Audio CAPTCHA with a few cocktails: it's so noisy I can't hear you

Benjamin Maximilian Reinheimer[1], Fairooz Islam[2], and Ilia Shumailov[2]

[1] Karlsruhe Institute of Technology
[2] University of Cambridge
name.surname@{kit.edu, cl.cam.ac.uk}

Abstract. With crime migrating to the web, the detection of abusive robotic behaviour is becoming more important. In this paper, we propose a new audio CAPTCHA construction that builds upon the Cocktail Party problem (CPP) to detect robotic behaviour. We evaluate our proposed solution in terms of both performance and usability. Finally, we explain how to deploy such an acoustic CAPTCHA in the wild with strong security guarantees.

Keywords: Acoustic CAPTCHA, cocktail party problem, natural language processing, language comprehension

1 Introduction

ARPANET, a precursor to the modern Internet, was first presented to the public in 1972 at International Computer Communication Conference [37]. A revolutionary application appeared the same year – the email software. It was the first application for people-to-people communication on scale and remained the largest network application for over a decade. A lot has changed since then – a large proportion of the world is now connected and more and more devices are produced with built-in networking capability.

By 1998 it became apparent that criminals have found a way to use connectivity to their advantage [16] and since then the war with spam has begun [25]. Computer abuse, ranging from spam and identity theft to cyberbullying, is a common occurrence in the modern world – by now it inhabits all modern platforms and is largely commoditised [38, 45, 30, 6]. It also scales, as abusers have figured out ways to automate their enterprises.

Completely Automated Public Turing Test To Tell Computers and Humans Apart (CAPTCHA) was created to stop automatic computer service abuse. All CAPTCHAs operate on a simple principle – they use problems that humans are good at and computers struggle to solve. Modern CAPTCHAs are ubiquitous and come in all forms and shapes. Most of them exploit the human ability to recognise objects even when only partial information is available.

When first introduced, CAPTCHAs were simple and imposed low usability costs. So long as image-recognition technology was primitive and Internet crime was still in its infancy, distorted images were sufficient to stop most robots. But as more commerce moved online and CAPTCHA solving was suddenly worth money, solving services started appearing – e.g. *anticaptcha* [44]. In fact, anti-captcha was so popular and so widely used in Russian underground forums, that at one point people started to use its credit as a currency.

Today machine learning software is getting good at image recognition and systems are forced to use many additional markers to identify human behaviour. For example, Google, amongst other things, monitors cursor movement extensively to find automatic behaviour. However, such techniques impose a usability cost: instead of having to solve one simple task as in the early days, today people may be asked to solve a whole series of problems and are not usually given any feedback on why the system doubts their humanity.

Moreover, behavioural factors like cursor tracking are not themselves sufficient to limit automatic computer service abuse; CAPTCHA itself has to evolve too. Abusers collect data over time, allowing them to simulate human-like behaviour and find heuristics to solve tasks that were once hard for them. That, in turn, means that for CAPTCHA to be effective it has to evolve at least as fast as the attacker.

In this paper, we propose a new way to detect robots based on our human ability to separate overlapping human voices – referred to by psychologists as the Cocktail Party Problem. We evaluate our CPP CAPTCHA's performance against the best speech transcribers available currently and run several user studies to explore its usability costs. We discuss it's implications and investigate the naive attack performance. Finally, we describe how to run a cocktail-party CAPTCHA in the real world, and explore security guarantees.

We need new types of defences, and this paper presents one possibility. The remainder of this paper is structured as follows. Section 2 tells the story of CAPTCHAs and describes how our proposal relates to previous work in the field. Section 3 describes the necessary background information and describes conducted experiments. Section 4 evaluates the performance of our audio CAPTCHA mechanism in terms of both usability and protection. Finally, Section 5 explains how one can use it in practice.

2 Related Work

CAPTCHA was invented by von Ahn and Blum, and started being used at scale by websites which were happy for anyone to open an account (e.g. for webmail service) but did not want to let scripts open thousands of accounts [58]. The earliest CAPTCHAs involved reading text from distorted images, but a short

time later, the first non-visual CAPTCHAs were introduced [34, 58].

There is a substantial literature on visual CAPTCHAs and automated algorithms used to break them, with an arms race proceeding through the 2000s [7, 13]. Audio CAPTCHAs were much less used in those early days (e.g. for visually impaired users) and were initially evaluated only for their usability [54, 36]; later they too became the target of attacks [56].

There have been attempts to combine visual and audio CAPTCHAs [29], where either the visual stimuli (images of an animal) or the auditory stimuli (animal sounds) can be recognised; other studies used bird noises and claimed promising results [48]. Another thread of research in the field of psychology exploited the mechanism of background speech, impairing short-term memory performance [20]. CAPTCHA designers have also attempted introducing cognitive complexity. Tam et al. proposed paraphrasing the question or answer to make life more difficult for machines, while still allowing humans to use contextual insight to solve the problem [56].

3 Methodology and Background

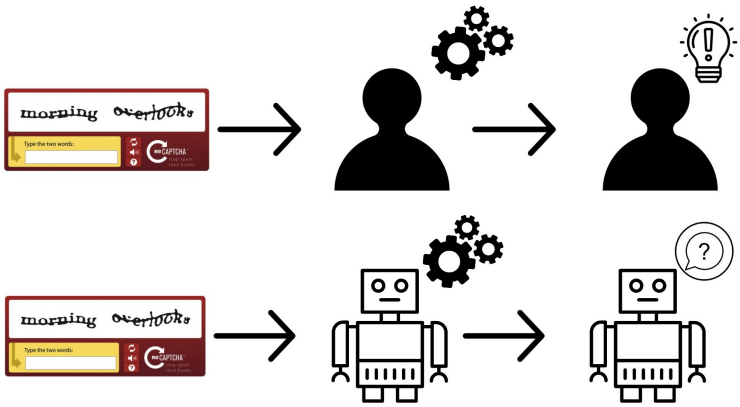

Fig. 1: reCAPTCHA process

3.1 What is a modern CAPTCHA?

Modern CAPTCHAs are trying to balance out four main objectives. First, a CAPTCHA must be solvable for humans; second, a CAPTCHA must be hard for robots to solve; third, it should be hard for the robots to collect system

feedback; and fourth, a CAPTCHA should be able to evolve with little change to the overall experience. These objectives are further discussed below.

