reCAPTCHA: Human-Based Character Recognition via Web Security Measures

Luis von Ahn,* Benjamin Maurer, Colin McMillen, David Abraham, Manuel Blum

CAPTCHAs (Completely Automated Public Turing test to tell Computers and Humans Apart) are widespread security measures on the World Wide Web that prevent automated programs from abusing online services. They do so by asking humans to perform a task that computers cannot yet perform, such as deciphering distorted characters. Our research explored whether such human effort can be channeled into a useful purpose: helping to digitize old printed material by asking users to decipher scanned words from books that computerized optical character recognition failed to recognize. We showed that this method can transcribe text with a word accuracy exceeding 99%, matching the guarantee of professional human transcribers. Our apparatus is deployed in more than 40,000 Web sites and has transcribed over 440 million words.

A CAPTCHA (*1*, *2*) is a challenge response test used on the World Wide Web to determine whether a user is a human or a computer. The acronym stands for Completely Automated Public Turing test to tell Computers and Humans Apart. A typical CAPTCHA is an image containing several distorted characters that appears at the bottom of Web registration forms. Users are asked to type the wavy characters to "prove" they are human. Current computer programs cannot read distorted text as well as humans can (*3*), so CAPTCHAs act as sentries against automated programs that attempt to abuse online services. Owing to their effectiveness as a security measure, CAPTCHAs are used to protect many types of Web sites, including free e mail providers, ticket sellers, social networks, wikis, and blogs. For example, CAPTCHAs prevent ticket scalpers from using computer programs to buy large numbers of concert tickets, only to resell them at an inflated price. Sites such as Gmail and Yahoo Mail use CAPTCHAs to stop spammers from obtaining millions of free e mail accounts, which they would use to send spam e mail.

According to our estimates, humans around the world type more than 100 million CAPTCHAs every day (see supporting online text), in each case spending a few seconds typing the distorted characters. In aggregate, this amounts to hundreds of thousands of human hours per day. We report on an experiment that attempts to make positive use of the time spent by humans solving CAPTCHAs. Although CAPTCHAs are effective at preventing large scale abuse of online services, the mental effort each person spends solving them is otherwise wasted. This mental effort is invaluable, because deciphering CAPTCHAs requires people to perform a task that computers cannot.

We show how it is possible to use CAPTCHAs to help digitize typeset texts in nondigital form by enlisting humans to decipher the words that computers cannot recognize. Physical books and other texts written before the computer age are currently being digitized en masse (e.g., by the Google Books Project and the nonprofit Internet

Computer Science Department, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA.

*To whom correspondence should be addressed. E mail: biglou@cs.cmu.edu

Archive) to preserve human knowledge and to make information more accessible to the world. The pages are photographically scanned and the resulting bitmap images are transformed into text files by optical character recognition (OCR) software. This transformation into text is useful because the books can then be indexed, searched, and stored in a format that can be easily analyzed and manipulated. One of the stumbling blocks in the digitization process is that OCR is far from perfect at deciphering the words in bitmap images of scanned texts. As we show below, for older prints with faded ink and yellowed pages, OCR cannot recognize about 20% of the words. By contrast, humans are more accurate at transcribing such print. For example, two humans using the "key and verify" technique, where each types the text independently and then any discrepancies are identified, can achieve more than 99% accuracy at the word level (*4*, *5*). Unfortunately, human transcribers are expensive, so only documents of extreme importance are manually transcribed.

Our apparatus, called "reCAPTCHA," is used by more than 40,000 Web sites (*6*) and demonstrates that old print material can be transcribed, word by word, by having people solve CAPTCHAs throughout the World Wide Web. Whereas standard CAPTCHAs display images of random characters rendered by a computer, reCAPTCHA displays words taken from scanned texts. The solutions entered by humans are used to improve the digitization process. To increase efficiency and security, only the words that automated OCR programs cannot recognize are sent to humans. However, to meet the goal of a CAPTCHA (differentiating between humans and computers), the system needs to be able to verify the user's answer. To do this, reCAPTCHA gives the user two words, the one for which the answer is not known and a second "control" word for which the answer is known. If users correctly type the control word, the system assumes they are human and gains confidence that they also typed the other word correctly (Fig. 1). We describe the exact process below.

