"Si $P2_{I,T,\lambda}$ es $(\delta, \tau) - dificil$ y $X = (I, T, L, \lambda, \tau)$ es $(\alpha, \beta) - humano\ executable$, entonces X es un CAPTCHA del tipo (α, β, δ)". [2]

FACTORES DE VULNERABILIDAD DE CAPTCHA [6][7]

El teorema establecido anteriormente, si bien define los parámetros de un CAPTCHA práctico, solo define el cómo debe ser escrito el programa, no con qué características debe ser presentado al usuario para que sea más o menos vulnerable a los nuevos sistemas de OCR.

En años recientes, y en función a estudios realizados sobre el sistema de seguridad de CAPTCHA[2] determinaron que con un adecuado sistema de segmentación de la imagen a ser atacada, es posible encontrar y vulnerar determinados tipos de sistemas según su tipo de construcción, caracteres y distorsión; sin embargo, la segmentación por sí sola no es suficiente en un entorno variable de imágenes en internet.

Estos estudios determinaron que para poder vulnerar CAPTCHA y romperlo según su complejidad se requieren de 5 pasos genéricos:

- Pre-Procesamiento
- Segmentación
- Post-Segmentación
- Reconocimiento
- Post-Procesamiento

Por segmentación estaba entendida la separación de una secuencia de caracteres en caracteres individuales, y por reconocimiento la identificación de los mismos, pero los nuevos sistemas han incorporado arreglos adicionales a los caracteres distorsionados como tal, que interfieren con los pasos básicos, por lo que los nuevos pasos previos aseguran la mayor normalización de los caracteres para su posterior reconocimiento.

[2] En este trabajo no se consideran los ataques manuales a CAPTCHA, por parte de personas contratadas para este propósito.

Finalmente el post-procesamiento puede mejorar la precisión de los caracteres para facilitar, por ejemplo, las funciones de aprendizaje de los sistemas atacantes, o determinar si los conjuntos de caracteres son formados al azar o si corresponden a palabras reales en un idioma determinado.

Este procedimiento se derivó de un estudio en el que se comprobaba la dificultad que tenían las personas para resolver CAPTCHAs que se utilizan en los sitios más visitados según Alexa[3], además de sistemas proporcionados por recaptcha.net y captchas.net. Este estudio determinó que los humanos, en promedio, solo acertaban a resolver CAPTCHA un 71% de las veces[4] [6], que las personas que no tienen un nivel fluido de inglés (idioma en el que se presentan la mayoría de CAPTCHA) tienen mayor problema que los que son angloparlantes, y que, con la edad, las personas se vuelven más precisas en resolver un CAPTCHA, aunque sacrificando la velocidad de resolución. Mientras más respuestas diferentes posea un CAPTCHA para los testadores, mayor es la posibilidad de fallo, y por tanto, mayor dificultad presenta para las personas.

Sin embargo, se demostró que lo que era difícil de leer para una persona no era necesariamente difícil de interpretar para un sistema de reconocimiento visual y viceversa. Los algoritmos planteados para resolver CAPTCHA se hicieron de acuerdo a las técnicas que usan las personas, aunque considerando aquellos que tenían mayor número de respuestas diferentes.

Para evaluar la efectividad real de los ataques propuestos, el primer paso es medir la precisión, es decir, la fracción de CAPTCHA que pueden ser resueltas adecuadamente. Cada sistema, sin embargo, según su constitución y diseño presentara diferentes resultados a los análisis.

[3] http://www.alexa.com/topsites (visto el 10/04/14)

[4] A pesar de que este estudio consideró CAPTCHAS de audio, este trabajo está enfocado en CAPTCHA de lectura, por lo que los resultados obtenidos no serán tomados en cuenta.

Adicionalmente, dependerá de las restricciones propias del sitio en cuestión y de la confianza del atacante para resolver determinado CAPTCHA sin cambiarlo por otro más fácil.

La cobertura se define como la cantidad de CAPTCHA que el atacante está dispuesto a resolver, y la precisión como la cantidad de CAPTCHA respondidos correctamente. En los apartados anteriores, y a partir del teorema señalado, se estableció como CAPTCHA exitoso a aquel que no puede ser vulnerado más de 1 de 10000 intentos, es decir en un porcentaje del 0.01%. En la realidad se puede considerar un CAPTCHA vulnerado si la precisión puede alcanzar al menos el 1%. Es decir, durante un experimento completo con un grupo de al menos 1000 CAPTCHA, si al menos se han podido resolver 10 que no han sido revisados previamente para designar un sistema como inseguro.

El proceso definido para vencer un CAPTCHA según experimentación previa es el de asignar un tipo de segmentación específica para cada sistema con un sistema de aprendizaje basado en un OCR. Primero, el programa pre-procesa la imagen para que pueda ser fácil de analizar, removiendo colores solidos de fondo y aplicando filtros reductores de ruido. Luego, el programa trata de separar la imagen en trozos que contengan un carácter, o un porcentaje muy elevado de uno, finalmente, estos trozos son vectorizados para análisis o ingresados en una red neural para determinar qué carácter está almacenado en cada uno de los trozos y dar una respuesta del CAPTCHA analizado. Los experimentos mencionados han demostrado que mientas más fácil sea segmentar un CAPTCHA más vulnerable es. Los diferentes CAPTCHAs en el mercado han establecido sus propias técnicas anti-segmentación para evitar que se puedan separar en caracteres y analizar el contenido, con mayor o menor éxito, ya que solo son efectivas si se han diseñado e implementado adecuadamente, además que deben ser implementados en todas las capas de la imagen, a fin de crear un esquema seguro.