Human usability with images and audio Human handling of different types of noise is a well-understood problem, especially in the image classification domain [47, 5, 23]. For example, Geirhos et al. investigated human performance for an object detection task and compared it to the state of the art Deep Neural Networks (DNNs) [23]. Authors find that additional noise degraded performance for both humans and DNNs, but human performance degraded at a lower rate. It should be noted that DNNs were not exposed to the transformations considered by the authors in the training phase. Similarly, human comprehension of speech is well researched [27, 46]. Humans non-linearly weight frequencies of acoustic signals and are better at distinguishing lower frequencies [42]. In practice, that means that humans can comprehend speech quite well, especially if the signal-to-noise (SNR) ratio in lower parts of the spectrum is large enough. It was also found that phase plays an equally important role in human understanding, particularly in low SNR settings [32, 51]. Usability impacts of different CAPTCHA designs have been thoroughly evaluated both in terms of human solution accuracy and response time [12, 48, 59, 60]. Additionally, anecdotal evidence was presented that even simple CAPTCHAs can drive legitimate users away [21].

Robot usability It is not clear what makes a problem hard. von Ahn and Blum, original creators of CAPTCHA, have defined an AI problem to be hard 'if the people working on it agree that it's hard' [2]. They argue that this definition captures the reality, and compare it to canonical cryptographic definitions, where cipher constructions are based on problems that are known to be hard.

Robot data collection As more data gets harvested it becomes easier to use machine learning tools to automatically find solutions for CAPTCHAs. There are multiple ways to make learning harder for the attacker. First, the amount of unique data exposed to the attacker can be reduced. That can be done by supplying the same sample to the attacker, but with varying transformation on top of it. With features of just one sample present, it would learn to recognise that sample, rather than to generalise to a task. Second, noise can be added to the interaction with the CAPTCHA. This additional randomness could affect the convergence of the attacker. Randomness could come in different forms: supplying a sample from a completely different task and observing performance; or randomly passing or failing him. This randomness could also be controlled for a more complex interaction. By adding a skip button, one directly reduces the chances of getting a correct answer and allows for asking of an incorrect question. What is more, there is no simple way to go around it. An attacker needs to add a classifier to the overall system and learn it separately. That is because the skip button cannot be proxied using classification confidence, which was previously shown to be an ineffective metric across tasks [19, 28].

Furthermore, by also asking two CAPTCHAs at the same time an attacker would struggle with attributing errors to made predictions. Finally, data could be supplied to the attacker with pre-defined bias, such that the attack would learn a backdoor [3, 39]. Ultimately, the interaction with an attacker should not be thought of as many individual interactions, but rather it should be viewed as a sequence of decisions.

CAPTCHA evolution It is always a matter of time until CAPTCHA becomes solvable. In practice, that means that CAPTCHA systems should be built in such a way that it can be changed with relative ease, whilst preserving human usability levels. For example, Geirhos et al. noted that DNNs were struggling with unknown types of noise applied to objects for the object recognition task [23]. Evolution against a DNN attacker can then be build on top of that principle – over time applied noise should be changing both in terms of noise distribution and noise magnitude.

CAPTCHA solution costs CAPTCHA solution services are a long established business that had been thoroughly studied both in technological [12] and business aspects [44, 17]. Multiple online services offer services for solving popular CAPTCHAs. As of early June 2020, solving CAPTCHAs does not cost much. Depending on the platform it costs from 0.6$ to 2.5$ for 1000 CAPTCHAs. These services provide almost unlimited bandwidth for solving purely image-based CAPTCHA and from 5 to 7 per minute for more complex behaviour-based ones. Back in 2010, it was reported that those businesses use a hybrid solution approach, i.e. they solve it automatically if CAPTCHA is vulnerable, otherwise, humans from low-cost labour markets are hired [44].

CAPTCHA example Figure 1 shows an example of a CAPTCHA asked by reCAPTCHA, one of the most popular CAPTCHA providers. Here, the user is asked to transcribe two words: 'morning' and 'overlooks'. The CAPTCHA is easy for humans to solve: there are only two main transformations applied to the images and text occlusion is minimal. According to the principals described in Section 3.1, it is easy to evolve such a CAPTCHA in response to improving attacker performance: other transformations could be applied, and the dictionary could be expanded to include more special characters. Finally, the data collection principle described in Section 3.1 is also followed. The robot is not provided feedback on the performed transcriptions and, if a mistake is made, it would not know which of two words it was mistaken about [3].

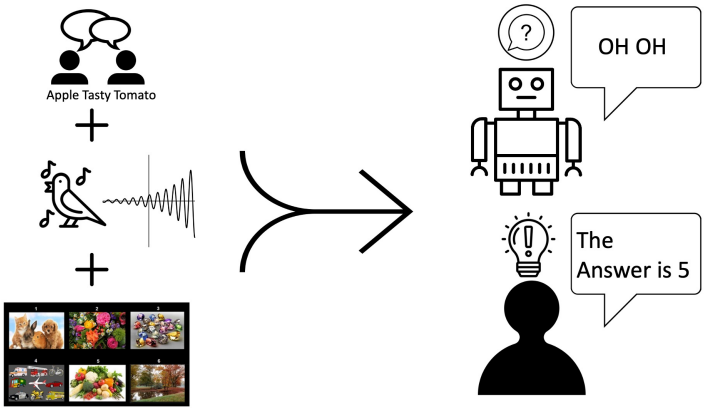

Fig. 2: CPP CAPTCHA process

3.2 What is a cocktail party problem and why does it work?

Humans are much better than machines at disambiguating a single speaker from a group of people speaking at once [26]. That phenomenon is commonly referred to as a Cocktail Party Problem. In essence, it refers to cases where one or more voices are talking concurrently with the speech of interest. Background noise, consistent of natural human speech, serves as a form of semantic noise which should hinder human understanding less than it hinders machines. The Cocktail Party effect has already been investigated from numerous angles: from the general phenomenon [14], the cues that impact effectiveness [33, 43], over the influence of working memory capacity [15] to the intentional control of auditory selective attention [24, 50]. In this paper, we propose an acoustic CAPTCHA construction based on the Cocktail Party Problem phenomenon.

3.3 CPP CAPTCHA

We propose to use the Cocktail Party Problem to build a robust acoustic CAPT–CHA system as is depicted in Figure 2. The problem itself can be formulated as follows. A user is provided a challenge speech sample M and a question q, such that $M = S_{orig} + \sum_{i=0..n} S_i + \sum_{j=0..m} N_j$, where S_{orig} is the speech of interest, $S_0..S_n$ are background speech samples and $N_0..N_m$ are non-speech noise samples. Question q is formulated in such a way that humans will be able to semantically extract S_{orig} from M, whereas computers will struggle to do so. Do note here, that question q can make semantic references to both speech background samples and noise. For example, the user may be asked "How many times did a bird sing after word 'cat' was said by a female voice?". The user is asked

[3] reCAPTCHA uses two words, out of which the correct solution is known for one [44]. The correctness is assessed based on the editing distance of the provided solution and the control word.

to solve a tuple (M, q) and respond to the CAPTCHA system.

