

HuMan: an accessible, polymorphic and personalized CAPTCHA interface with preemption feature tailored for persons with visual impairments

K. S. Kuppusamy¹ · G. Aghila²

Published online: 2 August 2017
© Springer-Verlag GmbH Germany 2017

Abstract Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA) is one of the major security components in the provision of fair web access by differentiating human access from malicious, automated access by bots. Though the CAPTCHA strengthens the security aspect of web access, their accessibility to people with visual impairments has inherent unresolved challenges. This paper presents an accessible CAPTCHA model termed HuMan (human or machine?) which aims at providing an audio-based CAPTCHA for people with visual impairments. The HuMan model incorporates personalization into the CAPTCHA access. The polymorphic nature of resolving the HuMan CAPTCHA facilitates kaleidoscopic behavior in CAPTCHA rendering. The presence of ambient noise and requirement of common sense knowledge to answer the questions presented by HuMan CAPTCHA model makes it friendlier toward human users. The HuMan model has a CAPTCHA preemption feature which enables the user to stop the challenge audio as soon as the answer is identified. The results of experiments conducted on the prototype implementation of HuMan model project the mean success rate of 92.46 % and system usability scale score of 82.44 for persons with visual impairments and 82.63 for sighted users.

Keywords Web accessibility · Accessible CAPTCHA · Non-visual access · CAPTCHA preemption

1 Introduction

The power of the World Wide Web heavily relies on the universality of its access which certainly includes persons with disabilities [6]. The World Health Organizations (WHO) report on disability has identified that 15% of the world population has some sort of disability. Among these disabilities, it has been estimated that around 285 million people are experiencing visual disabilities, either being blind or having low vision¹. All these staggering numbers emphasize the importance of enhancing access to the World Wide Web by persons with disabilities in general and visually impaired in particular. To satisfy this need, the World Wide Web Consortium has taken up the Web Accessibility Initiative which has provided various guidelines on making the web accessible to everyone. The updated Web Content Accessibility Guidelines (WCAG 2.0) and Accessible Rich Internet Applications (WAI-ARIA) provides detailed insights into enhancing the accessibility aspect of web interfaces. Though these recommendations have raised awareness among the web content providers and interface designers, there exist many unresolved issues, with respect to the accessibility of the web by people with disabilities [29]. The design of both hardware and software interfaces tailored for people with special needs such as disabled, elderly and low-literacy people has been an active research field with contributions in various dimensions [3, 15, 25].

✉ K. S. Kuppusamy
kskuppu@gmail.com
G. Aghila
aghilaa@gmail.com

¹ Department of Computer Science, Pondicherry University, Pondicherry, India

² Department of Computer Science and Engineering, National Institute of Technology Puducherry, Karaikal, India

¹ <http://www.who.int/mediacentre/factsheets/fs282/en/>.

Completely Automated Public Turing Test to tell Computers and Human Apart (CAPTCHA) is widely used as a mechanism to distinguish between human and *bot* access of web resources. The key factor with CAPTCHA is that it should be harder for the automated algorithms to break them and at the same time they should be simple enough for human use. The linear nature of audio CAPTCHA makes them harder to solve in comparison with their visual counter parts [10]. As the persons with visual impairments solely depend on the audio CAPTCHA, enhancing their access becomes an important issue in the context of the universality of the web and the presence of a substantial number of people with such problems. In this paper, a novel approach toward audio CAPTCHA is proposed which is termed as *HuMan* (*human* or *machine?*). The objectives of this research work are as listed below:

- Proposing an accessible audio CAPTCHA for non-visual access with semantic challenges and preemption features;
- Incorporating personalization into the CAPTCHA delivery model by composing the challenges which the user would solve with interest rather than consider it as an encumbrance;
- Proposing a polymorphic challenge–response system which would facilitate the one-to-many relationship between a single media and various challenges;
- Evaluating the acceptance of proposed *HuMan* model with user studies;

The remainder of this paper is organized as follows: Various motivational works done in the field of CAPTCHA and their accessibility are provided in Sect. 2; the *HuMan* model along with its components are explored in Sect. 3; the experimental setup is provided in Sect. 4 and the results of the experiments are analyzed in the same section; the conclusions and future directions for this research work are listed out in Sect. 5.

2 Motivational works

The CAPTCHA functions as a filter for blocking automated access of resources which are earmarked for human-only access [41, 57]. The fundamental working mechanism of this filter is by providing a challenge–response task. The challenge is designed in such a manner that it would be simple for the humans to solve them and rigid enough for preventing algorithms from breaking it. There exists a wide spectrum of efforts to build CAPTCHA by various researchers with the aforementioned characteristic in focus [2, 13, 20, 34, 50].

Based upon the use of media in the CAPTCHA, it can be classified into *text-based*, *audio-based*, *image-based* and

hybrid approaches [38]. Apart from these media-based CAPTCHAs, the adaptability of tactile feedback was also proposed by various studies [27]. However, due to the requirements of specialized sensors in gathering such feedback, they have not yet been widely adopted.

In the *text-based* approach, the challenge is to identify the key in the distorted text [1, 14, 36, 50]. A recent study has explored the application of unicode in providing stronger CAPTCHAs [39]. Some of the text-based challenges such as *ReCAPTCHA* provide an audio interface as well. The advantages and weakness of text-based CAPTCHA are explored by research studies which conclude that 13 out of 15 popular text-based CAPTCHA services are vulnerable to automated attacks [11]. There are studies which have focused on non-English text for the challenge presentation [46].