El reconocimiento de caracteres es un problema de fondo dentro del análisis para encontrar una forma eficiente de romper CAPTCHA. La base de datos de caracteres manuscritos del Instituto Nacional de Estándares y Tecnología de Estados Unidos (Mixed National Institute of Standards and Technology, o MNIST, por sus siglas en inglés) es la principal fuente de entrenamiento para sistemas basados en reconocimiento de caracteres distorsionados y de cómo reconocerlos eficientemente.

Cuando CAPTCHA no puede ser segmentado y se debe hacer un reconocimiento sobre toda una imagen, es posible realizar aproximaciones alternativas con descriptores de imagen complejas, como SURF y SIFT[5], que funcionan invariablemente ante la rotación de caracteres y son muy estables contra la distorsión, gracias a su uso de 'puntos de interés' que permiten una aproximación mucho más estable y rápida. Como el número de 'puntos de interés' no puede ser normalizado porque varía según cada tipografía, no se pueden aplicar clasificadores para hacer más eficiente el reconocimiento como tal.

Para aumentar la seguridad de anti-segmentación, se utiliza también fondos para las imágenes a ser descifradas que contribuyan a la confusión de la imagen general al mezclar el texto con las características del fondo, ya sean estos imágenes complejas o colores similares al de los caracteres o aumentando el ruido de la imagen.

La idea de usar fondos complejos parte de que usando sus líneas/formas se puede contribuir a la confusión de un detector de caracteres que impida la segmentación de los mismos. A pesar de esto, estudios anteriores de detección demostraron que usualmente este tipo de defensa es insegura, muchos CAPTCHAs en internet la siguen usando, valiéndose de fondos aleatorios generados en función de patrones pre-establecidos que

[5] SURF: Características robustas de alta velocidad o Speeded-Up Robust Features
SIFT: Transformada de Características de Escala Invariante o Scale Invariant Feature Transform

buscan evitar que los atacantes consigan el patrón de uso de las imágenes, mientras que mantienen destacados los caracteres para que los usuarios puedan resolverlos.

El problema radica en que las imágenes utilizadas como fondos no deben ser de alta calidad para que no comprometan el peso del programa como tal, por lo que necesariamente y aun a pesar de su aparente complejidad, el número de colores utilizados en cada una se mantiene finito; sin mencionar el color con el que se destaca cada carácter. A partir de estas características los analizadores remueven todos los pixeles que sean diferentes a los del color de los caracteres, si se ha comprobado que estos tienen el suficiente número de pixeles para ser considerado como tal. Otro uso de los fondos es el de asignarles un color muy similar al de los caracteres para que estos puedan perderse en el fondo. El problema está en que muchos colores que para las personas no son tan cercanos, si lo son a nivel del espectro RGB[6]; y viceversa. Sin embargo, una forma efectiva de contrarrestar consiste en romper el patrón con una diferente representación de colores que este más cercana a la percepción humana, y luego binarizar el CAPTCHA basados en el matiz o en la saturación del mismo para facilitar el trabajo de los analizadores.

Añadir ruido aleatorio como técnica de confusión y anti-segmentación está considerado como la manera más eficiente de asegurar un CAPTCHA, pero el mismo tiene que ser del mismo color de los caracteres o fácilmente los reconocedores de patrones pueden eliminarlo. Para eliminar el ruido de un CAPTCHA, se han utilizado varias técnicas de filtrado con los años, pero la más eficiente es usar un Campo Aleatorio de Markov, también conocido como algoritmo Gibbs.

Gibbs es un algoritmo iterativo que computa la carga de color de cada pixel, o su energía, basándose en los pixeles que tiene alrededor para

[6] RGB (Red, Green, Blue): modo de representación de imágenes en la cual cada pixel es una combinación de los tres colores mencionados, cada uno representado en 8 bits.

remover los pixeles cuya energía este por debajo de un umbral predeterminado, terminando cuando no hay más pixeles por remover. La energía de cada pixel es computada resumiendo los valores en una escala de grises de los 8 caracteres circundantes y dividiéndola para 8. Este tipo de algoritmo hace que la mayoría de fondos utilizados, con o sin ruido, sean inseguros en un ambiente de Internet.

Otra técnica para evitar la segmentación es el uso de líneas que crucen los caracteres que conforman el CAPTCHA, ya sea con líneas pequeñas a través de determinados caracteres o líneas largas que atraviesan todo el CAPTCHA.

Para cuando se utilizan líneas cortas entre los caracteres, el estándar es utilizar una segmentación basada en histogramas que proyecta a los pixeles de CAPTCHA dentro de coordenadas X o Y. Esta aproximación funciona porque las regiones donde hay mayor densidad de caracteres se crean picos en el histograma. El problema consiste en la determinación del umbral y el tamaño de las ventanas alrededor del mismo; para evitar las líneas pequeñas es mucho más efectivo utilizar el algoritmo de Gibbs con reconstrucción de caracteres y no requiere de una afinación tan intensiva como los histogramas.