The construction described above is generic and encompasses a lot of different possibilities. In this paper, we describe and evaluate three different CPP CAPTCHA construction possibilities. We use speech signals comprised of either numbers, digits or individual words as both signals of interest and background speech. We assume that there is only a single speaker in the background S_0 and background noise is either no noise, bird singing, elephant sounds or white noise. Before starting the CAPTCHA the user gets question q explaining which voice he has to focus on: in this paper, we used the speaker's gender as the semantic information. Figures 3 to 5 display the introduction examples of our study, where participants could repeat the CAPTCHA and see the actual solution.

First we start with the digit CAPTCHA, where two sets of 6 to 9 digits were read out by a female or male voice. The introduction screen is presented in Figure 3. While listening to the CAPTCHA the user had to focus on the specified voice and enter the corresponding digits.

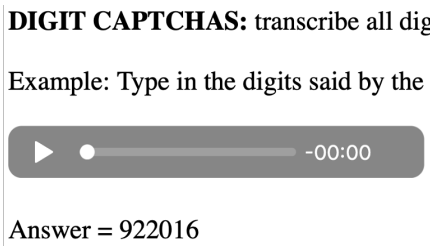

Fig. 3: Example interface for digit CAPTCHAs

The second type is the character CAPTCHA, where two sets of 6 to 9 characters were read out by a female or male voice. Similarly to digits, participants were presented with an introduction screen as depicted in Figure 4. While listening to the CAPTCHA the user had to focus on the specified voice and enter the corresponding characters.

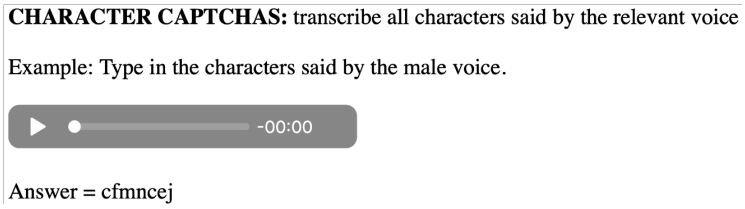

Fig. 4: Example interface for character CAPTCHAs

Finally we used a word CAPTCHA, where two sets of 3 to 6 English words were read out by a female or male voice. The introduction screen is presented in Figure 5. The words in each set are related to one of a given set of images. After listening to the CAPTCHA, the user had to select the relevant image for the specified voice. For example, the participant would be told 'apple tasty tomato broccoli carrot cucumber' with a male voice, and a female voice would be saying 'rose nature tulip daisy buttercup'. The user would then be expected to pick the picture with vegetables (5) for male voice and picture with flowers (2) for female voice in Figure 5.

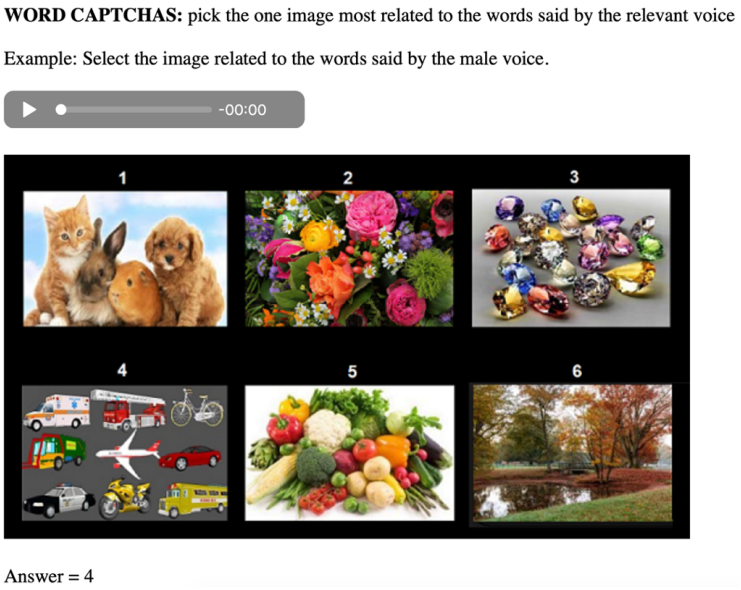

Fig. 5: Example interface for word CAPTCHAs

3.4 Ethics

The overall study is separated into two parts. First, we ran a preliminary study of the audio CAPTCHA mechanism. The participants were informed of their rights and both verbal and written consent was collected. As the study included extensive user interaction, we followed university guidelines and acquired ethics approval from the University of Cambridge Computer Laboratory Ethics Board. We made sure that the participants were not harmed in any way and no sensitive information was collected.

For the second usability evaluation, we followed the ethic guidelines of the Karlsruhe Institute for Technology. We made sure that all collected data followed uni-

versity guidelines on ethical data handling and the most recent GDPR policies. As a study platform, we used SoSci Survey [4], that is compliant with the European Data Protection Regulations. As a recruitment platform, we used Clickworker [5]. Participants were clearly explained what the study was about and it's purpose. Expectations were set out and no deception was used. Participants were told that they could terminate their participation at any point without providing reasons and without any negative consequences. Their participation was 100% voluntary and payment of €3 per participant for 50 participants was provided.

3.5 Experiments

The aim of the experiments is two-fold. First, we aim to assess the user-friendliness of different CAPTCHA schemes to allow comparison. Second, based on the results acquired, we identify what features make audio CAPTCHAs user-friendly. For this, we use conventional usability evaluation following the ISO 9241-11 [1] definition of usability. The ISO standard has three main components:

- **Effectiveness:** the audio CAPTCHA is unambiguous and therefore easy to use;
- **Efficiency:** the audio CAPTCHA is solved a high percentage of the time and in as short a time as possible;
- **Satisfaction:** the audio CAPTCHA triggers a high level of satisfaction among the users, i.e. they should be satisfied and motivated to continue using it in the future.

Most people today are familiar with visual CAPTCHAs, as they are ubiquitous and are used to deter robot activity practically everywhere online. Although the users can occasionally be annoyed at CAPTCHAs, they are largely accepted. We aim to develop an audio CAPTCHA scheme that is at least as good in terms of usability. It should be noted, however, that people have been exposed to visual CAPTCHAs for the past 20 years and it is hard to reproduce the same learning artefacts. Measurements are collected in the following three forms:

- **Effectiveness:** Number of failed attempts at the audio CAPTCHA;
- **Efficiency:** Repetitions and duration until successful completion;
- **Satisfaction:** SUS Questionnaire [4, 10, 11, 49].

