Nouncaptcha: An Image-Based CAPTCHA Backed by an ESP Game Implementation

Thesis by
Christopher Jeakle
cjjeakle@umich.edu

In Fulfillment of the LSA Computer Science
Honors Thesis Requirement

University of Michigan
Ann Arbor, MI

2014
(Submitted April 18, 2014)

Abstract

This paper introduces and evaluates a new image-based CAPTCHA named Nouncaptcha. Nouncaptcha has users analyze an image and select the nouns in that image from a list. This process is done for several rounds, until the user earns sufficient credit to be considered human. To generate and grade these prompts, Nouncaptcha draws from a database of tagged images. This tag database is fed by a video game based off of von Ahn and Dabbish's ESP Game [13].

User testing indicates that humans fare well on Nouncaptcha prompts. Moreover, surveyed users found Nouncaptcha easy and generally preferable when compared to reCAPTCHA, a popular CAPTCHA [26]. This paper also performs a threat analysis on Nouncaptcha. This threat analysis suggests that Nouncaptcha is capable of resisting brute-force attacks, dictionary attacks, and current computer vision methods.

1. Introduction and Motivation

CAPTCHAs (Completely Automated Public Turing tests to tell Computers and Humans Apart) are a class of artificial intelligence (AI) problems that have gained popularity preventing website abuse. CAPTCHAs are useful because malicious online activity is most destructive when automated and carried out at speeds much greater than humans can achieve. A CAPTCHA can be used to identify non-human users and prevent such automated activity, ranging from spamming online polls to rapidly registering fake accounts.

CAPTCHAs come in several forms, but by far the most popular designs are text-based and rely on obfuscation techniques to prevent character recognition. The wide-spread popularity of text-based CAPTCHAs has led to fierce competition between CAPTCHA designers and attackers. Unfortunately, this is a competition CAPTCHA designers are losing. For example, Bursztein, et al. were able to compromise or outright defeat 13 of 15 popular, previously unbroken text-based CAPTCHA designs in 2011 [6].

Fortunately, it can be argued that the defeat of those CAPTCHAs was a positive development. This is because the original CAPTCHA paper by von Ahn, et al. [1] proposed a twofold purpose for CAPTCHAs: to provide an effective Turing Test (eg: to tell humans and computers apart), and to present an unsolved AI problem. Such a system continually serves the greater good, either by successfully rate-limiting online activity, or by bringing about a meaningful technological advance in its defeat.

Unfortunately, there are no well-tested, widely-deployed schemes ready to compliment or replace character recognition CAPTCHAs. This will become a problem as the development of more and more sophisticated software eventually brings us to a point where text-based CAPTCHAs are easier for computers than for humans. In fact, as of April 2014, Google's street view technology can solve reCAPTCHA (an otherwise uncompromised design) with 99% accuracy [29]. Although the required technology for this attack is not publicly available, and reCAPTCHA is secured by additional measures, this is a serious blow to text-based CAPTCHAs. In a similar vein, Bursztein, et al. have shown that audio CAPTCHAs (which often compliment text CAPTCHAs for accessibility reasons) may be very near a similar Human-vs-AI tipping point, if not past it [7, 8].

Another motivating concern is usability. Current CAPTCHA schemes can prove quite difficult for some users to solve [8]. This is a problem, as CAPTCHAs ideally present a low barrier to entry for humans. On top of this, CAPTCHAs will only grow more cumbersome as smartphones and tablets continue to gain popularity. CAPTCHA designers can no longer assume that users will be using a keyboard and mouse. Likewise, mobile devices feature a wide range of screen sizes and aspect ratios that will need to be supported [25]. At the moment, it appears that simple button- and image-based CAPTCHAs are superior to text-based designs on mobile devices, both in terms of completion time and accuracy [10]. It is also worth noting that with current technology, voice

recognition is not particularly usable for text-based CAPTCHA responses [10].

Ultimately, having a variety of distinct, AI-hard (eg: difficult for a computer) problems is critical to the success of CAPTCHAs in their cat and mouse game balancing ease of use and security. Why, then, have few unique, provably secure classes of CAPTCHAs been widely deployed? It appears this is because CAPTCHAs are a particularly difficult class of problem to design. They must be automatically generated and graded, while remaining adequately difficult to thwart even the most advanced computer systems (ideally with a < 1/10,000 bot success rate [7, 12]). This is complicated further by human usability constraints. von Ahn, who is among those credited with the invention of the CAPTCHA, suggests that the maximum failure rate a user will tolerate is somewhere around 10% [3].

2. Types of CAPTCHAs

Today, many types of CAPTCHAs have been designed, each with unique characteristics. To analyze these CAPTCHA schemes, it is best to classify them into four general categories: text (character recognition), images (labeling or grouping), audio (interpreting stated letters/numbers), and video (identifying moving letters, or doing a semantic task like labeling).

As was mentioned earlier, text CAPTCHAs attempt to make it difficult for a computer to segment an image containing text into individual letters. This is done using character overlap, rotation, random character sizes/fonts, random string lengths, "waving" (bending the string up and down), and a variety of other distortions [6]. Text CAPTCHAs are well tested, and can be quite effective if they are implemented properly (both the Google CAPTCHA and *reCAPTCHA* are essentially uncompromised, for example) [6, 29]. Unfortunately, many text-based CAPTCHAs either rely on flawed obfuscation techniques, or are not implemented with the attention to detail needed to be secure [6, 11].

Image CAPTCHAs have been implemented by *ESP-PIX* (now defunct), *Confident*, *What's Up CAPTCHA*, *Sketcha*, and the (defeated) *Asirra* CAPTCHA, among others [14, 2, 3, 4, 9, 10, 24, 28]. These CAPTCHAs are generally cognitive tasks. For example, in *ESP-PIX* a user must select one of ~70 categories which 4 presented images fit, and in *Asirra* a user must click on pictures of cats. *Sketcha* and *What's Up CAPTCHA* are also cognitive tasks, and have users identify correct upright orientations for images. It is important to note that all of these CAPTCHAs are vulnerable to computer vision and machine learning attacks if their image databases do not have sufficient variety. In fact, insufficient variety was what ultimately compromised the *Asirra* CAPTCHA [9].

Image recognition CAPTCHAs must also avoid leaking solutions through their prompts. For example, the *IMAGINA-TION* CAPTCHA has users identify the boundaries between randomly selected, distorted, and overlapped images [28]. Unfortunately, such boundaries can be identified with no semantic knowledge at all. Rather than interpreting the meaning of the presented images, an attacker can identify a solution from broken object outlines and mis-matched textures [27]. Such information leaking can seriously undermine the efficacy of a CAPTCHA scheme.

Audio CAPTCHAs are a fairly common accessibility option, and are meant to assist users who have difficulty with text-based CAPTCHAs. The most popular types are "non-continuous" and state letters or numbers masked by adversarial noise and static. In their current state, however, audio CAPTCHAs are critically flawed and offer minimal protection from bots [7]. Audio CAPTCHAs have also proven quite difficult for humans to solve [8].

Finally, there are Video CAPTCHAs, which generally consist of a cognitive task like labeling the content of a video [15], or an identification task like that in the popular *NuCaptcha* [16]. *NuCaptcha* has users type a set of characters displayed in a video prompt. The security of *NuCaptcha* has been called into question, however, after being thwarted in a paper by

Reynaga, et al. as well as by Bursztein [12, 17]. Both studies compromised NuCaptcha by taking advantage of the extra information a video provides. Rather than having only one view of the prompt to analyze, an array of frames can be captured and studied, resulting in greater solution certainty than any one frame could deliver.

Reynaga, et al. also investigated the use of "emerging images" in a CAPTCHA. Emerging images are a phenomenon where humans can observe patterns in what computers would consider meaningless static. Ultimately, their study concluded that a more desirable CAPTCHA scheme would leverage human understandings (such as the relative size of objects), rather than continuing the development of character recognition CAPTCHAs [12].

3. Other Related Work

Beyond existing work in CAPTCHAs, studies on topics ranging from image tagging to usability were valuable while creating Nouncaptcha. For instance, Nouncaptcha's guessing game was inspired by von Ahn and Dabbish's ESP game and its use in the ESP-PIX CAPTCHA [13, 14]. Microsoft Research further refined the ESP game by developing strategies to get a greater number of higher quality tags than seen in the original ESP game [22].

Nouncaptcha also built off of CAPTCHA breaking research, particularly work by Zhu, et al. and Golle. Zhu, et al. defeated a wide range of image-based CAPTCHAs and outlined potential improvements to future image CAPTCHAs [27]. Golle defeated the Asirra CAPTCHA and explained its vulnerabilities, while also highlighting which aspects of Asirra were sound [9]. Lastly, Bursztein has several papers on the general art of breaking CAPTHCAs, particularly on breaking audio and text-based CAPTCHAs [6, 7].

Finally, several usability studies proved useful while designing and testing Nouncaptcha. Bursztein, et al. studied a wide array of text and audio CAPTCHAs using Amazon Mechanical Turk [8]. The resulting paper gave useful insight into running a CAPTCHA usability study on MTurk, as did Ross, Halderman, and Finkelstein's study of Sketcha [3]. Finally, Wismer, et al. conducted a study of several mobile-optimized CAPTCHAs, and identified ideal CAPTCHA inputs for touch devices [10].

4. Inspiration from an Unsolved Problem

So where would a new CAPTCHA fit in? What types of AI problems still need solving, and how could one leverage such a problem? An interesting space to consider is unconstrained object identification: the difficult AI problem of identifying objects in an image with no constraint on what the image could contain. This has been the basis of several proposed CAPTCHA designs [3, 4, 5], but such CAPTCHAs face a serious challenge: automatically generating prompts that are consistently solvable by humans. Asirra, a CAPTCHA proposed by Microsoft Research, hoped to address prompt generation by building off of a very large pet-adoption site's image database [2].

Unfortunately, because Asirra prompts were limited to only images of cats and dogs, it was quickly compromised. Golle was able to train a set of Support Vector Machines (SVMs) to solve an Asirra prompt with 10.3% accuracy, effectively rendering the CAPTCHA useless [9]. Still, Golle acknowledged that Asirra presents an appealing usability improvement over text-based CAPTCHAs, and notes that it could be improved to adequate levels of security with some modification [9].

5. A New CAPTCHA: Nouncaptcha

This paper proposes a new CAPTCHA, named Nouncaptcha, which is focused on recreating the usability of Asirra while avoiding the pitfalls of its limited breadth of prompts. Nouncaptcha is a relatively simple test which challenges a user's uniquely human ability to quickly and accurately identify what nouns appear in an image.