We start with an image of a scanned page. Two different OCR programs analyze the image; their respective outputs are then aligned with each other by standard string matching algorithms (*7*) and compared to each other and to an English dictionary. Any word that is deciphered differently by both OCR programs or that is not in the English dictionary is marked as "suspicious." These are typically the words that the OCR programs failed to decipher correctly. According to our analysis, about 96% of these suspicious words are recognized incorrectly by at least one of the OCR programs; conversely, 99.74% of the words not marked as suspicious are deciphered correctly by both programs. Each suspicious word is then placed in an image along with another word for which the answer is already known, the two words are distorted further to ensure that automated programs cannot decipher them, and the resulting image is used as a CAPTCHA. Users are asked to type both words correctly before being allowed through. We refer to the word whose answer is already known as the "control word" and to the new word as the "unknown word." Each reCAPTCHA challenge, then, has an unknown word and a control word, presented in random order. To lower the probability of automated programs randomly guessing the correct answer, the control words are normalized in frequency; for example, the more common word "today" and the less common word "abridged" have the same probability of being served. The vocabulary of control words contains more than 100,000 items, so a program that randomly guesses a word would only succeed 1/100,000 of the time (*8*). Additionally, only words that both OCR programs failed to recognize are used as control words. Thus, any program that can recognize these words with nonnegligible probability would represent an improvement over state of the art OCR programs.

To account for human error in the digitization process, reCAPTCHA sends every suspicious word to multiple users, each time with a different random distortion. At first, it is displayed as an unknown word. If a user enters the correct answer to the associated control word, the user's other answer is recorded as a plausible guess for the unknown word. If the first three human guesses match each other, but differ from both of the OCRs' guesses, then (and only then) the word becomes a control word in other challenges. In case of discrepancies among human answers, reCAPTCHA sends the word to more humans as an "unknown word" and picks the answer with the highest number of "votes," where each human answer counts as one vote and each OCR guess counts as one half of a vote (recall that these words all have been previously processed by OCR). In practice, these weights seem to yield the best results, though our accuracy is not very sensitive to them (as long as more weight is given to human guesses than OCR guesses). A guess must obtain at least 2.5 votes before it is chosen as the correct spelling of the word for the digitization process. Hence, if the first two human guesses match each other and one of the OCRs, they are considered a correct answer; if the first three guesses match each other but do not match either of the OCRs, they are considered a correct answer, and the word becomes a control word. To account for words that are unreadable, reCAPTCHA has a button that allows users to request a new pair of words. When six users reject a word before any correct spelling is chosen, the word is discarded as unreadable. After all suspicious words in a text have been deciphered, we apply a post processing step because human users make a variety of predictable mistakes (see supporting online text). From analysis of our data, 67.87% of the words required only two human responses to be considered correct, 17.86% required three, 7.10% required four, 3.11% required five, and only 4.06% required six or more (this includes words discarded as unreadable).

A large scale deployment of the system has enabled us to collect a number of findings (see supporting online text for more details about the deployment). The first finding is that the process of deciphering words with CAPTCHAs can match the highest quality guarantee given by dedicated human transcription services. A random sample of 50 scanned articles from five different years (1860, 1865, 1908, 1935, and 1970) of the *New York Times* archive (http://nytimes.com) was chosen and manually transcribed by two professionals to estimate the per word accuracy of reCAPTCHA, including the postprocessing corrections mentioned above. The total number of words was 24,080. Each word counted as a "hit" if the algorithm deciphered the entire word correctly or a "miss" if any of the letters were wrong. The error rate was defined as the number of misses divided by the