3.6 Experimental Process

The study evaluates three forms of CAPTCHAs. The whole process is depicted in Figure 6. Each type represents a different aspect of perception: numbers, letters, and entire words. First, the participants receive Informed Consent to read and accept. We then explain the procedure of the study. The participants are

[4] SoSci is a platform designed for running experiments https://www.soscisurvey.de
[5] Clickworker is a platform similar to MTurk for finding participants in Europe https://www.clickworker.de

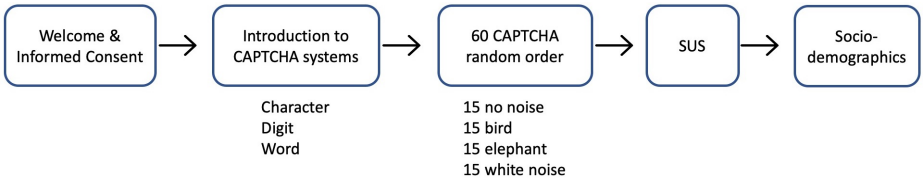

Fig. 6: Usability experiment pipeline

informed that participation can be terminated at any time without any consequences. After that, the participants are shown an example for each of the three CAPTCHA types. We explain how to deal with every type of CAPTCHA and what the right solution is. The participants can listen to the CAPTCHA as often as they want until they feel confident in using it. Each CAPTCHA, regardless of type, consists of two voices – one female and one male. It is accompanied by a description such as:

- **Description 1:** "Transcribe all digits said by the relevant voice - example: type in the digits said by the female voice";
- **Description 2:** "Transcribe all characters said by the relevant voice - example: type in the characters said by the female voice";
- **Description 3:** "Word CAPTCHAs: pick the one image most related to the words said by the relevant voice - select the image related to the words said by the male voice."

Then the entire sequence of the 60 CAPTCHAs is presented randomly. To avoid framing and cognitive load effects, the sequence of CAPTCHAs gets reshuffled for every subject. Those CAPTCHAs are categorised based on their type: number, character, and word. Once participants finish solving CAPTCHAs, they get to the usability questionnaire. The questionnaire follows the standard usability guidelines of System Usability Score. To avoid framing and order effects, the order of questions is randomised. Finally, the study concludes with sociodemographic questions on gender, age, and highest educational achievement.

4 Evaluation

4.1 Usability

For the analysis of usability, we consider four different factors. First, we look at the primary total **number of correctly solved** CAPTCHAs. In each case, we distinguish the **noise type**. Then, to assess **order effects**, we split the CAPTCHAs into three groups depending on the period in which they have been solved. The participants have seen a total of 60 CAPTCHAs, we split the three groups into equal-sized bins: 1st – 20th, 21st – 40th, and 41st – 60th. By evaluating the recognition rate in such a way we can assess the learning effect. Finally, we consider the **number of errors**. Consequently, we distinguish how the rate

Table 1: Percentage of correct answers for CAPTCHA and noise types

Noise Type	Digits ± std	Characters ± std	Word ± std
None	**85.71%** 15.94	26.53% 18.43	68.57% 17.32
Bird	75.51% 20.11	18.78% 14.95	79.59% 18.93
Elephant	73.47% 23.23	**39.59%** 23.63	**81.22%** 19.75
White	57.55% 25.37	20.82% 17.78	48.98% 24.17
Overall	71.80% 21.17	26.60% 18.7	69.90% 20.04

of CAPTCHA recognition changes if one or two errors are allowed. Finally, the System Usability Scale values (SUS) are analyzed to assess the subjective perception of the CAPTCHAs.

First, we turn to the number of correctly recognized CAPTCHAs. It can be seen in Table 1 that the character-based CAPTCHAs have a significantly worse recognition rate (mean = 26.60%) when compared to both word (mean = 69.90%) and digit-based (mean = 71.80%) CAPTCHAs. The overall best performance was observed for digit-based CAPTCHAs without noise with 85.71% recognition rate, and the worst recognition was observed for character-based CAPTCHAs with bird noise with 18.78% recognition rate. Interestingly, we observe that in some cases participants recognise characters and words better in the presence of noise, than without it. We have two hypotheses regarding why this happens. First, it might be the case that participants focus more in the presence of noise. Given CPP CAPTCHA is a low to medium complexity task, a possible explanation could be found with the Yerkes-Dodson Law. It was previously found that a certain level of arousal is beneficial for task performance, and additional noise could just trigger that response [18, 9, 57, 22]. Second, it might be the consequence of a shuffling procedure – some letters are harder to recognise when they overlap. The sequences were randomised for every participant and type of CAPTCHA. We have not controlled in the experiments that the overlaps are consistent across participants, and that could have affected the results. In the subsequent studies, both of those factors should be controlled for.

Table 2: Correct answers per position asked during the study

Type	1 to 20	21 to 40	41 to 60
Digits	4.13	**5.07**	**5.41**
Characters	1.67	1.88	2.04
Word	**4.33**	4.98	4.61
Overall	3.37	3.97	4.02

Next, we turn to the order effects. Table 2 shows how recognition rate changes over time. Here, it is noticeable that for all three types there is an increase in the number of correct answers over time. For all of the considered cases, except for word-based, the increase is observed through all of the time-periods. For word-based CAPTCHAs, improvement is only observed after the first period. Finally, we note that only in the digit-based CAPTCHAs, there is a significant increase for both transitions with p = .036, T =2.162 and p = .007, T = 2.835.

Table 3: Correct answers per error tolerance

Type	0 Error	1 Error	2 Errors
Digits	**71%**	**91%**	**96%**
Characters	26%	65%	86%
Overall	48%	78%	91%

In the third step, we consider how solvable the CAPTCHAs are when a participant is allowed to make a certain number of mistakes when making their transcription. Table 3 shows the results of the experiment. Note that we only consider the character- and digit-based CAPTCHA types here as word constitutes a binary choice. We find that participants were solving 71% of digit-based CAPTCHAs correctly without any errors, 91% with 1 error allowed and 96% with 2 errors allowed. Here, a significant performance increase of 20% is observed with just a single misclassification. An even more significant improvement is observed for character-based CAPTCHAs. Only 26% of CAPTCHAS are solved without any errors, but a single misclassification improved performance by almost 40% to 65%. With 2 errors allowed, we observe that performance improved by a further 20% up to 86%. Note here how close the human performance is for digit and character-based CAPTCHAs in the presence of just two errors. The complexity of digit and character tasks are extremely different – random guess probability of $\frac{1}{10^n}$ for n digits and $\frac{1}{26^n}$ for n characters.

Finally, Table 4 presents the SUS values for different types of CAPTCHAs. It appears that for different CAPTCHA types there is no significant difference in terms of usability. For all three types, the mean values are in the range of about 45, with a minimum of around 30 and a maximum of 70. We find that acoustic CPP CAPTCHA performs consistently worse in terms of usability compared to the visual-based ones. Yet, the difference to the closest contender – reCaptcha v2 – is not too large. reCAPTCHA is a very widely used system, meaning it is really hard to control for learning effects. Moreover, Jiang et al. specifically recruited active web users [31]. Given that participants of our study encountered CPP CAPTCHA for the first time, we believe that with more careful design and control for learning effects the usability could be further improved.