Nouncaptcha consists of two systems working in tandem. The first is an image tagging game similar to the "The ESP game" [13]. The second is a

CAPTCHA which feeds off of the game's tags. This CAPTCHA is in many ways inspired by the ESP-PIX CAPTCHA derived from the original ESP game [14]. ESP-PIX had users label a group of images by choosing a word from a large, unchanging list. This idea has been modified in various ways since it was created. Most notably a mobile version was implemented and tested in 2012 [10]. The mobile version was iPad optimized and presented all ~70 possible tags to the user as buttons below a set of prompt images.

Nouncaptcha presents a similar task through a much simpler UI. Users analyze only one image at a time, and pick tags from a short, dynamically generated list. The hope is that this design will be preferable to traditional text-based CAPTCHAs for all users, regardless of device.

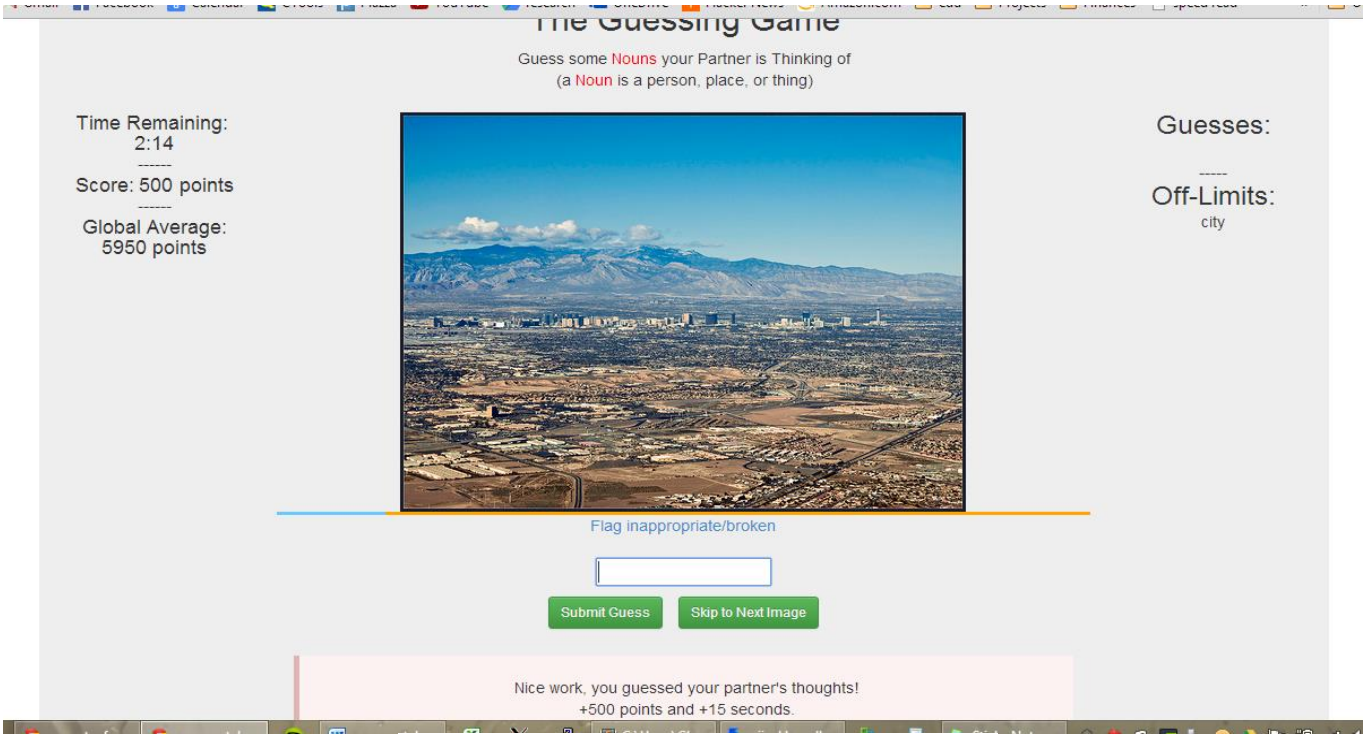
Beyond usability goals, there are security and efficiency factors to consider. With regards to this Google identified three core measures of a good CAPTCHA in the paper "What's Up CAPTCHA" [4]:

1. New challenges can be generated and graded automatically.
2. The CAPTCHA is easy for humans to solve (as noted earlier, our goal is a >90% success rate).
3. The CAPTCHA is difficult to solve using a computer (again, with a < 1/10000 AI success rate being acceptable).

All three of these points will be considered when Nouncaptcha's design is detailed and evaluated later in this paper. This paper will also measure user preference, and will evaluate whether Nouncaptcha is as usable as reCAPTCHA (a popular character recognition CAPTCHA) on various device form-factors.

6. Design (The Guessing Game)

The key to Nouncaptcha's design is "The Guessing Game", an image tagging video game meant to ensure Nouncaptcha presents a wide variety of tags on an equally wide variety of images. The guessing game fulfills these needs by grabbing liberally licensed images from Flickr, and having users tag those images while playing a video game. As was mentioned earlier, this concept was first pioneered by the ESP-Game in 2004 [13].



Pictured: The Guessing Game

At the game's start users are presented with a pop-up explaining the guessing game's rules in a few short sentences. The basic goal is to get points by matching nouns with a partner, while only using words not on an off-limits list. The catch is that players have no means of communication other than an image that both players see. When a player guesses a noun their partner agrees with, the player will advance to a new image.

Players are told that they will only play for either 15 images or until a timer is depleted. This timer starts at 2.5 minutes, and every match adds 15 seconds. Matches also reward the player with 500 points, while skips deduct 25 points and add no time. Flagging is an option, too, though it is not explicitly mentioned to the user. This feature is meant to help the game's operators identify potentially offensive or broken images. Because flags are important to keeping the game fun and friendly, a flag is rewarded in the same manner as a match. To encourage the player to at least attempt every image, the flag and skip options are hidden until 4 seconds have elapsed.

Once in a game, players are made aware of their progress by a completion bar below the image they are guessing on. Players are also shown a global average of all scores, so one can measure their performance relative to the average user, or even compete against it. Finally, the entire game is designed to work equally well on both desktop and mobile devices. This was achieved by using Twitter's bootstrap CSS libraries while designing the game.

These rules and design choices are meant to encourage a specific set of outcomes. In particular, the game is set up to encourage skipping images when no obvious nouns remain, while still motivating users to provide tags when they can. Simply getting lots of obvious tags is not enough, though, as Nouncaptcha relies on an image's tags being widely varied and unique. The game's off limits list is meant to encourage this variety. The off-limits list acts like the taboo list from the ESP game; it keeps users from repeatedly providing known tags by banning them from an image. This expands the variety of tags for a given image by motivating players to

consider nouns that do not necessarily come to mind first.

There is also the issue of determining which guesses should be considered valid tags, and which of those valid tags should be made off-limits. In the guessing game tags are found by comparing a current player's guesses to a previous player's guesses. If both players guess a certain word for an image, then that word is added to the image's list of tags. After a tag is added a sufficient number of times (5 in this implementation), the tag is considered valid and is made off limits.

There is a key difference between this scheme and that used in the ESP game: the guessing game is entirely single-player. Rather than paring users with another person, users are presented the illusion of multiplayer and play against a previous player's set of guesses. This design was chosen to make the guessing game easier to get off the ground with few concurrent users. It was also important to ensure every response from a small group of test users would be useful. The original ESP game took a different approach, focusing on live games between players. In the ESP game single player games were played against a recording of a player from a multi-player game. This design was ruled out for the guessing game, as it limits the value of a single player game to re-enforcing the tag set for previously played images.

To make single-player only gameplay work, the first time an image is presented the game waits for either 5 guesses or 7 seconds of guessing, and then assumes the user gave some valid tag. That initial set of guesses is then used with players who come afterward, and is replaced when a user skips (so long as that user has provided at least one guess). This design is based on the observation in pilot studies that users who skip generally guess as many words as they can before pressing the skip button. This makes it likely that the next user to play will have a match. An image is taken out of circulation after hitting a sufficient skip count, which is incremented by every skip, and decrements (to a minimum value of 0) with every successful match. Images with few or no remaining tags should be skipped often, allowing this scheme to keep users focused on images that still need tagging.

7. Evaluation of the Guessing Game

To ensure the Guessing Game can be relied upon for a variety of tags, a study was conducted using Amazon Mechanical Turk users (Turkers) and others recruited via Facebook and email. The study consisted of a basic usability and preference survey, along with an analysis of tag quality.

7.1 Demographics

Respondent breakdown			
Qty of responses: 62			
Age:	Mean: 30.7	Min: 19	Max: 60
Gender:	Male: 66%	Female: 33%	
Speak English:	Yes: 98.4%	No: 1.6%	
Native English Speaker:	Yes: 85.5%	No: 14.5%	

Education Level (Attending or Attained)	
No Response:	0%
Less than High School:	0%
High School:	8.1%
Associates:	11.3%
BS:	58.1%
Greater than a BS:	22.6%

Device Types (Self-reported)			
	Keyboard	Touch/ Mobile	No Response
What type of input did you use:	80.6%	6.5%	12.9%

In the above data 9 users natively spoke languages other than English. In specific, 5 users natively spoke Tamil, while other users natively spoke Hindi (1), Telugu (1), Macedonian (1), or Russian (1). It is worth noting that native language did not appear to influence enjoyment of the game, with 8 of 9 non-native English speakers saying they found the game fun. Also, game logs indicate that non-native speakers were able to agree with native speakers on tags for images.

7.2 Data

Survey Data			
	Yes	No	No Response
Did you enjoy the game:	88.7%	11.3%	0%
Did you follow the game's rules:	93.6%	4.8%	1.6%
Did you only provide nouns:	95.2%	4.8%	0%
Were prompt images large enough and easy to understand:	91.9%	6.5%	1.6%

Image Tag Data	
Images Grabbed:	50
Avg tags per image:	4.51
Unique nouns in tag set:	108
Non-nouns in tag set:	1
Images with taboo tags:	45
Avg taboo tags per image: (among images with taboo tags)	1.38

Game Log Data	
Unique games played: (with >1 round)	104
Avg rounds per game:	9.08
Avg successful tags per game:	8.29
Visits to free play link following survey: (some did not actually play)	111

Survey Instrument and logging detailed in Appendix I.