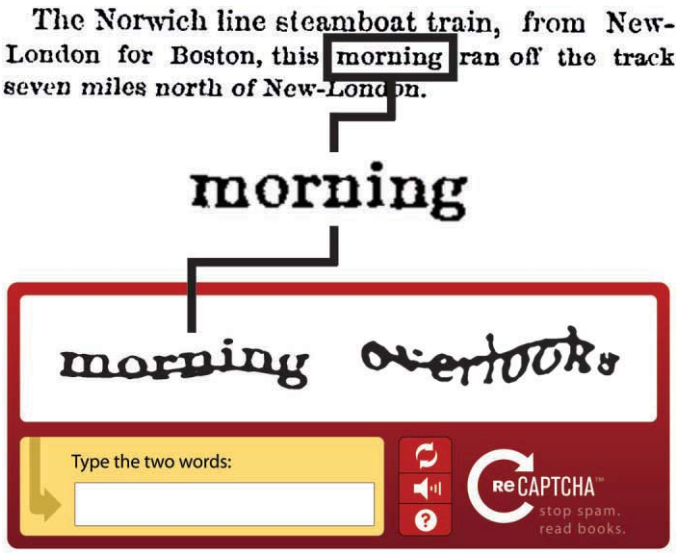

Fig. 1. The reCAPTCHA system displays words from scanned texts to humans on the World Wide Web. In this example, the word "morning" was unrecognizable by OCR. reCAPTCHA isolated the word, distorted it using random transformations including adding a line through it, and then presented it as a challenge to a user. Because the original word ("morning") was not recognized by OCR, another word for which the answer was known ("overlooks") was also presented to determine if the user entered the correct answer.

total number of words. To account for potential errors in the ground truth transcriptions, every miss was manually inspected in the ground truth and fixed in case it was an error. The results of one OCR program were run through the same process for comparison.

The reCAPTCHA system achieved an accu racy of 99.1% at the word level (216 errors out of 24,080 words), whereas the accuracy of stan dard OCR was only 83.5% (3976 errors). The percentage of words on which both OCR sys tems made a mistake was 7.3%. An accuracy of 99.1% is within the acceptable "over 99%" in dustry standard guarantee for "key and verify" transcription techniques in which two profes sional human transcribers independently type the data and discrepancies are corrected [e.g., see (*4*, *5*)]. As an anecdote, the professional man ual transcriptions of the articles that were col lected as "ground truth" to measure the accuracy of reCAPTCHA originally contained 189 errors, almost as many as those made by reCAPTCHA. There were many reasons for the mistakes made by reCAPTCHA, but the most common were issues of alignment and segmentation of the words by the OCR systems; for example, in some cases both OCR programs entirely missed a word or a set of words. The exact reason for errors in the professional human transcription is unknown to us, but the errors probably occur when the transcribers type a word differently from each other, and then a mistake is made in the correction of the discrepancy. The fact that reCAPTCHA can achieve an accuracy com parable to the "gold standard" accuracy of two independent humans can be counterintuitive be cause human transcribers make use of context (words immediately before and after), whereas words presented by reCAPTCHA are shown individually, in isolation from the original con text. On the other hand, reCAPTCHA uses a combination of OCR and multiple humans, which in some cases turns out to be more re silient to accidental typographical mistakes.

Another finding is that CAPTCHAs con stitute a viable mechanism to harness large amounts of human mental effort. After exactly 1 year of running the system, humans had solved more than 1.2 billion CAPTCHAs, amounting to over 440 million suspicious words correctly deciphered. Assuming 100,000 words per book (400 pages, 250 words per page), this is equiv alent to over 17,600 books manually transcribed (about 25% of the words in each book are marked as suspicious by our algorithm). The system continues to grow in popularity: The rate of transcription currently exceeds 4 million suspicious words per day, which is equivalent to about 160 books per day. Achieving this rate via conventional "key and verify" means (without aid from OCR, so every word in a text would be typed) would require a workforce of more than 1500 people deciphering words 40 hours per week (assuming an average rate of 60 words per minute).

There are many reasons why Web sites choose to use reCAPTCHA. First, because we use only words from scanned books upon which OCR failed, reCAPTCHA is currently more secure than the conventional CAPTCHAs that generate their own randomly distorted characters. As shown by (*9* *11*), it is possible to build algorithms that can read the distorted text generated by many widely used conventional CAPTCHAs with a success rate of more than 90% in some cases. As imple mented by us, the same algorithms fail to recog nize reCAPTCHA challenges 100% of the time. One reason for this is that the artificial distor tions of characters in conventional CAPTCHAs come from a limited (and usually simple) dis tribution of possible transformations that remain readable to humans. Therefore, it is feasible to build machine learning algorithms that, after some training, can recognize the distorted characters. The words displayed by reCAPTCHA, however, have three types of distortions. First, and most importantly, there are natural distortions that result from the underlying texts having faded through time. Second, the scanning process introduces noise. Third, we introduce artificial transforma tions similar to those used by standard CAPTCHAs so the challenges remain difficult for computer programs even if there is an OCR slightly better than ours. Although there has been work in the image processing community on modeling the natural degradation process of scanned books (*12*), such models are imperfect, so the dis tribution of distortions in reCAPTCHA is less limited. Additionally, reCAPTCHA displays only words on which two OCR programs failed. Because the words used as control words for reCAPTCHA constitute ~4% of the words from our distribution of scanned books, any program that can recognize a fraction p of the reCAPTCHA challenges without having compromised our data base can be directly used to improve the accu racy of these OCR programs by a fraction of $0.04p$ on this same distribution (*13*). This would represent an advance in state of the art OCR technology.