Table 4: SUS Score for CAPTCHA types

Type	Mean	SD	Min	Max
Digits	**46.17**	**9.07**	27.5	**70.0**
Characters	44.54	8.25	27.5	67.5
Word	45.56	7.82	**32.5**	**70.0**
Overall	45.42	8.38	29.17	69.17
TapCHA [31]	85.0	-	-	-
Web Interfaces [4]	68.2	-	-	-
Cell Phones Interface [4]	65.9	-	-	-
ReCAPTCHA v2 [31]	65.0	-	-	-

4.2 Naive transcription performance

Now we turn to the adversarial evaluation. In this paper, we assume a naive attacker that is ignorant of the existence of the background noise in the sample. Although naive, that scenario represents a realistic case of a scalable attack in the real-world. For example, it was previously shown that acoustic reCAPTCHA can be easily solved using Google Cloud Speech out of the box with little to no modifications [8]. For the study, we chose Sphinx and Google Cloud Speech as benchmarks. To make it easier for the transcribers, unless stated otherwise, we used no additional noise and clean generated speech. We have attempted using recorded speech and got similar performance as is shown in Figure 7. We decided to use generated speech as it presents an idealistic clean scenario. We turn to the case of a more skilful attacker in Section 5.

Figures 8 to 10 show the recognition rate for text generated with MaryTTs, IBM Text-to-Speech, and Google Text-to-Speech. As text transcription tools we chose Google Cloud Speech and Spinx. Overall, we observe that simply using transcribers does not work any more – the recognition rate for digits and characters is consistently low, even when reduced dictionaries are used.

First, we turn to per-character recognition rates which are shown in Figure 8. Similarly, the character CAPTCHA recognition rates are low. Here, we further evaluate the character recognition rates for the speech of different genders. Interestingly, we find that when faced with interleaved voices, both Google Cloud Speech and Sphinx appear to choose male voices over female voices.

Digit CAPTCHA recognition rate is relatively higher than that for characters. We observe that Sphinx with a reduced dictionary is capable of recovering from 60% to 95% of digit sequences correctly, yet full dictionary approaches perform worse. Similarly to characters, Google Cloud Speech favours male voices over female, yet conflicting evidence is observed for Sphinx. The detection rate for per-digit recognition with source-separated IBM and recorded digit CAPTCHAs

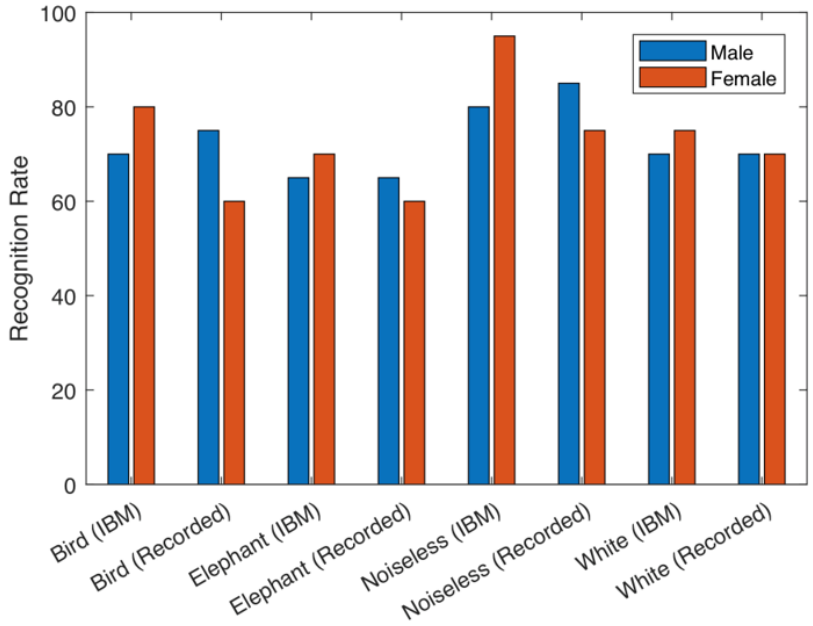

Fig. 7: Per-digit recognition rates for source-separated IBM and recorded digit CAPTCHAs, using a digit dictionary for Sphinx