7.3 Results

For this study 50 Creative Commons images were pre-fetched from Flickr using a script. Users then played the guessing game against those images. By the end of usability testing the game had generated a total of 221 tags using 108 unique nouns. There was also one adjective tag, which was only agreed upon by players once.

Unfortunately, the game only generated an average of 1.38 off-limits tags per image. This was largely the result of the limited number of survey participants, and the comparatively large number of images used in the study.

Luckily, if the off-limits threshold was reduced from 5 tag occurrences to 4, the average image had 1.62 off-limits tags. Likewise, if the off-limits threshold was further reduced to 2, the average image had 2.73 off-limits tags. Being able to adjust the off-limits threshold proved useful in Nouncaptcha's implementation, as Nouncaptcha relies on off-limits tags to generate prompts. Ultimately, the off-limits threshold was revised to 4 after the guessing game's testing finished. As a result, 26 of 50 images had 2 or more off-limits tags. This allowed Nouncaptcha to present a fairly diverse set of prompts, while still using tags with a high level of certainty.

The usability survey showed encouraging results, with 88.7% of users saying they found the game enjoyable. Even more promising were the mobile user responses, with all 4 mobile users saying they found the game enjoyable. Though this was a small sample, it is good to see mobile users found the game fun. Given the guessing game's simple rules and short duration, it may be a good candidate for a casual mobile game.

It also appears that users were willing to play of their own accord. 111 games were served from a link that followed the usability survey. Of these games 44 were actually played and generated usable data.

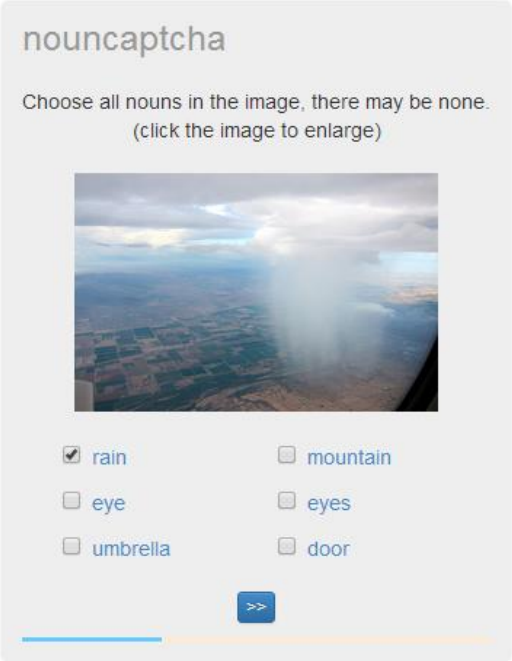
Ultimately, the game worked. The tags generated were nearly all nouns, and there was a fairly wide variety of unique nouns used. Also, users generally found the game fun. Still, there were two frequent improvement requests worth noting for a real-world deployment:

- Make the game's instructions very concise. Some users reported not understanding the game's mechanics, such as what the off-limits list was for. These concepts were explained on an instructions page, but it appears most did not read it. A popup with a summary of rules was added later in the study to address this.
- Ensure the game is served with adequate resources. This study was implemented on Heroku's free tier, so some users had to wait for a "dyno wakeup" [18]. This

degraded the user experience by adding a ~30 second wait to the game's start.

8. Design (Nouncaptcha)

As was mentioned earlier, Nouncaptcha is designed to take advantage of image tags generated by the guessing game. Nouncaptcha does this by serving random images from the guessing game's database. The user is then asked to indicate what nouns appear in the image from a set of 6 check boxes. Once submitted, Nouncaptcha grades the user's selection using a point scale. The user gets 1 point for being served a prompt, and loses 1 point for every incorrect check box. For example, if a user selects one box incorrectly and neglects to select a box that should have been checked, the user's score is -1 (eg: $1 - 2 = -1$). The goal in Nouncaptcha is to score 2 points in 5 or fewer prompts.



Pictured: A nouncaptcha prompt

This scheme was designed to strongly discourage guessing and to punish errors by potential AI, while allowing human users the opportunity to recover from small mistakes. The overall probability of a random success in the first two prompts is $\frac{1}{(2^6 * 2^6)}$ or $\frac{1}{4096}$. If a user makes one mistake in their first two prompts, then they must complete at least three prompts. This is because the user's revised minimum score after a mistake would be: 1 + 0 + 1 = 2 points. The probability of success in this case is $\frac{1}{(2^6 * 2^6) * 2^5}$ or $\frac{1}{131072}$. Following this point, the probability of a random success continues to drop at an exponential rate. This makes $\frac{1}{4096}$ (eg: making no mistakes in first two prompts) an attacker's best chance of success with truly random guessing.

Nouncaptcha grades user responses by determining what nouns are in the prompt image and what nouns are unlikely to be in that image. Correct choices are chosen from a random sample of 0 to 6 of the image's off-limits tags. Afterward another 0 to 6 incorrect tags are selected for a total of 6 options. These incorrect tags are randomly selected from all tags in the guessing game's database not associated with the prompt image. Incorrect tags also cannot appear in an image's "contentious tags" list.

The contentious tags list is a set of tags that were presented as incorrect for an image in the past and were mistaken as correct. This list allows Nouncaptcha to learn what tags are frequently mistaken as correct for an image, and avoid presenting those options.

Despite its potential usability value, the contentious tag system presents an attack vector that deserves to be addressed. An attacker could potentially spam Nouncaptcha and select every check box. Eventually, every incorrect tag would become contentious, and only correct options would be presented by Nouncaptcha. This type of attack is addressed by having contentious tags maintain a count of how often they are mistaken on an image. This count increments every time a tag is mistakenly selected for an image, and decrements every time a tag is correctly not selected. This count can become infinitely negative, so tags that were mistaken once can remain in circulation if obviously incorrect. Also,

in this study's contentious tag implementation, tags are taken out of circulation after reaching a count of two. A higher count could be required if an operator would like to have greater safety from spam attacks. Finally, a token bucket scheme like that proposed in Asirra could be used. The token bucket would assign a quantity of "tokens" to an IP address and consume the tokens with mistakes. Such a scheme could be used to detect and automatically ignore the mistakes of spam attackers [2]. It is worth nothing that a bot net, or other attacker with many IP addresses would require different countermeasures. Such a situation has been deemed out of scope for this paper, though it will be touched upon in section 10.2.

Another feature of Nouncaptcha is the progress bar below the prompt. This indicates how far the user has progressed, moving very quickly for the first two images. If the user does not succeed in the first two images, the bar moves more slowly for the remaining prompts. This lets users know how they are doing, while not revealing any information about mistakes to attackers. In particular, the progress bar reveals no information about the user's score or which prompts were failed. This is meant to prevent an attacker from randomly answering prompts and assembling a dictionary of correct and incorrect prompt solutions.

9. Nouncaptcha Usability Evaluation

Nouncaptcha was usability tested through a survey comparing it to a sample of 12 real-world reCAPTCHA prompts. The survey recorded completion times for both reCAPTCHA and Nouncaptcha, logged user accuracy in Nouncaptcha, and asked users about their preferences. This survey was advertised using Amazon Mechanical Turk, and via word of mouth. A total of 60 users tested Nouncaptcha, of whom 53 were in the US. The remaining users were in India (4), Djibouti (1), Russia (1), or did not specify (1).

9.1 Demographics

Respondent breakdown			
Qty of responses: 60			
Age:	Mean: 35.6	Min: 19	Max: 56
Gender:	Male: 60.3%	Female: 39.7%	
Speak English:	Yes: 100%	No: 0%	
Native English Speaker:	Yes: 91%	No: 9%	

Education Level (Attending or Attained)	
No Response:	1.7%
Less than High School:	3.3%
High School:	20%
Associates:	15%
BS:	46.7%
Greater than a BS:	13.3%

Device Types (Self-reported)			
	Keyboard	Touch/ Mobile	No Response
What type of input did you use:	81.6%	16.7%	1.7%

Four users performed very poorly on Nouncaptcha, with average prompt scores at least 2.48 standard deviations below the mean. Based on survey data, it appears these users may have mis-interpreted the goal of Nouncaptcha as "check all the nouns" and did not consider the prompt image. There was also one user who reported connection problems during the survey that prevented them from properly completing their CAPTCHA prompts. Because these users were particularly anomalous, they are not included in the following data analysis.

9.2 Data

- Appendix III contains histograms of reCAPTCHA and Nouncaptcha completion times.
- Appendix IV contains a breakdown and analysis of user preferences when comparing reCAPTCHA and Nouncaptcha.
- Appendix V outlines a test of statistical significance for Nouncaptcha pass rates.
- Appendix VI evaluates touch and mobile user perceptions of Nouncaptcha.

- Appendix VII contains a summary of data for omitted users.

Survey Data			
	Yes	No	No Response
Did you understand Nouncaptcha's instructions:	96.4%	1.8%	1.8%
Were images large enough and easy to understand:	94.5%	3.6%	1.8%
Were Nouncaptcha answers obvious:	76.4%	21.8%	1.8%
Were reCAPTCHA answers obvious:	74.5%	21.8%	3.6%

Preferred Input Style		
Checkboxes	Text Input	No Response
74.5%	23.6%	1.8%

Contentious Tag Data	
Avg candidate tags per image:	7.8
Unique nouns in candidate set:	68
Avg contentious tags per image: (among those with contentious tags)	2.36
Unique nouns in contentious set:	37
Percentage of images with contentious tags:	66%

Contentious tags were pre-seeded by word of mouth requests that colleagues and friends try Nouncaptcha.

CAPTCHA Completion Times (in seconds)			
Type	Mean:	Min:	Max:
reCAPTCHA	11.79	4.19	72.34
Nouncaptcha (per-prompt)	12.73	2.45	117.18
Nouncaptcha (full version)	38.93	11.49	46.64

Nouncaptcha success rate (A pass requires 2 or more points in 5 rounds)		
Type	Pass:	Fail:
Practice Round	81%	19%
Real-world Round	89%	11%

Survey Instrument and logging detailed in Appendix II.

9.3 Results

Among the 55 non-anomalous users, reCAPTCHA completion times averaged 11.58 seconds, while individual Nouncaptcha prompts averaged 12.73 seconds. The survey also tested a full Nouncaptcha, which required a user to get 2 points in 5 prompts to succeed. This full Nouncaptcha took an average of 38.27 seconds to complete. Ultimately, no statistically significant conclusions could be drawn from user perceptions of speed. In future studies it may be worthwhile to determine whether users find the speed difference between reCAPTCHA and Nouncaptcha noticeable.