Para cuando se utilizan líneas largas es común utilizar líneas que tengan el mismo grosor de los segmentos de los caracteres, lo que hace que no sean vulnerables a las técnicas anti-ruido, como Gibbs, pero es susceptible frente a filtros buscadores de líneas, como las transformadas de Hough o la detección de borde de Canny, que encuentran líneas sobre un texto de manera muy eficiente. La dificultad radica en no deformar los caracteres cuando las líneas son removidas, lo que se logra comprobando los pixeles circundantes a uno con posibilidad de remoción para determinar si en efecto debe ser removido o no. Esta técnica se dificulta cuando los caracteres son huecos, lo que hace que queden deformados más allá de

toda recuperación cuando las líneas que han sido superpuestas son removidas.

Finalmente, la unión de caracteres o colapso es considerada como la mejor y más segura técnica de anti-segmentación, dependiendo solamente de si su uso al generar las imágenes es correctamente aplicado. Esta es la razón por la que se distinguen dos tipos de colapso: uno donde el atacante puede predecir la segmentación de caracteres a pesar de encontrarse colapsados y otro donde no se puede predecir y el atacante debe realizar ataques de ‘fuerza bruta’ al sistema.

El colapso predictivo se da cuando a pesar de los caracteres están juntos, el atacante aún puede definir donde están las separaciones de cada uno para la segmentación, sobre todo si los caracteres son lo suficientemente regulares en su construcción o si el número de los mismos viene predefinido. Conocido como segmentación oportunista, ya que se basa en información indirecta para realizar su trabajo. Es la más vulnerable de las uniones de caracteres.

El colapso no predictivo se da cuando el número de caracteres no es conocido por defecto y cada carácter tiene su propio tamaño y tipografía, lo que impide obtener patrones definidos para segmentar y obliga a los atacantes a tratar de reconocer el CAPTCHA en su totalidad sin segmentación. Ante este tipo de CAPTCHA es que se han utilizado plantillas de caracteres manuscritos a partir de la MNIST o el uso de redes neurales para el reconocimiento.

Las características mencionadas están en más o menos todos los CAPTCHA que existen en Internet, probando la validez de su construcción, pero su avance puede verse vulnerado por la evolución de la Inteligencia Artificial por sí misma, como se explicará a continuación.

DESARROLLO DE REDES NEURONALES AVANZADAS

REDES NEURONALES CONVOLUCIONALES

[13][14][15][19][20][21][24]

Las operaciones que para el cerebro humano son triviales no lo son para las maquinas, ya que el funcionamiento secuencial y paralelo del procesamiento del cerebro requiere obligatoriamente un elevado número de interconexiones masivas y nodos de interpretación, en este caso neuronas y sinapsis.

Los sistemas que más se han acercado a emular el sistema de procesamiento del cerebro son las redes convolucionales con sistemas y algoritmos de entrenamiento basados en gradiente. Sin embargo, la desventaja de estas redes está en que su emulación de los procesos cerebrales se limita a reconocer fotogramas, es decir, en imágenes estáticas escaneadas y no están diseñados para procesar eventos en tiempo real. De hecho, se ha comprobado que estos sistemas son más eficientes en reconocimiento que en segmentación, ya que el número de conexiones necesarias aun es una limitante física al momento de su construcción.

En este sentido, en 2005 un grupo de investigadores de Microsoft [24] determino que cuando se ha resuelto el problema e la segmentación, se puede asumir directamente que también se ha resuelto el problema del reconocimiento. Para este objetivo, se propusieron comprobar la eficiencia de una red convolucional en el reconocimiento de caracteres singulares y evaluar su rendimiento frente a la capacidad de reconocimiento de un ser humano. Por ello diseñaron un experimento en el cual generaron caracteres aleatorios y los deformaron según secuencias computadas para que puedan ser consideradas como parte de la base de datos de la MNIST y por intervalos se evaluaron según su complejidad por la red neuronal y por los humanos, y comprobaron que, en este escenario controlado, las redes neuronales son mucho más eficientes en el reconocimiento que los

humanos, comprobando además que el reconocimiento no es un problema una vez que se ha salvado el paso de la segmentación.

Este experimento aún se basaba en imágenes estáticas para realizar las comprobaciones con las redes neuronales, hasta que en 2011, un grupo de investigadores de la Universidad de Sevilla [21] diseño una red convolucional bio-inspirada, a través de chips convolucionales programables que manejan el protocolo AER (Representación de Eventos mediante Direcciones, o Adress-event representation) para el reconocimiento ya no de fotogramas, sino de eventos en tiempo real.

Este sistema de 6 capas emula el funcionamiento de las neuronas donde cada neurona en un mapa específico está conectada únicamente con neuronas de la capa siguiente. Esta característica hace que para cada entrada de un evento, en este caso letras manuscritas que son similares a las almacenadas en la base de datos del MNIST, se separan por etapas, cada una con diferentes imágenes de salida conocidas como ‘mapas de características’, que están compuestas de los mencionados mapas de neuronas.