In essence, the words used by reCAPTCHA are the "hardest" words from scanned texts for computers to decipher. Humans, by contrast, can decipher the reCAPTCHA challenges with ease: Users of the Web sites that switch to using reCAPTCHA typically complain less often than when the sites used a different type of CAPTCHA. This is partly due to some users being more willing to accept reCAPTCHA because their work is con tributing to the digitization of human knowledge [as can be seen by the vast number of blog entries praising reCAPTCHA (*14*)]. Additionally, based on more than 1 billion responses, the overall suc cess rate for reCAPTCHA is 96.1%, a healthy number considering that a simple typing mistake would imply a failure. We do note that the success rate is not the same across all users. For example, non English speakers seem to perform slightly worse than English speakers: Internet Protocol (IP) addresses from countries where the native language is not English have success rates that vary from 92.6 to 96.9%, depending on the country, whereas IP addresses from English speaking countries range from 97.1 to 97.4% (because we have millions of data points, all of these differences are statistically significant with $P < 0.01$). Furthermore, even including English speaking countries, conditioned on failing the first challenge, IP addresses that at tempted a second challenge within 30 s have a success rate of only 89.9%. Another observation is that the success rate is proportional to the length of the control word: Four character words have a success rate of 93.7%; five character words, 95.7%; six character words, 96.4%; seven character words, 96.7%; etc. This can be explained by longer words providing more context for the users. The same relation holds when restricting attention to countries where the native language is not English, but to a lesser extent (consistent with our explanation that knowledge of the English language helps with longer words).

A second reason why Web sites adopt reCAPTCHA is that, although reCAPTCHA presents two words instead of just one, it typ ically takes no more time for users to solve a reCAPTCHA than to solve a standard CAPTCHA. Standard CAPTCHAs present six to eight ran domly chosen characters (not an English word), which take about the same time to decipher as two English words. User testing on our site (http:// captcha.net) showed that it took 13.51 s on aver age (SD 6.37) for 1000 randomly chosen users to solve a seven letter conventional CAPTCHA (25th percentile was 8.28 s, median was 12.62 s, and 75th percentile was 17.12 s), whereas it took 13.06 s on average (SD 7.67) for a different set of 1000 randomly chosen users (also from http:// captcha.net) to solve a reCAPTCHA (25th per centile was 5.79 s, median was 12.64 s, and 75th percentile was 18.91 s). The difference is not statistically significant, and indeed, the average time spent on reCAPTCHA was lower (although the median was 0.02 s higher). The fact that both standard CAPTCHAs and reCAPTCHAs take roughly the same amount of time to solve should not be surprising, because English words have patterns to which human users are accustomed. In addition, the time taken to solve reCAPTCHAs varies more widely because English words vary in length (*15*).

We believe the results presented here are part of a proof of concept of a more general idea: "Wasted" human processing power can be har nessed to solve problems that computers cannot yet solve. Some have referred to this idea as "human computation." In previous work (*16* *18*), we have shown that such processing power can be harnessed through computer games: People play these games and, as a result, collectively perform tasks that computers cannot yet perform. Inspired by this work, biologists have recently built Fold It (http://fold.it/) (*19*), a game in which people com pete to determine the ideal structure of a given protein. Here, we have shown that CAPTCHAs

constitute another avenue for "reusing" wasted computational power, while serving the useful purpose of preventing automated abuse over the Internet. A related, but different, line of work is ASIRRA (*20*), which has shown that CAPTCHAs can be used for humanitarian purposes. In this system, pictures of cats and dogs are presented to the user, who has to determine which ones are cats and which ones are dogs. The humanitarian twist is that the pictures come from animal shelters: If users like one of the cats or dogs, they can adopt it. More generally, computers do not perform as well as humans in visual recognition tasks. Perhaps a method similar to reCAPTCHA can be used to annotate or tag large quantities of images.