Beyond speed, users were questioned about ease of use and preference. There was statistically significant evidence that users more often found Nouncaptcha easier than reCAPTCHA. Likewise, users more frequently preferred Nouncaptcha than reCAPTCHA for general use (see Appendix IV). Unfortunately, in an analysis of touch and mobile users, there were too few respondents to draw statistically significant conclusions (see Appendix VI). In a future study, it may be valuable to specifically recruit mobile users so their experience can be evaluated. Still, these findings are promising and provide evidence that the Nouncaptcha design is one that users prefer.

Along with a comparison to reCAPTCHA, there were two stages of Nouncaptcha performance testing. The first was a practice round, where users were not given any feedback and completed 5 Nouncaptcha prompts. User performance was tracked during this round in terms of score ($eg: 1 - \#\,incorrect$). The practice round scores were then evaluated to see if the users would pass a real Nouncaptcha. Based on the 2 points in 5 prompts criteria, 81% of users passed the practice round. After statistical analysis it was concluded that this pass rate was not in line with this paper's 90% goal (see Appendix V). Fortunately, users fared much better on the real thing, with 89% of users passing a real-world Nouncaptcha prompt. After analysis, this pass rate was found statistically consistent with a 90% pass rate (see Appendix V).

Despite this preliminary success, future tests will be necessary. For example, in an ideal world

Nouncaptcha would be backed by many more tags than this test had access to. Also, a full-scale deployment would benefit from an ever growing contentious tags list. Understanding how the contentious tags list performs in conjunction with a much larger tag set is important. Such interactions may make Nouncaptcha easier, and improve the pass rate beyond 90%. On the other hand, it is possible that the contentious tags list simply cannot be grown fast enough to keep up with the confusing options for an image. The contentious tags list may also grow too quickly, reducing Nouncaptcha's underlying complexity.

Finally, it is important to discuss the 5 dropped outliers. 4 of these users were dropped due to extremely anomalous scores on the Nouncaptcha practice round. Based on survey input, it appears at least one of these scores was the result of misunderstanding or ignoring Nouncaptcha's instructions. The fifth dropped user reported connection issues that prevented their completion of a full Nouncaptcha. This appears to be supported by the user's Nouncaptcha completion time of 212.27 seconds. This completion time was more than 5.5 standard deviations above the mean. The decision was made to omit these users from this paper's analysis because their experience was not representative of the typical Nouncaptcha user.

Still, making Nouncaptcha accommodating to all people requires addressing the difficulties these users experienced. For example, future iterations of nouncaptcha should be designed to prevent connection issues. Nouncaptcha currently uses websockets to communicate with the Nouncaptcha server. Unfortunately, this means Nouncaptcha requires a persistent connection to function. Moving from websockets to a RESTful design would hopefully make Nouncaptcha usable on more inconsistent networks. Also, instructions should be made more obvious. One possible option would be to implement a tutorial that walks users through Nouncaptcha. This tutorial could be played only when a user does not have a cookie set, or if a user's IP address has not been seen a sufficient number of times.

10. Threat Analysis

For Nouncaptcha to be viable it must not only be easy to use, but also secure. To evaluate Nouncaptcha's security, this paper will consider several attack vectors image based CAPTCHAs are vulnerable to. Due to the nature of CAPTCHAs, Nouncaptcha cannot guarantee an attacker cannot pass. This is because by definition any human should be able to pass a prompt. Instead Nouncaptcha's goal is to prevent automated solutions being realistically viable. Acceptable bot success rates are estimated between .6% (60/10,000) and .01% (1/10,000) [2, 7, 12]. In general, it appears researchers opt for the more stringent 1/10,000 figure.

10.1 Brute Force Guessing

The most straightforward method of compromising a CAPTCHA is by brute force. In this case an attacker would randomly guess until admitted by a Nouncaptcha. Under the current scheme an attacker has a 1/4096 probability of admittance via guessing. Despite this moderately high probability of success, random guessing would be easy to identify at the IP address level. Nouncaptcha could then rate limit such users, perhaps by requiring a higher score for admittance. Even a minimum score of 3 would drop the probability of random success to $1 / 2^{18}$, or .00038%.

10.2 Brute Force Using A Botnet

A bot net poses a unique threat to Nouncaptcha, because the attacker is able to use many IP addresses at once. This distributes the risk of detection and makes it difficult for Nouncaptcha to use the suggested brute force prevention strategy. Ultimately, bot nets are a very difficult adversary to address. As such, they have been deemed outside the scope of this paper. Such an attack is possible, however, and it would be of value for future research to investigate a solution.

10.3 Dictionary Attacks

Nouncaptcha is hard for bots because the database backing it is difficult for an attacker to recreate. Should that database be compromised, however, Nouncaptcha would become trivial to

defeat. Because of this, the information backing Nouncaptcha must be secured. Such security considerations extend beyond a direct data breech, as slowly leaking information through prompts is equally compromising.

One strategy to capture information from prompts would be to assemble a dictionary of correct responses, perhaps by paying users to solve Nouncaptcha prompts. Those responses could then be replayed as a solution when the same image is displayed again. This attack would depend on a fairly limited database size, both in terms of images, and in terms of tags on those images. Nouncaptcha defends itself from this form of attack by randomly selecting prompt images, as well as by randomly selecting the correct and incorrect nouns for a prompt. This makes it very unlikely that an attacker would see the same image again, or even the same set of options on that image. The overall probability of such an event approaches 0 as the size of Nouncaptcha's database grows.

It is the goal of the guessing game to ensure this database size assumption holds. The guessing game pulls from Flickr's creative commons licensed images. At the moment there are several hundred million such images on Flickr, and the number continues to grow [19]. Also, there are more than 85,000 nouns currently being used in the English language [20]. Given such a problem space, it is not unreasonable to believe the guessing game could keep Nouncaptcha's database sufficiently large.

If assembling a dictionary could be automated, however, the low probability of success becomes a non-issue for an attacker. To automate collection an attacker could simply guess and attempt to expose every prompt, identifying correct guesses along the way. To mitigate this, no information about a user's performance on individual Nouncaptcha prompts is exposed. For example, the user is not told their current score, nor is the user informed exactly how many prompts remain. Such information would allow an attacker to determine which prompts an algorithm guessed incorrectly. Instead the attacker is forced to view their performance in aggregate, as a pass or fail on the entire Nouncaptcha. Without information on exactly which prompts were guessed incorrectly, the attacker cannot determine which guesses to store as solutions from partially correct runs. This is important because such partially correct runs would make up the majority of a bot's attempts. The attacker could always have a human audit the computer's guesses, but this brings the attacker back to a much lower maximum speed.

10.4 Computer Vision Attacks

Finally, there is the threat of a computer vision attack defeating Nouncaptcha. At the moment, computer vision is most effective when targeted classifiers are used, such as Support Vector Machines (SVM) or a neural network. The state of the art in image classification can be observed in the PASCAL Visual Object Classes (VOC) Challenge [21]. The winning solutions to the 2012 VOC Challenges all required that an algorithm be trained for a specific classification. Given this, defeating Nouncaptcha using computer vision would require a classifier be trained for a significant portion of English language nouns. This is a tremendous problem space, and it is unlikely that building and training such a huge set of classifiers would be easier than paying humans to solve Nouncaptcha prompts.

In a similar vein, Zhu, et al. concluded that image based CAPTCHAs are weakened when they rely on a specific, pre-determined set of categories [27]. This is because an attacker can create individual classifiers to identify members of pre-determined sets. To create an adequately difficult CAPTCHA, Zhu, et al. suggest CAPTCHA designers use "…an unlimited number of different types of objects" [27]. Should the guessing game succeed in the real world, this is a constraint Nouncaptcha has been designed to meet.

There are exceptions to this assumed difficulty, however. Nouncaptcha may be fairly easy to defeat if guessing game users give obvious tags to a large sample of simple images. For example, by

analyzing color distribution and other factors an attacker could determine whether words like forest, beach, ocean, sky, and the like are correct tags. Nouncaptcha could defend itself, however, by searching its database for images tagged with a high percentage of pre-determined "trivial nouns". Those images could then be pruned or limited to use in conjunction with harder images.

Finally, recent developments in computer vision may weaken the assumption that classifiers must be individually designed and trained. A 2011 study by Stanford and Google researchers implemented an unsupervised learning system [30]. In that study, a face detector and other object classifiers were implemented using only unlabeled training data. This system achieved a groundbreaking, but relatively low 15.8% success rate on a range of test categories. As unsupervised learning systems develop, the assumption that unconstrained object identification is hard for computers may be invalidated. Such a development would undoubtedly break Nouncaptcha; but as the original CAPTCHA paper points out, that is the cost of a tremendous AI advancement [1].

11. Conclusion and Future Investigations

This paper described and tested Nouncaptcha, an image based CAPTCHA backed by an ESP game implementation. By building off of existing literature on image tagging, usability, and CAPTCHA design, Nouncaptcha hopes to offer an alternative to text-based CAPTCHAs. Nouncaptcha fared well in a study of users' preferences, proving both easier and more pleasant than reCAPTCHA. Nouncaptcha also achieved a reasonable user success rate. Finally, Nouncaptcha fared well in this paper's security analysis.

This is not to say there are no experiments that could improve Nouncaptcha. For example, it may be worthwhile to test a reverse Nouncaptcha, where users must identify what nouns are not in an image. Such a design may result in faster solving times or less ambiguity, because the contentious tags list would help make correct answers more obvious, rather than making incorrect answers more obvious. Further research may also implement and test a training system that steps new users through a Nouncaptcha. Some users did very poorly in Nouncaptcha's testing due to misinterpretation of instructions. Addressing this with a tutorial or some other tool may alleviate that usability concern.

The guessing game could also be improved. For example, Microsoft has done research on ESP game adjustments that yield a greater variety of tags that are also much more specific [22]. Such specific tags may make both correct and incorrect options more obvious in Nouncaptcha, hopefully increasing the human success rate and solution speed.

Finally, Nouncaptcha's data could be used to assist research in various fields. For example, as images grow old they could be pruned from Nouncaptcha's database and released as a training set for computer vision research. Nouncaptcha may also be of use to human psychology researchers. For example, contentious tags were never explicitly listed for an image, but still appear valid to users. It may be interesting to determine why such tags did not occur to players of the guessing game.