Para su experimento realizaron una simulación de sistemas AER donde multiplexaron cada evento de entrada en 6 canales conectados a la primera capa de módulos AER, que implementan filtros de Gabor para el análisis. A la salida de la primera capa se envían a la siguiente donde se submuestrean y se re-codifican las direcciones de las entradas para obtener unas que las siguientes capas puedan reconocer. Cada una de estos canales almacena la dirección especifica de una parte del evento y se replican en 4 canales adicionales, que envían la información a la tercera capa que no es más que una nueva estructura de convolución con 6 puertos de entrada, donde la máscara de convolución que corresponde a cada entrada se añade a la dirección previamente codificada en la matriz de pixeles.

En función de umbrales y tiempos definidos, los eventos de salida son enviados a la capa siguiente para su procesamiento mientras que la previa es reseteada. Estos tiempos refractarios son utilizados para emular las no-linealidades de los modelos habituales de redes neuronales. En la cuarta etapa se vuelve a submuestrear los eventos resultantes y a enviarse a la quita etapa donde son replicados 10 veces y enviados a una sexta capa compuesta por analizadores i&f (integración y disparo), que presentan resultados positivos o negativos en función de la correspondencia entre la entrada y su salida. El uso de las no-linealidades hace que se presenten solo valores positivos en el que la salida corresponda al digito de entrada.

A través de esta configuración de los chips AER y de las no-linealidades lograron una tasa de reconocimiento del 93%, lo que minimiza los tiempos de respuesta al obtener un primer resultado a la salida del analizador, que en este caso fue de 4.3 [μs]. También comprobaron que cuando el flujo de entrada variaba entre dígitos diferentes el tiempo para una respuesta positiva luego de la transición es de 22.4 [μs].

Sin embargo, y a pesar que la evolución del hardware hacia nuevos sistemas CMOS/noCMOS, aún no ha logrado resolver el problema de la correcta segmentación de caracteres para su reconocimiento.

Adicionalmente, y fruto de StreetView, Google se hizo con miles de números de direcciones de todas las calles a las cuales su sistema de Maps ha llegado. A través de ReCaptcha, los usuarios han podido descifrar muchos de los números de direcciones, y, siguiendo este patrón, ha diseñado una red neural para leer los números adquiridos por Street View sin necesidad del componente humano [20].

Esta red realiza un reconocimiento completo de la imagen, emulando el funcionamiento del cerebro humano en lugar de dividirlo en números iguales para su procesamiento. Esta funcionalidad, entrenada a través del Conjunto de Datos de Números de Casas de Street View (SVHN, o Street

View House Numbers, por sus siglas en ingles) en algo más de 6 días tiene un nivel de precisión de al menos el 96% frente a la precisión conseguida por los seres humanos a través de ReCaptcha, que es del 98%, el cual los ingenieros de Google se han puesto como el umbral de éxito.

Para su cometido emplearon una red convolucional avanzada basada en la implementación de DistBelief (2012) que permite entrenar redes neuronales avanzadas distribuidas para analizar imágenes de alta calidad. A través de arquitectura muy avanzada han logrado mejorar el rendimiento con 11 capas ocultas.

La particularidad del uso de esta red está en que las fotografías tomadas por Street View, aunque no presentan letras colapsadas, según el ángulo de fotografía, la luz, el tiempo meteorológico y la forma de la nomenclatura de cada casa hacen que su análisis no sea tan simple, sobre todo si se lo va a realizar sin segmentación.

Para conseguir el nivel mencionado de precisión, realizaron experimentos preliminares con caracteres únicos desplazados aleatoriamente para entrenar a la red en el reconocimiento a través de aumentar la base de datos de estudio en un 30%, aumentando las variables, como la escala de las imágenes, haciendo que un digito pueda entrar en un contenedor definido, mientras que más caracteres debieron ser reducidos para que puedan caber en los contenedores definidos. Sin realizar este previo entrenamiento, comprobaron que se perdia casi la mitad de precisión en puntos porcentuales.

La arquitectura del sistema consiste en 8 capas convolucionales ocultas, una adicional conectada localmente y dos conectadas densamente entre sí, todas conectadas de tal modo que la salida de la una invariablemente iba a la siguiente sin perder conexiones intermedias. Mientras que la primera contiene las unidades de estudio completas las siguientes están constituidas de rectificadores, dando lugar a 3702 unidades

entre todas las capas conectadas todas ellas normalizadas y configuradas para preservar su tamaño de representación.

Google aspira con estos avances en inteligencia artificial poder leer y almacenar direcciones de una manera más eficiente, y extrapolar los resultados a señalética urbana y anuncios publicitarios. Todos estos avances, sin embargo, siguen sin resolver el problema de la segmentación. Tal como se expondrá a continuación, una startup afirma haber resuelto el problema con otro tipo de red neuronal.

REDES NEURONALES RECURSIVAS CORTICALES Y EL ANUNCIO DE VICARIUS [9][10][11][12][16][17][18][22][24]

El desarrollo de las redes corticales está íntimamente relacionado con el modo en que neurológicamente los receptores y transmisores envían las señales que los ojos reciben para que sean interpretados en los centros cerebrales de reconocimiento.

Este es un paso intermedio entre la neurología y la Inteligencia Artificial, ya que intenta simular el método en que la experiencia visual consiente a través de áreas corticales interconectadas por medio de un flujo combinado de estímulos y el resultado integral de las respuestas a estos estímulos y de toda la actividad subyacente en estas áreas corticales de la corteza occipital, en donde se encuentran almacenados los receptores de visuales, es decir, una red neuronal recursiva.