We hope that reCAPTCHA continues to have a positive impact on modern society by helping to digitize human knowledge.

References and Notes

1. L. von Ahn, M. Blum, N. Hopper, J. Langford, in *Advances in Cryptology*, E. Biham, Ed., vol. 2656 of *Lecture Notes in Computer Science* (Springer, Berlin, 2003), pp. 294–311.
2. L. von Ahn, M. Blum, J. Langford, *Commun. ACM* **47**, 56 (2004).
3. K. Chellapilla, K. Larson, P. Simard, M. Czerwinski, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Association for Computing Machinery, New York, 2005), pp. 711–720.
4. F. A. Long, *Am. J. Econ. Sociol.* **52**, 223 (1993).
5. Advantage Research Incorporated, *Data Collection Services Quality Control* (www.advantageresearchinc.com/dcs_qualitycontrol.htm).
6. Based on the number of unique Web sites that have registered to use the reCAPTCHA service on www.recaptcha.net as of 15 July 2008.
7. This alignment and the word segmentation done by the OCR programs are imperfect and are the biggest sources of errors in our system.
8. Because computer programs can easily attempt to pass the CAPTCHA multiple times, if a computer has a success rate of even 5%, the CAPTCHA is considered broken. A typical convention is that a program should not be able to pass the CAPTCHA with a success rate of more than 1 in 10,000. (Downloading 10,000 CAPTCHA images requires substantial usage of bandwidth, exposing the IP address as potentially abusive.) Our system uses more than 100,000 words, which yields a probability of random guessing that is much smaller than 1/10,000. By contrast, conventional CAPTCHAs that use seven random characters yield an even smaller probability of success for random guessing: $1/36^7$.
9. K. Chellapilla, P. Y. Simard, in *Advances in Neural Information Processing Systems 17*, L. K. Saul, Y. Weiss, L. Bottou, Eds. (MIT Press, Cambridge, MA, 2005), pp. 265–272.
10. G. Mori, J. Malik, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* **1**, 134 (IEEE Computer Society, Los Alamitos, CA, 2003).
11. A. Thayananthan, B. Stenger, P. H. S. Torr, R. Cipolla, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* **1**, 127 (IEEE Computer Society, Los Alamitos, CA, 2003).
12. T. Kanungo, Q. Zheng, *IEEE Trans. Pattern Anal. Mach. Intell.* **26**, 520 (2004).
13. The assumption that the adversary's program has not compromised our database is necessary. The practical concern is not a hacker somehow infiltrating our system, but an adversary that poisons our database by submitting large quantities of bogus answers, because answers to the control words in reCAPTCHA come from the users themselves. Such an attack, however, is infeasible. For an unknown word to become a control word, the first three user answers must match each other and must have correct answers to the three different associated control words. An attacker randomly guessing the answer to three different control words would have a probability of success of $1/10^{15}$.
14. Based on manual review of the results from a search of http://blogsearch.google.com/ for the term "recaptcha" on 21 July 2008.
15. These timing numbers are for users whose IP addresses come from both English and non English speaking countries. Restricting the timing numbers to IP addresses from non English speaking countries yields an average of 16.32 s for reCAPTCHA and 13.91 s for the standard CAPTCHA.
16. L. von Ahn, *Computer* **39**, 92 (2006).
17. L. von Ahn, L. Dabbish, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Association for Computing Machinery, New York, 2004), pp. 319–326.
18. L. von Ahn, R. Liu, M. Blum, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Association for Computing Machinery, New York, 2004), pp. 55–64.
19. *Economist* **387**, 105 (2008).
20. J. Elson, J. Douceur, J. Howell, in *Proceedings of the 14th ACM Conference on Computer and Communications Security* (Association for Computing Machinery, New York, 2007), pp. 366–374.

Supporting Online Material

www.sciencemag.org/cgi/content/full/1160379/DC1
SOM Text

12 May 2008; accepted 5 August 2008
Published online 14 August 2008;
10.1126/science.1160379
Include this information when citing this paper.