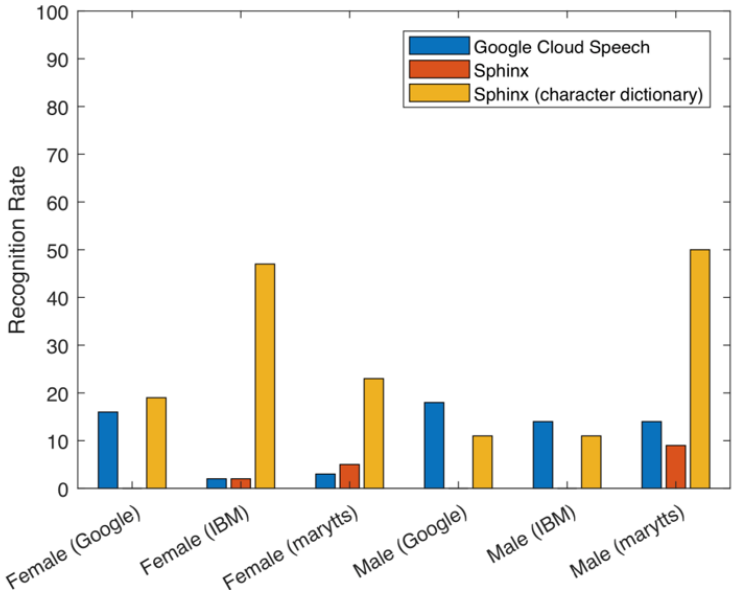

Fig. 8: Per-character recognition rate for character CAPTCHAs

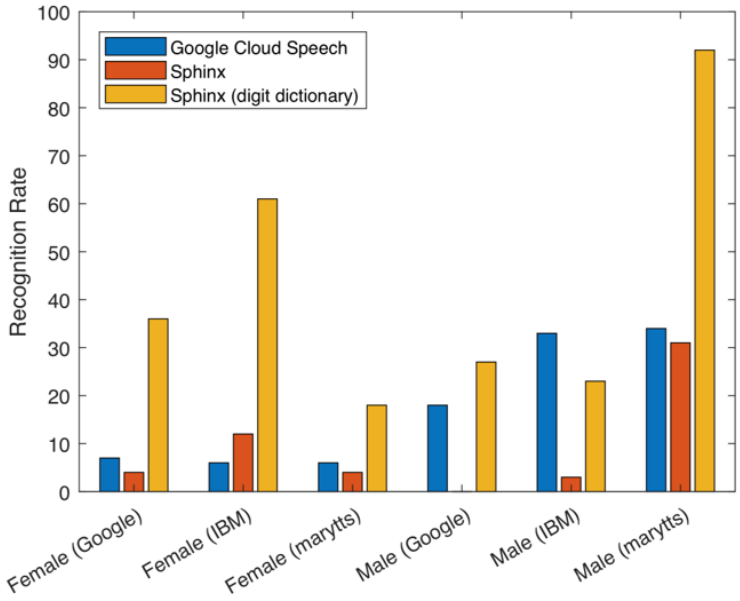

Fig. 9: Per-digit recognition rate for digit CAPTCHAs

ranges from a minimum of 60% for female voices with bird noises to close to 100% for noiseless scenarios. The per-word recognition rate for character CAPTCHAs ranges from 0% with Male speech produced by Google Text-to-Speech for Sphinx, up to 75% for Male voices produced by IBM Text-to-Speech for Google Cloud Speech.

This observation is also reflected in the recognition rates for Sphinx as is shown in Figure 7, which is much higher than the mixed recognition rates for both synthesised and recorded speech. However, recorded speech still seems to be much more resistant than synthetic speech.

To conclude, per-character, per-digit and per-word recognition rates are relatively poor. That is not surprising – the transcribers were not designed to solve multi-speaker situations. We do observe that digits are handled relatively better, which is consistent with the observation that it is the easiest of three tasks to solve complexity-wise.

Interestingly, we observe inconsistencies in the way that multi-speaker speech is handled. We see that different transcribers, when faced with a choice, consistently pick male voices over female. That finding suggests that similarly to other natural language models, transcribers used in this paper overfit to the dataset gender biases [41, 55].

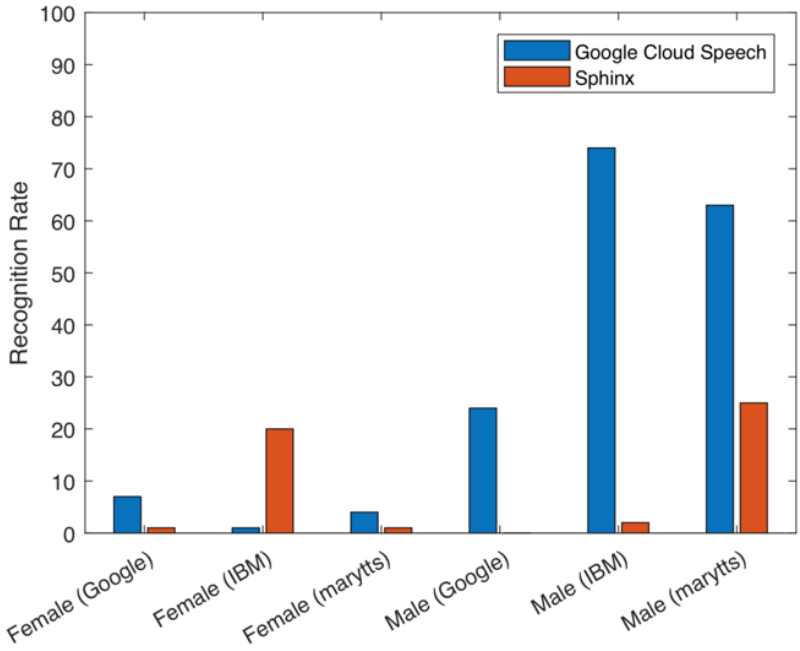

Fig. 10: Per-word recognition rate for character CAPTCHAs

5 Conclusion and Discussion

In this paper, we have presented Cocktail Party Problem-based CAPTCHA construction. We have evaluated its performance in terms of usability and robustness against modern transcribers. We have observed that the Cocktail Party Problem does have an effect on the way transcription works, practically making it impossible for transcribers to be used out of the box. Interestingly, we observed that when faced with overlapping voices, transcribers have a gender bias, consistently picking male voices over female. Finally, we ran a user study to evaluate proposed CAPTCHA usability. We observe that participants could successfully solve it without prior experience and were getting better over time. We find that the usability scores were lower than the ones for the textual CAPTCHAs. Interestingly we observe that the closest contender is reCaptcha v2 – one of most commonly used visual-based CAPTCHAs – with a SUS difference between the two of 20. That, in turn, suggests that with more careful design and control for learning effects the usability of audio CAPTCHA based on the Cocktail Party Problem can be further improved.

CPP CAPTCHA in the real world CAPTCHA systems are not impenetrable. In the past, they have set off an arms race with new attacks and schemes

being proposed several times a year. The same is to be expected for any new CAPTCHA that finds its way into use.

Same holds for the CAPTCHA construction described in this paper. Recently, a machine learning-based solution to the Cocktail Party Problem was proposed. In particular, Simpson described how one might go about generating a neural network that separates the speakers in a given audio file, with a network trained on the dataset of separated audio files [53]. The mechanism is based on the idea of binary mask generation and the network itself learns the relationship between the different frequencies of a human speaker.

We re-implemented Simpson's paper and evaluated the performance of our CAPTCHA on a synthesised speaker against his CPP solver and standard text-to-speech transcribers. Using the method, we successfully managed to separate the speakers in the audio files and the speech to text system, and have been able to get the original text with 95% accuracy against almost 0% with CPP. Although a voice synthesiser represents an idealistic scenario (very idealistic scenario with perfect information and practically no task associated) of the attack with a very well-defined speaker, it is still representative of the machine learning ability to solve the problem.

Despite the success of the attack described above, we believe that the proposed solution is still deployable in the wild. We have seen successful attacks on image-based CAPTCHAs that ask a human to solve classification tasks, that we know computers solve much better [35]. There are several reasons why these solutions remain relevant. First and foremost, despite machine learning being able to perform a task very well, it can do so only for a local problem. Cloudflare simply changes the classification task every now and then, forcing the adversaries to either collect a new dataset or learn the task in an online manner. Both of those problems are already extremely hard but are made even harder with the help of a few heuristics. For example, if you suspect that the attacker is trying to do online learning you start to misguide that learning by saying that correct guesses are wrong and the wrong ones are correct. This will also allow the defender to embed trojans into the dataset [40]. Or one can simply ask the robot to solve multiple CAPTCHAs and not report to it if it solved any of them. Second, in practice, there is a lot more then just a single CAPTCHA, behavioural profiles are being built of an individual interacting with a system, which can also be used to detect fraudulent behaviour. For example, with audio one knows the minimum amount of time it should take for a user to solve a task. Similarly, one can also construct samples that she expects machine learning to solve slowly and measure how long it took the model to answer [52].