El principio de una Red Neuronal Cortical es el desarrollo de un *framework* sistemático para determinar el funcionamiento de los bucles recurrentes de las unidades corticales que funcionan como foto receptores y foto transmisores y que son básicas para producir la experiencia de fenómenos visuales consientes asimilados por el sistema nervioso.

Parte del estudio neurológico de estas redes y la parte más difícil de emular por parte de los sistemas computacionales son las representaciones neuronales del sentido de si-mismo como filtro de las representaciones sensoriales que las redes corticales necesitan como inicio de la experiencia consiente y de cómo los bucles de información sobrealimentada en los niveles corticales añaden experiencias sensoriales. Sin embargo para su aplicación solo se toman aproximaciones de su funcionamiento el cual pueda ser emulado mediante algoritmos.

Las redes corticales están basadas en un modelo de aprendizaje de maquina conocido como Memoria temporal jerárquica (Hierarchical Temporary Memory o HTM, por sus siglas en inglés) [25] que recoge estas aproximaciones en el funcionamiento de los sistemas occipitales y las propiedades algorítmicas resultantes del neo córtex como un método de presentar un modelo mucho más complejo de la actividad sensorial cerebral.

La idea de su implementación se basa es someter a la red a variados flujos sensoriales en vez de a estímulos propios de respuestas programadas, dando lugar a una red matricial claramente dividida en sub-espacios donde se almacena y procesa la información recibida en función de la ‘memoria’ que contenga la red, es decir, el entrenamiento al que ha sido expuesta previamente.

La jerarquía de estas redes está determinada en función del tiempo y se establece en tres niveles de jerarquía compuestos por nodos específicos. Mientras más elevada la jerarquía, requiere de menos nodos, ya que reutilizan información adquirida en niveles previos lo que contribuye al mejor procesamiento de patrones complejos pero que limita el procesamiento espacial en niveles superiores. Cada uno de estos nodos tiene básicamente la misma funcionalidad, y se diferencian en la forma del procesamiento de la información que reciben y de las secuencias temporales en las que se producen.

El método de aprendizaje distribuido hace que los estímulos solo interactúen con determinados nodos activos en momentos dados, y aumentando o reduciendo el volumen de nodos involucrados en tanto el nivel de complejidad del estímulo de entrada. Cada uno de estos bloques de nodos activados distribuidamente están interconectados fuertemente entre si simulando las capas del neo córtex, donde las células son capaces de recordar determinados estados previos, pudiendo estar en modo activo, pasivo o predictivo.

Estos tres modos de cada nodo le permiten a la red poder emular predicciones en función de las conexiones que han sido creadas por las entradas previas, enseñado a los bloques sobre cuales deben permanecer encendidos o apagados según evoluciona el flujo de entrada, lo que hace que a la salida de cada sección existan nodos que estén al mismo tiempo en modo activo y predictivo, proporcionando estabilidad temporal cuando el flujo de entrada consiste en patrones largos.

Estos algoritmos de aprendizaje son capaces de mantener un nivel de admisión de datos continuo mientras continúa el flujo de datos, lo que hace que el algoritmo necesite inferir si los datos recibidos están en secuencias previamente aprendidas para evitar el doble procesamiento y avisar a los nodos que contienen esa información que la adicionen a la salida del nivel, reduciendo los costos de procesamiento y aumentando la velocidad de aprendizaje en niveles superiores, además de llenar patrones faltantes e interpretar datos ambiguos o de difícil interpretación.

Bajo estos conceptos, la startup de Inteligencia Artificial Vicarius FPT Inc. utilizo los conceptos de las redes corticales para diseñar una aproximación matemática del funcionamiento del cerebro humano a partir de lo que esta empresa consideraba fallas de procesamiento de información en los niveles superiores, que impedían un aprendizaje escalado eficiente y un correcto tratamiento de datos, sobre todo en lo que se refería a multimedios.

A través de estos algoritmos, la empresa californiana afirma que emula el funcionamiento de reconocimientos de patrones del cerebro humano, haciendo que la segmentación y el reconocimiento ocurran en el mismo proceso sin realizar una limpieza previa del CAPTCHA en cuestión, resolviéndolo en poco tiempo y eliminando la barrera que hacía a CAPTCHA seguro, tal como lo muestran en su video demostrativo[7], donde afirman comprobar que su sistema algorítmico puede romper con un éxito del 90% todos los CAPTCHA del mercado, incluyendo los más seguros de Google y ReCaptcha.

Sin embargo, y a pesar de estos resultados aparentemente sólidos, Vicarius se ha negado a publicar su investigación a través de una publicación académica acreditada, por lo que sus resultados no han podido ser evaluados por entidades independientes. Luis von Ahn, de la Universidad Carnegie-Mellon y co-creador de CAPTCHA no siente que el sistema, al menos al corto plazo se vea amenazado. En una entrevista con relación a los resultados de Vicarius, afirmo que “es la décima o doceava vez que alguien clama haber roto CAPTCHA”. Estas circunstancias han hecho, que a pesar de la tecnología supuestamente desplegada, una herramienta que pueda derrotar a CAPTCHA no sea viable en el futuro próximo.