Links

This paper's implementation of Nouncaptcha and the Guessing Game can be viewed online at www.nouncaptcha.com.

Acknowledgements

I would like to thank Doctors Halderman, Johnson-Roberson, and Bailey for their advice and input throughout this project. I would also like to thank Anne Drolet for her suggestions while choosing statistical tests.

The image in the Guessing Game screenshot can be found at:
https://www.flickr.com/photos/22746515@N02/8309080001

The image in the Nouncaptcha screenshot can be found at: https://www.flickr.com/photos/pagedooley/7899921242

Revisions after Submission for Grading

The dropped users mentioned in section 9.1 were originally noted as performing 2.5 standard deviations below the mean. That figure was based on notes that were rounded, and has been updated to 2.48 standard deviations.

The word "euqal" was corrected to "equal" in Appendices IV and V.

References

[1]: L. V. Ahn, M. Blum, N. J. Hopper, and J. Langford. "CAPTCHA: using hard AI problems for security," In Eurocrypt, pages 294–311, 2003.

[2]: J. Elson, J. Douceur, J. Howell, and J. Saul. "Asirra: a Captcha that exploits interest-aligned manual image categorization," In Proceedings of ACM CCS 2007, pages 366–374, 2007.

[3]: Steven A. Ross, J. Alex Halderman, and Adam Finkelstein. "Sketcha: A Captcha Based on Line Drawings of 3D Models," WWW 2010, April 2010.

[4]: Rich Gossweiler, Maryam Kamvar, and Shumeet Baluja. "What's up captcha? A captcha based on image orientation," In *Proceedings of WWW 2009*, the 18th International World Wide Web Conference, 2009.

[5]: Chew, M. and Tygar, J. D. "Image recognition captchas." In Proceedings of the Information Security Conference, 2004.

[6]: Elie Bursztein, Matthieu Martin, and John C. Mitchell. "Text-based CAPTCHA Strengths and Weaknesses," In Computer and Communications Security (CCS), 2011.

[7]: Bursztein, E.; Beauxis, R.; Paskov, H.; Perito, D.; Fabry, C.; Mitchell, J., "The Failure of Noise-Based Non-continuous Audio Captchas," 2011 IEEE Symposium on Security and Privacy (SP), 2011.

[8]: Elie Bursztein, Steven Bethard, Celine Fabry, Dan Jurafsky, and John C. Mitchell. "How Good are Humans at Solving CAPTCHAs? A Large Scale Evaluation," In Security and Privacy, 2010.

[9]: Philippe Golle. "Machine learning attacks against the Asirra CAPTCHA". In Proceedings of the 15th ACM conference on Computer and communications security (CCS '08). ACM, 2008.

[10]: Wismer A.J., Madathil K.C., Koikkara R., Juang K.A., Greenstein J.S. "Evaluating the usability of CAPTCHAs on a mobile device with voice and touch input," In Proceedings of the Human Factors and Ergonomics Society, 2012.

[11]: Mori, G. and Malik, J., "Recognizing objects in adversarial clutter: breaking a visual CAPTCHA," In Proceedings of 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003.

[12]: Yi Xu, Gerardo Reynaga, Sonia Chiasson, Jan-Michael Frahm, Fabian Monrose, and Paul Van Oorschot, "Security and Usability Challenges of Moving-Object CAPTCHAs: Decoding Codewords in Motion," the 21st USENIX Security Symposium, 2012

[13]: Luis von Ahn and Laura Dabbish. "Labeling images with a computer game," In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '04), 2004.

[14]: ESP-PIX Home Page (not maintained), see: http://server251.theory.cs.cmu.edu/cgi-bin/esp-pix/esp-pix, accessed April 2014

[15]: Kurt Alfred Kluever and Richard Zanibbi. "Balancing usability and security in a video CAPTCHA," In Proceedings of the 5th Symposium on Usable Privacy and Security (SOUPS '09), 2009.

[16]: Nucaptcha home page, see: http://www.nucaptcha.com/, accessed April 2014

[17]: How we broke the NuCaptcha video scheme and what we proposed to fix it, see: http://www.elie.net/blog/security/how-we-broke-the-nucaptcha-video-scheme-and-what-we-propose-to-fix-it, accessed April 2014

[18]:Heroku documentation of dyno behavior, see:https://devcenter.heroku.com/articles/dynos, accessed April 2014

[19]: Flickr creative commons statistics, see: https://www.flickr.com/creativecommons/, accessed April 2014

[20]: Approximation of words in English language, see:http://www.oxforddictionaries.com/words/how-many-words-are-there-in-the-english-language, accessed April 2014

[21]:PASCAL VOC Home Page, see: http://pascallin.ecs.soton.ac.uk/challenges/VOC/, accessed April 2014

[22]: Stephen Robertson, Milan Vojnović, and Ingmar Weber. "Rethinking the ESP Game," In Conference on Human Factors in Computing Systems, 2008

[23]: Christoph Fritsch, Michael Netter, Andreas Reisser, and Günther Pernul. "Attacking Image Recognition Captchas: A Naive but Effective Approach," In proceeding of: Trust, Privacy and Security in Digital Business, 7th International Conference, TrustBus, 2010.

[24]: Shardul Vikram, Yinan Fan, and Guofei Gu. "SEMAGE: a new image-based two-factor CAPTCHA," In Proceedings of the 27th Annual Computer Security Applications Conference (ACSAC '11), 2011.

[25]: Open Signal's Android Fragmentation Report July 2013, see: http://opensignal.com/reports/fragmentation-2013/, accessed April 2014

[26]: reCAPTCHA home page, see: https://www.google.com/recaptcha/intro/index.html, accessed April 2014

[27]: Bin B. Zhu, Jeff Yan, Qiujie Li, Chao Yang, Jia Liu, Ning Xu, Meng Yi, and Kaiwei Cai. "Attacks and Design of Image Recognition CAPTCHAs," ACM Conference on Computer and Communications Security (ACM CCS), 2010.

[28]: Ritendra Datta, Jia Li and James Z. Wang. "IMAGINATION: A Robust Image-based CAPTCHA Generation System," Proceedings of the ACM Multimedia Conference, pp. 331-334, Singapore, ACM, 2005.

[29]: Google Online Security Blog: Street View and reCAPTCHA Technology Just Got Smarter, see: http://googleonlinesecurity.blogspot.com/2014/04/street-view-and-recaptcha-technology.html, accessed April 2014

[30]: Quoc Le, Marc'Aurelio Ranzato, Rajat Monga, Matthieu Devin, Kai Chen, Greg Corrado, Jeff Dean, and Andrew Ng. "Building high-level features using large scale unsupervised learning," International Conference in Machine Learning (2012), 2012.

Appendices:

Appendix I:

Guessing Game: Survey Instrument and Logging Details

Survey Prompts:

Users played the guessing game and were asked the following questions. All questions were optional.

- What is your native language?
  - Text box
- Do you speak English?
  - Yes/No
- What country are you in?
  - Text Box
- If in the US, what state are you in?
  - Text Box
- What is your age?
  - Text Box, numeric only
- What is your gender?
  - Male or Female radio buttons
- How much education do you have (or are getting)?
  - Radio Buttons
  - Less than high school, high school, associates, bachelors, greater than a bachelors
- What type of input did you use?
  - Radio buttons, with other corresponding to a text box
  - Mouse and keyboard, touch, or other (Please List)
- Did you enjoy the game?
  - Yes/No
- Did you follow the game's instructions?
  - Yes/No
- Did you only provide nouns in your responses?
  - Yes/No
- Were the images large enough and easy to understand?
  - Yes/No
- How did you find this survey?
  - Text
- Do you have any suggestions or input on the game?
  - Free response text box
- Leave any additional comments here.
  - Free response text box

Survey available at: http://www.nouncaptcha.com/game_survey

Logging Strategy

The game logged 7 kinds of events:

1. New game start
2. Seed guesses generated (from first round for an image)
3. Match (W/ JSON of partner's guess list, and matched word)
4. Skip (W/ JSON of partner's guess list and player's guess list)
5. New image sent
6. Disconnect
7. Game issues

The Game also logged the number of visits to the guessing game through a specific link.

Appendix II:

Nouncaptcha: Survey Instrument and Logging Strategy

Survey Prompts:

Users completed 3 reCAPTCHA prompts, 5 Nouncaptcha practice rounds, a real Nouncaptcha, and answered the following questions. All questions were optional.

- At the top of the survey users are reminded that "reCAPTCHA was entering text, nouncaptcha was checking boxes"
- What is your native language?
    - Text box
- Do you speak English?
    - Yes/No
- What country are you in?
    - Text Box
- If in the US, what state are you in?
    - Text Box
- What is your age?
    - Text Box, numeric only
- What is your gender?
    - Male or Female radio buttons
- How much education do you have (or are getting)?
    - Radio Buttons
    - Less than high school, high school, associates, bachelors, greater than a bachelors
- What type of input did you use?
    - Radio buttons, with other corresponding to a text box
    - Mouse and keyboard, touch, or other (Please List)
- Did you understand nouncaptcha's instructions?
    - Yes/No
- Were the images large enough and easy to understand?
    - Yes/No
- Which style of CAPTCHA did you prefer?
    - Radio buttons
    - Checkboxes, Text input
- Did the answers to the check box CAPTCHA seem obvious?
    - Yes/No
- Did the answers to the reCAPTCHA prompts seem obvious?
    - Yes/No
- Which CAPTCHA was easier?
    - 5 Radio buttons on a Likert scale
    - The first indicates the user strongly felt reCAPTCHA was easier

    - The next one indicates a less strong sentiment, but a preference for reCAPTCHA
    - The third button indicates no sentiment either way
    - The fourth indicates slight preference for Nouncaptcha
    - The last one indicates a strong preference for Nouncaptcha
- Which CAPTCHA was faster?
    - Same 5 radio button format as above
- Which CAPTCHA would you rather use for both mobile and desktop computing?
    - Same 5 radio button format as above
- How did you come across this survey?
    - Text box
- Do you have any suggestions, comments, or input for nouncaptcha?
    - Free response text box