Visual and Language-based CAPTCHAs When compared to traditional visual CAPTCHA systems, language-based acoustic CAPTCHA represents a much richer interaction environment. Even if an attacker manages to learn to

solve a local Cocktail Party Problem, it would still not destroy the CPP construction. First, the decision space for the attacker is still a lot larger than in the case of object detection. Language is a lot more discrete and interactions are a lot more subtle. Especially with audio in the analogue form – numerous distinct transformations can be used to diffuse human speech. There is little to no physical limitation on the way language is comprehended by humans, whereas there are large physical limitations to picture representations. Second, both natural language and acoustic natural language tasks are not yet solved by the state of the art machine learning. Although there have been models built which perform consistently well on a number of benchmarks, there is still no model available that is capable of solving all of the natural language comprehension tasks and quickly change between them. Third, most modern language models largely limit the language space with which they work. That is either done through reducing dictionary sizes or controlling the embedding space size. Performance is the issue here: if models are too complex or large they either do not fit in memory, take too long to train or have very large latency. In practice, that reduction is affordable for natural language tasks because models still extract information from unknown words by approximating them with known words. Same does not hold true for language-based CAPTCHAs, as the precision of the answer here is paramount. Finally, language and speech are really easy to change and numerous questions can be asked about the exchange of multiple speakers in an audio sample, with each one of those questions being of a different size. That makes the evolutionary principle of CAPTCHA construction particularly strong.

Future research directions For future research, several approaches are worth pursuing. For example, in the field of word CAPTCHA, there is the possibility to use it entirely as an audio CAPTCHA and to design the response via audio playback. Furthermore, similar to visual CAPTCHAs, users could be allowed to select more than one image. Different languages could be used together for CPP construction to target multi-language or non-native speakers. Motoyama et al. previously categorised the manual labour pool of CAPTCHA solving services using CAPTCHAs in different languages [44]. The authors find significant differences in CAPTCHA service performance, suggesting that language mixture might be a potential way to control CAPTCHA complexity and protect against those services. reCAPTCHA already used a Navajo language speech as background noise, here we propose to extend it to many more languages [60]. Furthermore, the current study could be repeated with non-synthetic voices to evaluate the influence of human speech on performance and usability. The voices could also be modified in different facets, e.g. prosodic elements such as pitch and rate of speech. In the field of study design, users could be given the possibility to play CAPTCHA as often as possible. It would be interesting to see how the performance of robots changes compared to that of humans as they learn. Furthermore, the type of questions could be varied by linking them to the background noise, for example, “how often could you hear a bird in the back-

ground?". Finally, some CAPTCHAs showed apparent learning effects for the participants. Therefore, it would be interesting to conduct a longitudinal study to check if the familiarity with a CAPTCHA affects usability.

Acknowledgements

We thank Darija Halatova, Dmitry Kazhdan and Ross Anderson (all affiliated with Cambridge University) for valuable discussions and suggestions. Conducted research was partially supported with funds from Bosch-Forschungsstiftung im Stifterverband.

References

1. ISO 9241-11:2018(en) (2018), last accessed 18 December 2018
2. von Ahn, L., Blum, M., Hopper, N., Langford, J.: CAPTCHA: Using Hard AI Problems for Security. In: EUROCRYPT (2003)
3. Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., Shmatikov, V.: How to backdoor federated learning. arXiv preprint arXiv:1807.00459 (2018)
4. Bangor, A., Kortum, P., Miller, J.: Determining what individual SUS scores mean: Adding an adjective rating scale. Journal of Usability studies (2009)
5. Banko, E., Kortvelyes, J., Weiss, B., Vidnyanszky, Z.: How the Visual Cortex Handles Stimulus Noise: Insights from Amblyopia. PLOS ONE (2013)
6. Bhalerao, R., Aliapoulios, M., Shumailov, I., Afroz, S., McCoy, D.: Mapping the Underground: Supervised Discovery of Cybercrime Supply Chains. In: 2019 APWG Symposium on Electronic Crime Research (eCrime) (2019)
7. Bigham, J.P., Cavender, A.C.: Evaluating existing audio CAPTCHAs and an interface optimized for non-visual use. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM (2009)
8. Bock, K., Patel, D., Hughey, G., Levin, D.: unCaptcha: a low-resource defeat of recaptcha's audio challenge. In: 11th {USENIX} Workshop on Offensive Technologies ({WOOT} 17) (2017)
9. Broadhurst, P.: The interaction of task difficulty and motivation: The yerkes dodson law revived. Acta Psychologica, Amsterdam (1959)
10. Brooke, J.: SUS: a retrospective. Journal of usability studies (2013)
11. Brooke, J., et al.: SUS – A quick and dirty usability scale. Usability evaluation in industry (1996)
12. Bursztein, E., Bethard, S., Fabry, C., Mitchell, J.C., Jurafsky, D.: How good are humans at solving CAPTCHAs? a large scale evaluation. In: 2010 IEEE symposium on security and privacy (2010)
13. Chellapilla, K., Larson, K., Simard, P., Czerwinski, M.: Designing human friendly human interaction proofs (HIPs). In: Proceedings of the SIGCHI conference on Human factors in computing systems. ACM (2005)
14. Cherry, E.C.: Some experiments on the recognition of speech, with one and with two ears. The Journal of the acoustical society of America (1953)
15. Conway, A.R., Cowan, N., Bunting, M.F.: The cocktail party phenomenon revisited: The importance of working memory capacity. Psychonomic bulletin & review (2001)