[7] https://vimeo.com/77431982

EVOLUCION Y MANUTENCION DE CAPTCHA

Es de gran importancia mantener parámetros establecidos sobre como CAPTCHA debe ser implementado para poder seguir funcionando como barrera entre humanos y maquinas, tanto a corto plazo, como a mediano; en función de los avances de Inteligencia Artificial, los ya expuestos previamente, y todos los avances futuros.

CARACTERÍSTICAS PARA MANTENER CAPTCHA SEGURO AL CORTO PLAZO [6]

Los estudios sobre seguridad de CAPTCHA determinaron las características que estos sistemas comparten, tanto en creación como en vulnerabilidades. A partir de estos estudios, y en función de todas las características de anti-segmentación y anti-reconocimientos, se han condensado ciertas características y recomendaciones que debe seguir CAPTCHA para poder mantenerse seguro en internet y mantener las características del teorema expuesto previamente.

La construcción de un CAPTCHA seguro parte desde el diseño del núcleo y de las características de anti-segmentación y anti-reconocimiento. La anti-segmentación solo funcionará adecuadamente si el núcleo y el anti-reconocimiento están correctamente aplicados. Si uno de los niveles falla, el atacante ampliará sus probabilidades de vencer al sistema.

Para estas tres capas se han propuesto principios que en su conjunto permiten la creación de CAPTCHAS de texto más seguros:

- Principios de diseño de núcleo:

    - Aleatorizar la longitud de la palabra utilizada como CAPTCHA, para evitar que el atacante pueda usar el número de caracteres como herramienta.
    - Aleatorizar el tamaño de cada carácter, para evitar que los algoritmos de reconocimiento puedan utilizar los mismos puntos de reconocimiento normalizado en cada carácter, haciendo más confusa la lectura para el lector.
    - 'Ondular' los CAPTCHA, para aumentar la dificultad de encontrar los puntos de corte en caracteres colapsados o uso de líneas.
    - Evitar usar fondos, ya que los pre-procesadores pueden eliminarlos sin problema y no aportan a la seguridad del sistema.

- Principios de anti-reconocimiento:

    - Fortalecer los esquemas de anti-reconocimiento aplicados en el núcleo, para reducir la posibilidad de detección ante analizadores cada vez más sensibles.
    - No usar un complejo esquema de caracteres, ya que los caracteres especiales y otros símbolos no aumentan la seguridad del esquema, sino solo hacen que sea más difícil para las personas resolverlos.

- Principios de anti-segmentación:

    - Usar líneas o unión de caracteres, aplicados adecuadamente, aseguran que los sistemas atacantes no puedan realizar la separación de caracteres individuales, creando CAPTCHAs seguros.
    - Cuidar la implementación, sin sobre-estimar los niveles de seguridad proporcionados por la unión de letras o el uso de líneas, siguiendo las recomendaciones presentadas.

- Crear esquemas alternativos, en función de los nuevos avances en inteligencia artificial, utilizar esos conceptos para renovar el concepto de CAPTCHA, como se comprobara en el apartado siguiente.

POSIBLES EVOLUCIONES DE CAPTCHA AL MEDIANO PLAZO

CAPTCHA, como textos relativamente ilegibles o como audios difícilmente entendibles está basado en escenarios estáticos donde la respuesta esta visible para todo aquel que pueda resolverla, sin tener que en el proceso mediar ninguna interacción más allá del reconocimiento de patrones linguistico-numericos que han sido deformados. No existe ningún tipo de razonamiento ante el estímulo, lo que hace que la nueva generación de CAPTCHAs haga uso de sistemas interactivos para estar un paso delante de la Inteligencia Artificial.

Los siguientes pasos que CAPTCHA debería seguir son:

- **Personalizar la experiencia lingüística:** Estudios determinaron que a los no angloparlantes tienen dificultades para resolver CAPTCHAs en inglés, por lo que se puede asumir que para los angloparlantes les será de la misma manera difícil resolver CAPTCHAs en otros idiomas que no sean presentados en inglés, por lo que, según la región donde sea desplegado el CAPTCHA, este se presente en un idioma diferente al hablado en el determinado territorio, complicando su resolución en masa.

- **Ampliar el tamaño de los textos a ser descifrados:** Habitualmente para su resolución se presentan imágenes que contienen dos palabras modificadas. Si se comienzan a añadir oraciones cortas con sentido semántico en lugar de solo palabras al azar, para las personas vendría a ser más fácil la interacción con el sistema al verse ayudados por el idioma

y la sintaxis lingüística propia de las personas, aumentando para las maquinas la dificultad al aumentar palabras que segmentar y reconocer.

- **Comenzar la creación de CAPTCHAs interactivos:** La ventaja que existe entre la brecha humano-maquina es la imposibilidad de las máquinas de razonar frente a estímulos sensitivos interpretados, por lo que CAPTCHA debería evolucionar en un sistema interactivo que requiera obligatoriamente que la respuesta que el usuario de sean fruto de un ejercicio de razonamiento, y que pueden ser de dos tipos:

    - **De respuesta visible:** Se compone de una pregunta de elección en la que el usuario debe escoger la respuesta en función de la solicitud realizada y en donde se muestran las posibles respuestas. En función de la respuesta obtenida, el sistema determinaría la lógica de la misma en función de la pregunta y validaría o no la respuesta. La complejidad del sistema reside en utilizar preguntas que estén en sintonía con la pagina que se esté visitando y que, a pesar de la opción múltiple, la respuesta sea lo suficientemente aparente para los humanos pero que sea un ejercicio de probabilidad para las maquinas.