Survey available at: http://www.nouncaptcha.com/captcha_survey

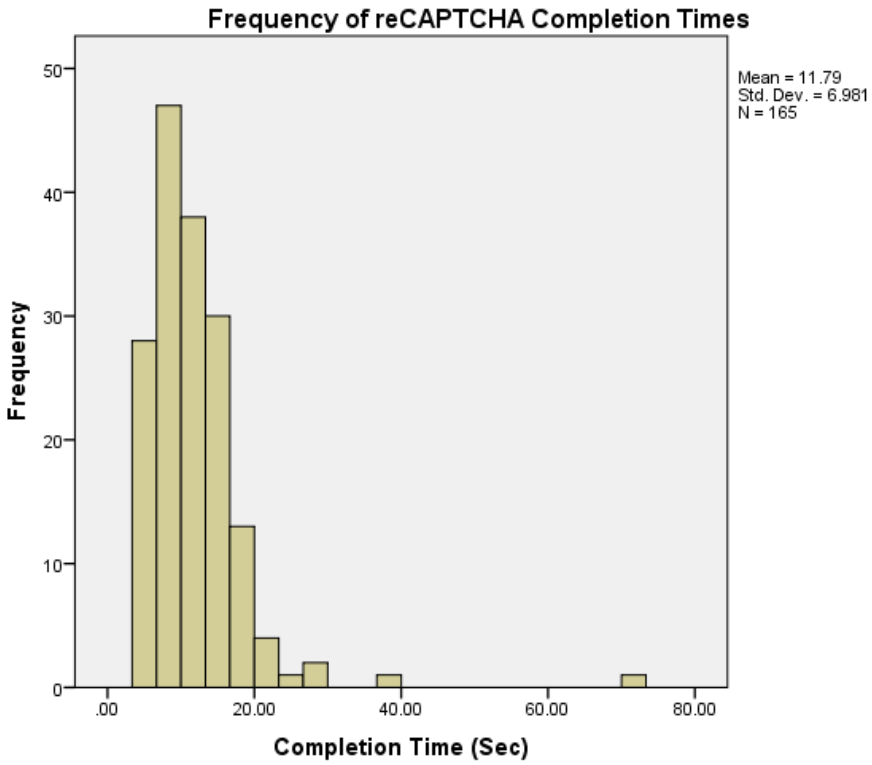
Logging Strategy

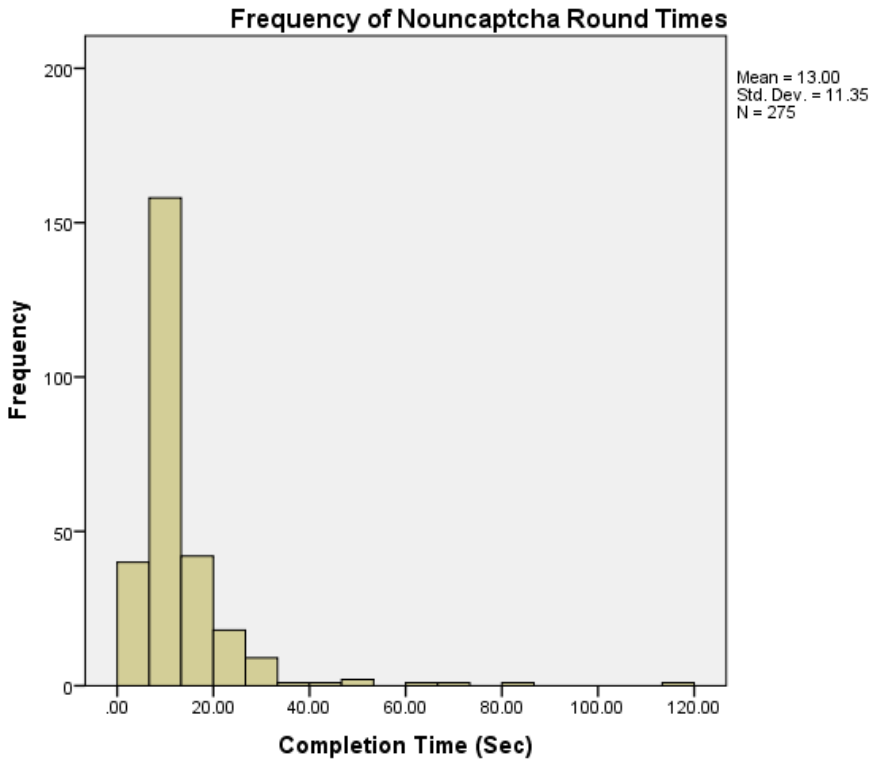
The Nouncaptcha survey logged:

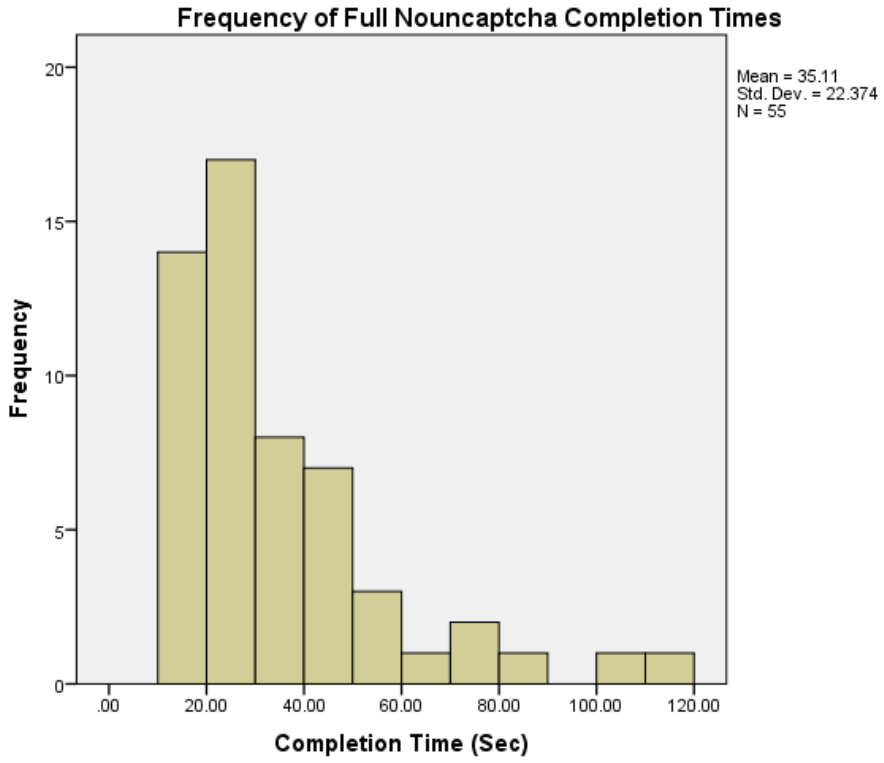
1. Completion time and answers for 3 reCAPTCHA prompts
2. Completion time and score for 5 Nouncaptcha rounds (practice)
3. Completion time and whether or not passed for a full Nouncaptcha prompt

Appendix III:

CAPTCHA Response Time Histograms




Frequency of Nouncaptcha Round Times
Mean = 13.00
Std. Dev. = 11.35
N = 275
Frequency
200
150
100
50
0
.00
20.00
40.00
60.00
80.00
100.00
120.00
Completion Time (Sec)


Frequency of Full Nouncaptcha Completion Times
Mean = 35.11
Std. Dev. = 22.374
N = 55
Frequency
20
15
10
5
0
.00
20.00
40.00
60.00
80.00
100.00
120.00
Completion Time (Sec)

Appendix IV:

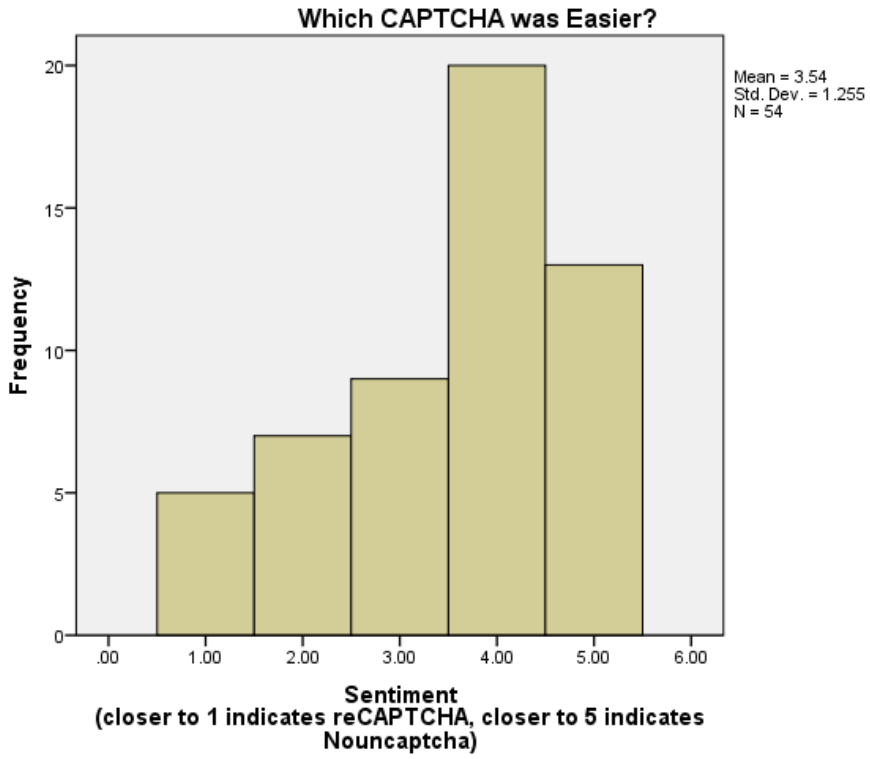
reCAPTCHA vs Nouncaptcha Comparison, Statistical analysis

(Responses are on a Likert scale)

Which CAPTCHA is Easier?	
Strongly Prefer reCAPTCHA:	9.1%
Somewhat Prefer reCAPTCHA:	12.7%
No Preference:	16.4%
Somewhat Prefer Nouncaptcha:	36.4%
Strongly Prefer Nouncaptcha:	23.6%
No Response:	1.8%

Which CAPTCHA is Faster?	
Strongly Prefer reCAPTCHA:	18.2%
Somewhat Prefer reCAPTCHA:	9.1%
No Preference:	16.4%
Somewhat Prefer Nouncaptcha:	27.3%
Strongly Prefer Nouncaptcha:	27.3%
No Response:	1.8%

Which CAPTCHA Would You Rather Use?	
Strongly Prefer reCAPTCHA:	16.4%
Somewhat Prefer reCAPTCHA:	3.6%
No Preference:	14.5%
Somewhat Prefer Nouncaptcha:	29.1%
Strongly Prefer Nouncaptcha:	34.5%
No Response:	1.8%



Chi-Squared Test:

H_0: These results were due to random chance.