16. Cranor, L.F., LaMacchia, B.A.: Spam! Communications of the ACM (1998)
17. Danchev, D.: Inside India's CAPTCHA solving economy (2020), https://www.zdnet.com/article/inside-indias-captcha-solving-economy/
18. Denenberg, V.H., Karas, G.G.: Supplementary report: The yerkes-dodson law and shift in task difficulty. Journal of experimental psychology (1960)
19. DeVries, T., Taylor, G.W.: Learning confidence for out-of-distribution detection in neural networks. arXiv preprint arXiv:1802.04865 (2018)
20. Ellermeier, W., Kattner, F., Ueda, K., Doumoto, K., Nakajima, Y.: Memory disruption by irrelevant noise-vocoded speech: Effects of native language and the number of frequency bands. The Journal of the Acoustical Society of America (2015)
21. Elson, J., Douceur, J.R., Howell, J., Saul, J.: Asirra: a CAPTCHA that exploits interest-aligned manual image categorization. (2007)
22. Gawron, V.J.: Performance effects of noise intensity, psychological set, and task type and complexity. Human factors (1982)
23. Geirhos, R., Janssen, D.H.J., Schütt, H.H., Rauber, J., Bethge, M., Wichmann, F.A.: Comparing deep neural networks against humans: object recognition when the signal gets weaker. arXiv preprint arXiv:1706.06969 (2017)
24. Getzmann, S., Jasny, J., Falkenstein, M.: Switching of auditory attention in "cocktail-party" listening: ERP evidence of cueing effects in younger and older adults. Brain and cognition (2017)
25. Goodman, J., Cormack, G.V., Heckerman, D.: Spam and the ongoing battle for the inbox. Communications of the ACM (2007)
26. Handel, S.: Listening: An Introduction to the Perception of Auditory Events. MIT Pres (1993)
27. Heiko, P., Meine, N., Edler, B.: Sinusoidal coding using loudness-based component selection. IEEE International Conference on Acoustics, Speech, and Signal Processing (2002)
28. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. arXiv preprint arXiv:1610.02136 (2016)
29. Holman, J., Lazar, J., Feng, J.H., D'Arcy, J.: Developing usable CAPTCHAs for blind users. In: Proceedings of the 9th international ACM SIGACCESS conference on Computers and accessibility. ACM (2007)
30. Holt, T.J., Smirnova, O., Chua, Y.T.: The Marketing and Sales of Stolen Data (2016)
31. Jiang, N., Dogan, H., Tian, F.: Designing mobile friendly CAPTCHAs: an exploratory study. In: Proceedings of the 31st British Computer Society Human Computer Interaction Conference (2017)
32. Kim, D.S.: Perceptual phase quantization of speech. IEEE Transactions on Speech and Audio Processing (2003)
33. Koch, I., Lawo, V., Fels, J., Vorländer, M.: Switching in the cocktail party: Exploring intentional control of auditory selective attention. Journal of Experimental Psychology: Human Perception and Performance (2011)
34. Kochanski, G., Lopresti, D., Shih, C.: A reverse turing test using speech. In: Seventh International Conference on Spoken Language Processing (2002)
35. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems (2012)
36. Lazar, J., Feng, J., Brooks, T., Melamed, G., Wentz, B., Holman, J., Olalere, A., Ekedebe, N.: The SoundsRight CAPTCHA: an improved approach to audio human interaction proofs for blind users. In: Proceedings of the SIGCHI conference on human factors in computing systems. ACM (2012)

37. Leiner, B.M., Cerf, V.G., Clark, D.D., Kahn, R.E., Kleinrock, L., Lynch, D.C., Postel, J., Roberts, L.G., Wolff, S.: A brief history of the internet. ACM SIGCOMM Computer Communication Review (2009)
38. Levchenko, K., Pitsillidis, A., Chachra, N., Enright, B., Felegyhazi, M., Grier, C., Halvorson, T., Kanich, C., Kreibich, C., Liu, H., McCoy, D., Weaver, N., Paxson, V., Voelker, G.M., Savage, S.: Click Trajectories: End-to-End Analysis of the Spam Value Chain. In: 2011 IEEE Symposium on Security and Privacy (2011)
39. Liao, C., Zhong, H., Squicciarini, A., Zhu, S., Miller, D.: Backdoor embedding in convolutional neural network models via invisible perturbation. arXiv preprint arXiv:1808.10307 (2018)
40. Liu, Y., Ma, S., Aafer, Y., Lee, W.C., Zhai, J., Wang, W., Zhang, X.: Trojaning attack on neural networks (2017)
41. Lu, K., Mardziel, P., Wu, F., Amancharla, P., Datta, A.: Gender bias in neural natural language processing. arXiv preprint arXiv:1807.11714 (2018)
42. Mermelstein, P.: Distance measures for speech recognition, psychological and instrumental. Pattern recognition and artificial intelligence (1976)
43. Moray, N.: Attention in dichotic listening: Affective cues and the influence of instructions. Quarterly journal of experimental psychology (1959)
44. Motoyama, M., Levchenko, K., Kanich, C., McCoy, D., Voelker, G.M., Savage, S.: Re: CAPTCHAs-Understanding CAPTCHA-Solving Services in an Economic Context.
45. Motoyama, M., McCoy, D., Levchenko, K., Savage, S., Voelker, G.M.: An Analysis of Underground Forums. In: Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference (2011)
46. Paliwal, K.K., Alsteris, L.: Usefulness of Phase Spectrum in Human Speech Perception. In: Proc. Eurospeech (2003)
47. Rajashekar, U., Bovik, A.C., Cormack, L.K.: Visual search in noise: Revealing the influence of structural cues by gaze-contingent classification image analysis. Journal of Vision (2006)
48. Sauer, G., Hochheiser, H., Feng, J., Lazar, J.: Towards a universally usable CAPTCHA. In: Proceedings of the 4th Symposium on Usable Privacy and Security (2008)
49. Sauro, J.: Measuring usability with the system usability scale (sus) (2011)
50. Scharf, B.: On hearing what you listen for: the effects of attention and expectancy. Canadian Psychology (1990)
51. Shi, G., Shanechi, M.M., Aarabi, P.: On the importance of phase in human speech recognition. IEEE Transactions on Audio, Speech, and Language Processing (2006)
52. Shumailov, I., Zhao, Y., Bates, D., Papernot, N., Mullins, R., Anderson, R.: Sponge Examples: Energy-Latency Attacks on Neural Networks (2020)
53. Simpson, A.J.: Probabilistic binary-mask cocktail-party source separation in a convolutional deep neural network. arXiv preprint arXiv:1503.06962 (2015)
54. Soupionis, Y., Gritzalis, D.: Audio CAPTCHA: Existing solutions assessment and a new implementation for voip telephony. Computers & Security (2010)
55. Sun, T., Gaut, A., Tang, S., Huang, Y., ElSherief, M., Zhao, J., Mirza, D., Belding, E., Chang, K.W., Wang, W.Y.: Mitigating gender bias in natural language processing: Literature review. arXiv preprint arXiv:1906.08976 (2019)
56. Tam, J., Simsa, J., Huggins-Daines, D., Von Ahn, L., Blum, M.: Improving audio captchas. In: Symposium On Usable Privacy and Security (SOUPS) (2008)
57. Teigen, K.H.: Yerkes-dodson: A law for all seasons. Theory & Psychology (1994)
58. Von Ahn, L., Blum, M., Langford, J.: Telling humans and computers apart automatically. Communications of the ACM (2004)

59. Wang, S.Y., Bentley, J.L.: CAPTCHA challenge tradeoffs: Familiarity of strings versus degradation of images. In: 18th International Conference on Pattern Recognition (ICPR'06). IEEE (2006)
60. Yan, J., El Ahmad, A.S.: Usability of CAPTCHAs or usability issues in CAPTCHA design. In: Proceedings of the 4th symposium on Usable privacy and security (2008)