    - **De respuesta oculta:** Esta prueba juega más con la psicología de las personas, ya que las respuestas dejan de ser aparentes para convertirse en completamente subjetivas en función de la complejidad de pregunta. Las respuestas podrían ser tan sencillas como definir un color en función de parámetros establecidos, hasta manifestación de ideas y sentimientos en frases concretas. Para las personas es un ejercicio fácil de reconocimiento, pero para las maquinas implica encontrar un sentido semántico a la pregunta presentada y darle un contexto en específico antes de emitir una posible respuesta; es decir, la maquina necesita obligatoriamente razonar y conceptualizar la información que está recibiendo antes de emitir una respuesta válida.

Dado que la neurología aun no es capaz de encontrar una respuesta final a los ejercicios que hacen posible la conciencia y el entendimiento del ser, es que se puede hallar un punto medio entre enseñar a computadoras a leer e interpretar datos fijos y otorgarles la capacidad de razonar sobre lo interpretado y entregar una respuesta fehaciente en correspondencia. A partir de esta diferencia se hace posible que las pruebas sugeridas mantengan la brecha humano-maquina, permitiendo que CAPTCHA pueda seguir siendo usado bajo el mismo concepto como es manejado hoy en día, y mantener la ventaja sobre la Inteligencia Artificial.

CONCLUSIONES

Cuando se diseñó Internet para ser usado por el público, los ingenieros no contemplaron todo lo que era posible hacer con la red, y todas las formas de innovación que venían con el hecho de poder comunicarnos y compartir información en casi tiempo real. CAPTCHA es uno de esos proyectos que se diseñaron para un objetivo casi específico, pero que el avance de la tecnología hizo que se utilizara en muchos y más variados escenarios, al nivel de convertirse en una prueba de la evolución de la Inteligencia Artificial.

En la actualidad, cuando la vida moderna nos ha hecho casi compartir el espacio físico y virtual con miles de dispositivos electrónicos que están permanentemente conectados a Internet, es cuando se ha vuelto fehaciente la necesidad de siempre diferenciar que es lo que proviene de un ser humano y que proviene de una máquina, considerando como dato que en 2013 más del 58% del tráfico que circulaba por Internet lo producían las maquinas, fruto del Cloud Computing y del nuevo paradigma de información distribuida que se está desarrollando al globalizar la información, el Big Data.

No es realmente posible afirmar cuánto tiempo más podremos disfrutar del relativo control sobre nuestros aparatos electrónicos de la forma en la que veníamos acostumbrados, donde somos nosotros los que utilizamos la tecnología como herramientas de investigación, desarrollo y diversión; para comenzar a considerar a los implementos tecnológicos como seres relativamente sintientes que interactúan con nosotros directamente ayudando y ejecutando tareas sin necesidad de orden alguna. Actualmente estamos en los umbrales de la real inteligencia artificial, la que se aleja de la programación tradicional causística y se enfoca, fruto de los avances de la neurología y la neurobiología en alimentar a los nuevos sistemas con información sensorial real. Todos estos sistemas en los últimos años dejaron de ser simplemente fruto de programación para convertirse en redes neuronales matriciales que emulan con mucha cercanía el funcionamiento

de nuestro cerebro. El presente estudio es una prueba real de cómo estos avances están pisándonos los talones.

La evolución de la Inteligencia Artificial, sumado al descubrimiento de nuevos materiales de construcción de circuitería cada vez más reducida y potente, como el grafeno, hacen que los avances algorítmicos y físicos comiencen a romper los límites que hace 20 años hubiéramos pensado imposibles para entornos electrónicos, o situaciones solo posibles para el entretenimiento y la ciencia-ficción. Es importante señalar y mantener sobre el tapete que si bien CAPTCHA aún tiene validez como método de seguridad y autenticación, aun con toda la evolución posible para asegurarlo, es un sistema con un tiempo de vida finito y que se acorta a medida que la Inteligencia Artificial avanza a desarrollar algoritmos de conciencia y autonomía.

Parte del desafío de la evolución de CAPTCHA es comenzar a buscar un reemplazo para el mismo que no dependa de la brecha humano-maquina, sino que esté basada en otro tipo de interacciones que efectivamente puedan diferenciar, en un futuro tal vez no tan lejano, el razonamiento humano del razonamiento que pueda ser producido por una entidad autoconsciente. Todos los sistemas que dependan de Internet corren el riesgo de verse superados por el crecimiento y penetración de la tecnología en la sociedad moderna, y es parte del desafío hacia el futuro encontrar soluciones para estos problemas antes que los problemas nos apuren a encontrar las soluciones.