H_A: These results were not a random occurrence.

A random distribution would have $\frac{54}{5} = 10.8$ responses per option.

Chi squared statistic approximation: $\frac{(5-10.8)^2}{10.8} + \frac{(7-10.8)^2}{10.8} + \frac{(9-10.8)^2}{10.8} + \frac{(20-10.8)^2}{10.8} + \frac{(13-10.8)^2}{10.8} = 13.04$

With 4 degrees of freedom this evaluates to: .01 < p < .05

- **Conclusion: One can be 95% confident the easiness comparison data were not a random result.**

Sign Test - Binomial:

H_0: The frequency at which users found reCAPTCHA easier was equal to the frequency at which users found Nouncaptcha easier.

H_A: The frequencies are different.

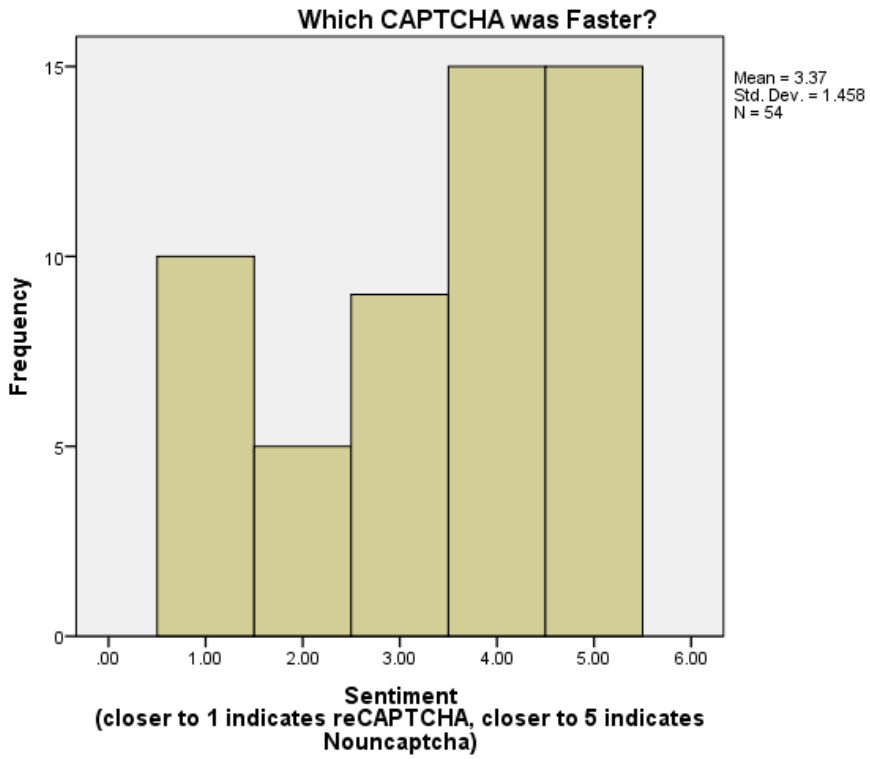
Probability of a random selection being in favor of a given option: 0.5

Left-side ("Failures"/favor reCAPTCHA): 12

Right-side ("Successes"/favor Nouncaptcha): 33

P(2-tailed) = 0.0024589 < .05

- **Conclusion: There was statistically significant evidence that the frequencies differed, at the 95% level.**
- **By inspection one can see the frequency at which users found Nouncaptcha easier outweighs the frequency at which reCAPTCHA was found easier.**
- **With 95% confidence, one can conclude more often users find Nouncaptcha easier than recaptcha.**



Chi-Squared Test:

H_0: These results were due to random chance.

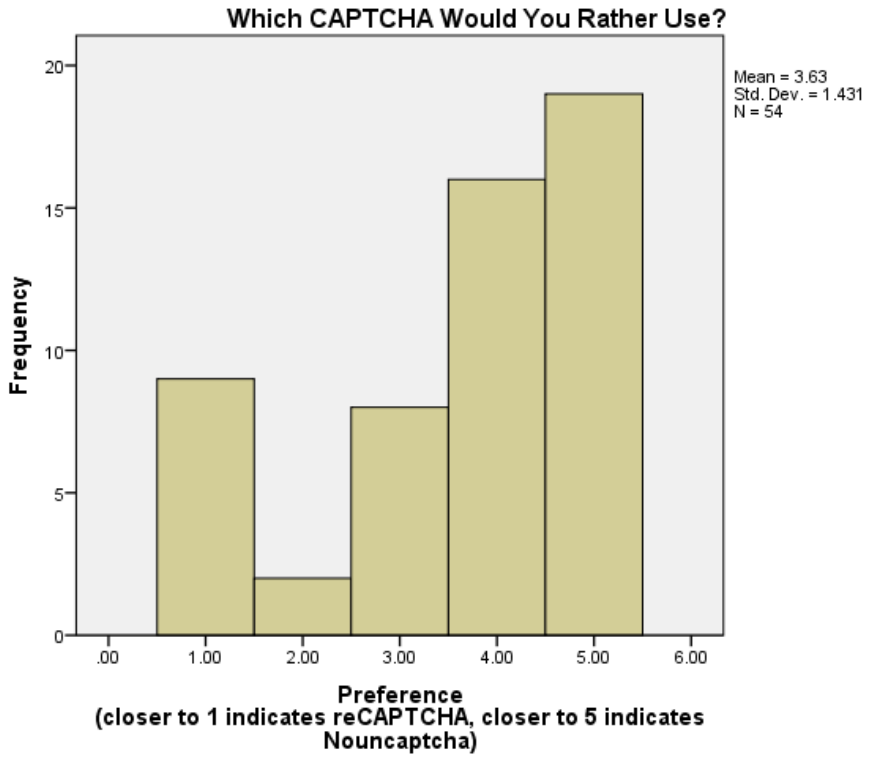
H_A: These results were not a random occurrence.

A random distribution would have $\frac{54}{5} = 10.8$ responses per option.

Chi squared statistic approximation: $\frac{(10-10.8)^2}{10.8} + \frac{(5-10.8)^2}{10.8} + \frac{(9-10.8)^2}{10.8} + \frac{(15-10.8)^2}{10.8} + \frac{(15-10.8)^2}{10.8} = 6.74$

With 4 degrees of freedom this evaluates to: $.10 < p < .20$

- **Conclusion: There was no particularly significant evidence that this data is not a random result. No meaningful infrences can be reliably derived from this sample.**



Chi-Squared Test:

H_0: These results were due to random chance.

H_A: These results were not a random occurrence.

A random distribution would have $\frac{54}{5} = 10.8$ responses per option.

Chi squared statistic approximation: $\frac{(9-10.8)^2}{10.8} + \frac{(2-10.8)^2}{10.8} + \frac{(8-10.8)^2}{10.8} + \frac{(16-10.8)^2}{10.8} + \frac{(19-10.8)^2}{10.8} = 16.93$

With 4 degrees of freedom this evaluates to: .001 < p < .01

- **Conclusion: One can be 99% confident that these results were not due to random chance.**

Sign Test - Binomial:

H_0: The frequency at which users found reCAPTCHA preferable was equal to the frequency at which users found Nouncaptcha preferable.

H_A: The frequencies are different.

Probability of a random selection being in favor of a given option: 0.5

Left-side ("Failures"/favor reCAPTCHA): 11

Right-side ("Successes"/favor Nouncaptcha): 35

P(2-tailed) = 0.000535586 < .05

- **Conclusion: There was statistically significant evidence that the frequencies differed, at the 95% level.**
- **By inspection one can see the frequency at which users found Nouncaptcha preferable for general use outweighs the frequency at which reCAPTCHA was found preferable.**
- **With 95% confidence, one can conclude more often users preferred Nouncaptcha than reCAPTCHA for general use.**
- (Note: general use in this case means “both mobile and desktop use”, as that is how the corresponding question was written on the survey)

Appendix V.

Chi-Squared test of significance for observed pass rate vs. 90% goal CAPTCHA pass rate.

Nouncaptcha success rate (A pass requires 2 or more points in 5 rounds)		
Type	Pass:	Fail:
Practice Round	45	10
Real-world Round	49	6

H_0: Nouncaptcha pass rate was 90%
H_A: Nouncaptcha pass rate was different from 90%
Degrees of freedom: 1
Critical Value for 95% confidence interval (CI), at 1 deg of freedom: 3.84
Critical Value for 20% confidence interval (CI), at 1 deg of freedom: 0.06

Practice Round Nouncaptcha:
Observed Pass Rate:45
Expected Pass Rate: 55 * .9 = 45.9
Observed Fail Rate:10
Observed Fail Rate: 55 * .1 = 5.5

Chi Squared Test: $\frac{(45-49.5)^2}{49.5} + \frac{(10-5.5)^2}{5.5} = 4.09$
Critical Value Comparison (95% CI): 4.09 is not < 3.84

- **Conclusion: The practice round pass rate is not statistically consistent with the desired 90% pass rate when requiring 95% confidence to reject H_0.**

Real-World Nouncaptcha:
Observed Pass Rate: 49
Expected Pass Rate: 55 * .9 = 49.5
Observed Fail Rate: 6
Expected Fail Rate: 55 * .1 = 5.5

Chi Squared Test: $\frac{(49-49.5)^2}{49.5} + \frac{(6-5.5)^2}{5.5} = .0505$
Critical Value Comparison (95% CI): 0.0505 < 3.84
Critical Value Comparison (20% CI): 0.0505 < .06

- **Conclusion: The real-world Nouncaptcha pass rate is statistically consistent with the desired 90% pass rate, even when requiring 20% confidence to reject H_0.**

Appendix VI:

Touch and mobile user data

Respondent breakdown			
Qty of responses: 10			
Age:	Mean: 35.0	Min: 21	Max: 54
Gender:	Male: 50%	Female: 50%	
Speak English:	Yes: 100%	No: 0%	
Native English Speaker:	Yes: 90%	No Response: 10%	