BIBLIOGRAFÍA

[1] CAPTCHA. http://en.wikipedia.org/wiki/CAPTCHA (Consultado el 8 de abril de 2014)

[2] VON AHN, Luis; BLUM, Manuel; HOPPER, Nicholas J.; LANGFORD, John. CAPTCHA: Using Hard AI Problems for Security. E. Biham, Editor. EUROCRYPT 2003, LNCS 2656, pp. 294–311. International Association for Cryptologic Research 2003

[3] VON AHN, Luis; BLUM, Manuel; LANGFORD, John. Telling Humans and Computers Apart (Automatically) or How Lazy Cryptographers do AI. Communications of the ACM. 2003

[4] PINKAS, Benny; SANDER, Tomas. Securing Passwords against Dictionary Attacks. In Proceedings of the ACM Computer and Security Conference (CCS' 02), pages 161–170. ACM Press. 2002.

[5] The CAPTCHA Web Page: http://www.captcha.net. 2000. (Consultado el 8 de abril de 2014)

[6] BURSZTEIN, Elie; MARTIN, Mattheu; MITCHELL, John C.; Text-based CAPTCHA Strengths and Weaknesses. ACM Computer and Communication security 2011

[7] BURSZTEIN, Elie; BETHARD, Steven; FABRY, Celine; MITCHELL, John C.; JURAFSKY, Dan. How Good are Humans at Solving CAPTCHAs? A Large Scale Evaluation. Standford University 2010

[8] NAOR, Moni. Verification of a human in the loop or Identification via the Turing Test. Unpublished Manuscript, Weismann Institute. 1997.

[9] AI Startup Vicarious Claims Milestone In Quest To Build A Brain: Cracking CAPTCHA. http://www.forbes.com/sites/roberthof/2013/10/28/ai-startup-vicarious-claims-milestone-in-quest-to-build-a-brain-craking-captcha/ (Consultado el 18 de noviembre de 2013)

[10] Captcha test 'cracked' by US firm Vicarious. http://www.bbc.co.uk/news/technology-24710209 (Consultado el 18 de noviembre de 2013)

[11] Software Firm Claims Breakthrough in Computer Vision Will Lead to Better AI. http://www.scientificamerican.com/article.cfm?id=ai-captcha-computer-vision (Consultado el 18 de noviembre de 2013)

[12] Vicarious AI passes first Turing Test: CAPTCHA. http://news.vicarious.com/ (Consultado el 18 de noviembre de 2013)

[13] Kumar Chellapilla, Patrice Y. Simard. Using Machine Learning to Break Visual Human Interaction Proofs (HIPs), NIPS 2004

[14] BROWNING, Adam; KOLAS, Dave. Defeating CAPTCHAs: Applying Neural Networks. Virginia Tech. 2009

[15] CAI, Tianhui. CAPTCHA Solving With Neural Networks. TJHSST Computer Systems Lab 2007-2008

[16] Artificial Intelligence Breaks CAPTCHA Protection, Coders Claim. http://txchnologist.com/post/65426369724/artificial-intelligence-breaks-captcha-protection (Consultado el 15 de abril de 2014)

[17] Vicarious AI breaks CAPTCHA 'Turing test'. http://www.kurzweilai.net/vicarious-ai-breaks-captcha-turing-test (Consultado el 15de abril de 2014)

[18] Vicarious announces $15 million funding for AI software based on the brain. http://www.kurzweilai.net/vicarious-announces-15-million-funding-for-ai-software-based-on-the-brain (Consultado el 15 de abril de 2014)

[19] Google's Street View neural network can now decrypt captchas better than a human. http://www.extremetech.com/computing/174275-google-has-built-a-neural-network-to-identify-100-million-house-numbers-for-streetview (Consultado el 15 de abril de 2014)

[20] GOODFELLOW, Ian J.; BULATOV, Yaroslav; IBARZ, Julian; ARNOUD, Sacha; SHET, Vinay. Multi-digit Number Recognition from Street View Imagery using Deep Convolutional Neural Networks. Google Inc. 2013

[21] PÉREZ-CARRASCO, J. A.; SERRANO, C.; ACHA, B.; SERRANO-GOTARREDONA, T.; LINARES-BARRANCO, B.. Red neuronal convolucional rápida sin fotogramas para reconocimiento de dígitos. Dpto. Teoría de la Señal, ETSIT, Universidad de Sevilla. Avda de los

Descubrimientos, s/n, CP41092. Instituto de Microelectrónica de Sevilla (IMSE-CNM-CSIC) Avda. Reina Mercedes, s/n, Sevilla. CP41012. 2011

[22] POLLEN, Daniel A. Explicit Neural Representations, Recursive Neural Networks and Conscious Visual Perception. Department of Neurology, University of Massachusetts Medical School, Worcester, MA 01655, USA. Cerebral Cortex Aug 2003, Oxford.

[23] Who Made That Captcha? http://www.nytimes.com/2014/01/19/magazine/who-made-that-captcha.html?_r=0 (Consultado el 15 de abril de 2014)

[24] CHELLAPILLA, Kumar; LARSON, Kevin; SIMARD, Patrice; CZERWINSKI, Mary. Computers beat humans at single character recognition in reading based human interaction proofs (HIPs). 2nd Conference on Email and Anti-Spam. 2005

[25] HAWKINS, Jeff; GEORGE, Dileep. Hierarchical temporal memory including HTM cortical learning algorithms. Numenta Inc. White Paper. 2011