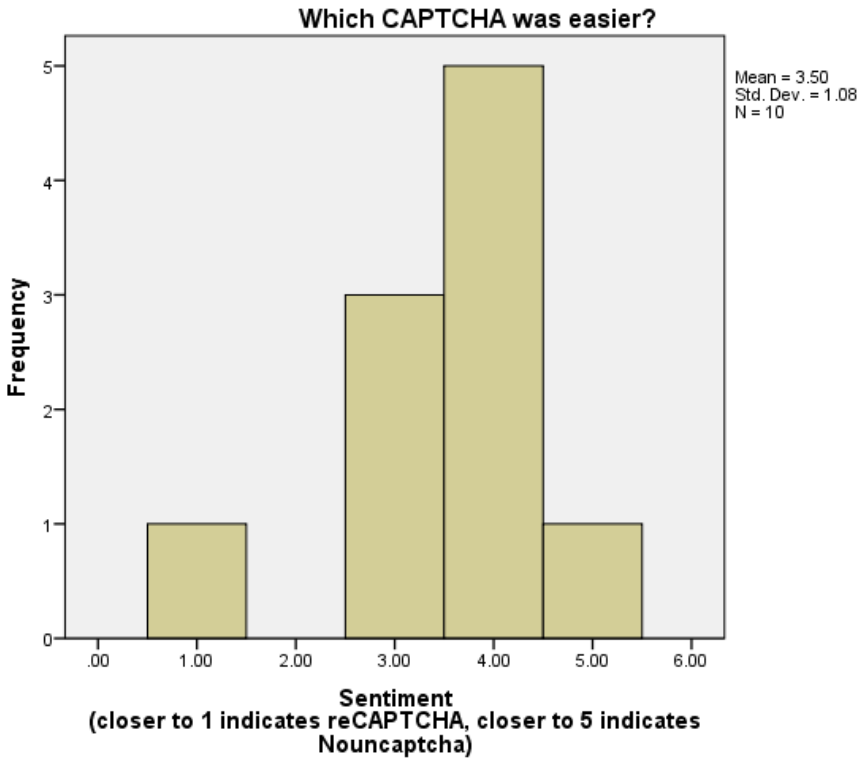
Education Level (Attending or Attained)	
No Response:	0%
Less than High School:	0%
High School:	10%
Associates:	0%
BS:	80%
Greater than a BS:	10%

Survey Data		
	Yes	No
Did you understand Nouncaptcha's instructions:	90%	10%
Were images large enough and easy to understand:	100%	0%
Were Nouncaptcha answers obvious:	80%	20%
Were reCAPTCHA answers obvious:	80%	20%

Preferred Input Style	
Checkboxes	Text Input
70%	30%

CAPTCHA Completion Times (in seconds)			
Type	Mean:	Min:	Max:
reCAPTCHA	11.77	4.62	37.39
Nouncaptcha (per-prompt)	11.56	2.45	44.47
Nouncaptcha (full version)	29.92	11.49	53.51

Nouncaptcha success rate (A pass requires 2 or more points in 5 rounds)		
Type	Pass:	Fail:
Practice Round	80%	20%
Real-world Round	90%	10%



Chi-Squared Test:

H_0: These results were due to random chance.

H_A: These results were not a random occurrence.

A random distribution would have $\frac{10}{5} = 2$ responses per option.

Chi squared statistic approximation: $\frac{(1-2)^2}{2} + \frac{(0-2)^2}{2} + \frac{(3-2)^2}{2} + \frac{(5-2)^2}{2} + \frac{(1-2)^2}{2} = 8$

With 4 degrees of freedom this evaluates to: .05 < p < .10

- **Conclusion: One can be 90% confident the easiness comparison results were not a random result.**

Sign Test - Binomial:

H_0: The frequency at which users found reCAPTCHA easier was equal to the frequency at which users found Nouncaptcha easier.

H_A: The frequencies are different.

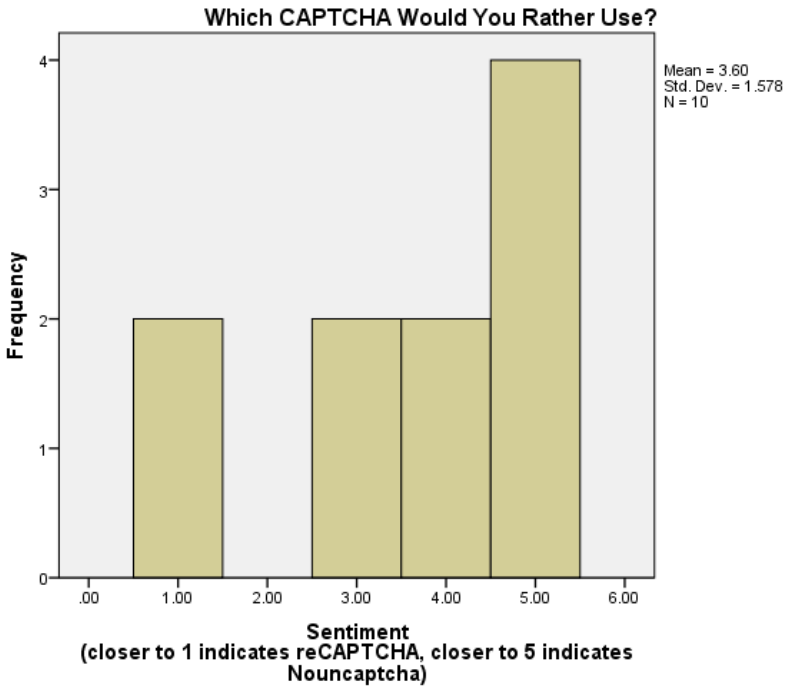
Probability of a random selection being in favor of a given option: 0.5

Left-side ("Failures"/favor reCAPTCHA): 1

Right-side ("Successes"/favor Nouncaptcha): 6

P(2-tailed) = 0.1250 is not < .05

- **Conclusion: There was not statistically significant evidence that the frequencies differed, at the 95% level.**



Chi-Squared Test:

H_0: These results were due to random chance.

H_A: These results were not a random occurrence.

A random distribution would have $\frac{10}{5} = 2$ responses per option.

Chi squared statistic approximation: $\frac{(2-2)^2}{2} + \frac{(0-2)^2}{2} + \frac{(2-2)^2}{2} + \frac{(2-2)^2}{2} + \frac{(4-2)^2}{2} = 4$

With 4 degrees of freedom this evaluates to: .50 < p < .30

- **Conclusion: At the 95% level, one cannot be confident the preference comparison results were not a random result.**

Appendix VII:

Dropped user data (5 users)

Respondent breakdown			
Qty of responses: 5			
Age:	Mean: 42.6	Min: 39	Max: 47
Gender:	Male: 60%	Female: 40%	
Speak English:	Yes: 100%	No: 0%	
Native English Speaker:	Yes: 80%	No: 10% (Spoke Tamil)	

Education Level (Attending or Attained)	
No Response:	0%
Less than High School:	0%
High School:	0%
Associates:	20%
BS:	60%
Greater than a BS:	20%

Device Types (Self-reported)			
	Keyboard	Touch/ Mobile	No Response
What type of input did you use:	100%	0%	0%

Survey Data			
	Yes	No	No Response
Did you understand Nouncaptcha's instructions:	80%	20%	0%
Were images large enough and easy to understand:	100%	0%	0%
Were Nouncaptcha answers obvious:	60%	40%	0%
Were reCAPTCHA answers obvious:	60%	20%	20%

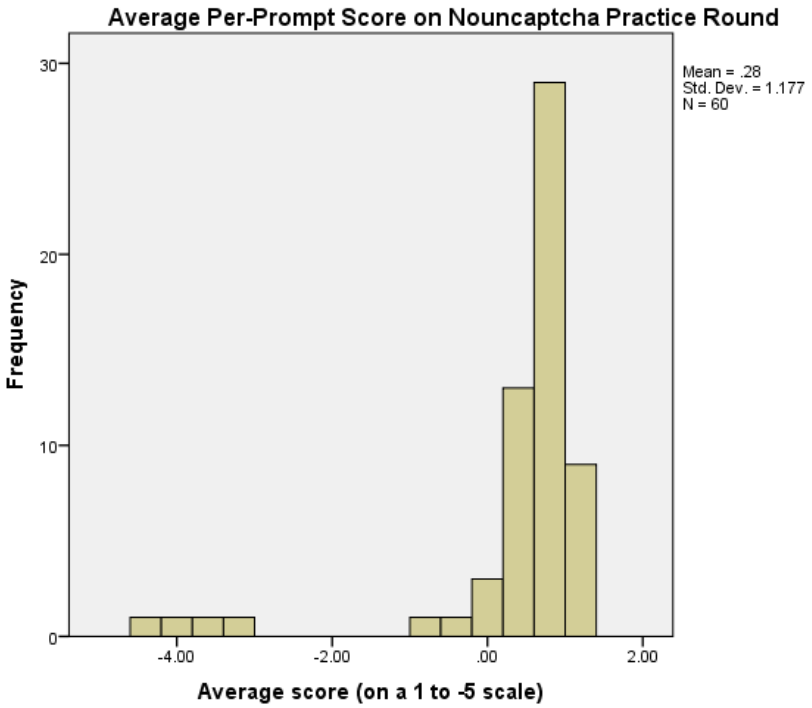
Preferred Input Style		
Checkboxes	Text Input	No Response
60%	40%	0%

CAPTCHA Completion Times (in seconds)			
Type	Mean:	Min:	Max:
reCAPTCHA	12.73	8.08	23.36
Nouncaptcha (per-prompt)	11.78	3.74	39.90
Nouncaptcha (full version)	81.01	40.88	212.27

Nouncaptcha success rate (A pass requires 2 or more points in 5 rounds)		
Type	Pass:	Fail:
Practice Round	0%	100%
Real-world Round	0%	100%

Sentiment Data Summary			
	Easier	Faster	Preferable for Desktop and Mobile Use
1 – Strongly prefer reCAPTCHA	40%	40%	40%
2 – prefer reCAPTCHA	0%	20%	0%
3 – No Preference	20%	0%	20%
4 – prefer Nouncaptcha	40%	0%	0%
5 – Strongly prefer Nouncaptcha	0%	40%	40%

Due to the small sample size no statistics were evaluated on these data.



This graph shows the performance difference between typical users, and the 4 dropped low performers.

2 points in 5 rounds are required to pass, and 5 prompts were presented in the practice round. An average score of .4 (2/5) is considered passing.

Relevant user feedback:

> "My internet connection went a bit wonky on the last bit so my data may be a bit off "
>
> “I misunderstood the nouncaptcha directions. I thought I was supposed to select every noun and wasn't sure what the picture was for. I read the instructions but don't remember them saying to only pick the nouns associated with the pictures, if that was what I was supposed to do. I just checked every noun.”

(The other 3 users said “Thanks”, “No”, and “No” in the feedback/comments box)