In the *image-based* approach, the challenge is composed of images and responses would be based on the interactions with these images [16, 18]. The interactions with the images shall include tasks such as identification of a particular type of image or pointing out an image which does not belong to a thematic group [45]. The image-based approach has been extended to include 3D models from random viewpoints [40]. Apart from the normal images, the recognition of human face is also explored in the image-based approach [21, 22, 33].

Both the text and image-based approaches are dependent on the visual perception of the challenge which the persons with visual impairments are unable to address. In the audio-based approach, the challenge primarily depends on the auditory capabilities rather than the visual perception of the user, which is more suitable for non-visual access [23, 26]. There are studies conducted on the accessibility of CAPTCHA not only for the visually impaired but also for people with disabilities of all types, and the measures for improvements have been proposed [34]. There exist studies which utilize the characteristics of human voice which are gathered by asking the user to read out the displayed text [19]. However, such a method would not be optimal for the visually impaired users as they cannot directly read the sentence which appears on the screen.

The accessibility of CAPTCHA by visually impaired users has been addressed with hearing the challenge and saying the response [47]. In the *HearSay* CAPTCHA model, the challenge is that audio would be played and the user has to *say* the answer instead of providing textual input. The perceived success rate of the *HearSay* model is reported as 83%.

The *SoundsRight* CAPTCHA presents a sequence of 10 sounds to the users, and based on the user's identification of the sound with a key press, the challenge–response model is established [28]. This study has reported a 96% success rate in the third round of evaluation of the

CAPTCHA. The effect of adding sound masks on the SoundsRight CAPTCHA is also studied, and the results show that the blind participants were capable of solving these audio challenges better than the sighted users [35]. An interesting and pioneering study in the field of CAPTCHA, HIPUU (*Human Interaction Proof, Universally Usable*), has presented multimodal representation of same tasks through image and audio channels [43]. The CAPTCHA model proposed in the aforementioned study has facilities to solve the CAPTCHA through either menu-based or free-form keyboard-based inputs.

In another interesting recent study, a CAPTCHA based on the jumbled words, termed as *jCAPTCHA*, has been tested with screen reader users with encouraging results in terms of usability and resistance to automatic CAPTCHA resolving [17]. Losing their interest to solve a CAPTCHA by users has been identified as one of the major problems in the related studies. To address this problem, a study has proposed gamification of CAPTCHA with the help of movie scenes [24]. The results of the study conclude that with gamification, the users feel more comfortable to resolve CAPTCHA challenges.

Other issues identified with audio-based approaches for non-visual access are linear playback of the audio challenge and the interference of *screen reader* tools along with the presented audio challenge [7]. The provision of finer control in the interface of the CAPTCHA challenge has shown that 68.5% users were capable of clearing the challenge in the first attempt itself.

Large-scale analytical studies have been carried out on the effectiveness of solving the CAPTCHA by real users [10]. It has been reported that audio CAPTCHAs are more difficult than their visual counterparts, with only 31% perfect agreement among three different solvers of the audio challenge. Though this proves the fact that audio challenge is harder, there is another interesting finding of this study which states that audio CAPTCHA constitutes a non-negligible percentage of access which establishes the point that not only the visually impaired uses the audio challenge but also a fair-sized portion of sighted users has chosen the audio CAPTCHA. This study has also reported that the major portion of time is consumed for listening to the audio challenge. The proposed *HuMan* model incorporates preemption features in order to handle this drawback. These facts emphasize the importance of conducting more works on the audio CAPTCHAs and making them more accessible.

As the individual preferences of users are varying in nature, their interactions would also be diverse. Various types of users might prefer different types of CAPTCHA challenges. There are studies conducted on CAPTCHA personalization based on the cognitive factors of the users [4, 5]. This study has focused on the utilization of factors

such as processing speed, working memory capacity to personalize the CAPTCHA. It has been observed that presentation of the text-based CAPTCHA with personalization enhances the solving efficiency of the user. Geo-CAPTCHA has attempted to incorporate personalization based on the geographic concept to defend against, in the CAPTCHA interfaces [52].

The incorporation of semantics in solving a CAPTCHA would bring them closer to the human and make them complex for the machines to solve. There are studies based on the semantic aspects which present a challenge requiring semantic abilities such as linguistic skills for solving the CAPTCHA [30, 56].

A Comparison of the following seven interesting CAPTCHA studies, HIPUU (2.0 & 3.0) [43], *jCAPTCHA* [17], HearSay [47], accessibility study of ReCAPTCHA [42], HIPUU 1.0 [23], SoundsRight [28], SoundsRight with sound masking [35] for persons with visual impairments are presented in Table 1. The methods are compared using ten parameters. Each of the aforementioned studies has made noteworthy contributions toward making the CAPTCHA accessible for persons with visual impairments.

The *recognition type* refers to the class of recognition to be employed by the user when solving a CAPTCHA. Various recognition types are listed below with their description:

- CSR—Common sound recognition
- LBR—Language-based recognition
- WR—Word recognition
- DR—Digit recognition
- RTRA—Real-time response to audio

The proposed *HuMan* model adopts semantics-based recognition which utilizes common sense world knowledge-based comprehension abilities of the user.

The *response matching* parameter is used to indicate the degree of errors allowed in the answer provided by the user. If the value of this parameter is *exact*, then zero tolerance is employed in matching the user's response with the actual answer. In the case of *fuzzy*, the user shall provide the answer with an allowed degree of mismatch with the actual answer. This fuzzy type comparison is better suited for many real-time scenarios, and hence, the proposed *HuMan* model incorporates fuzzy response matching.

The *noise type* parameter indicates the nature of noise mixed with the challenge audio. As illustrated in Table 1, the HIPUU and SoundsRight approaches did not include any noise. The effect of multiple types of noise (orchestra, laughing, etc.) is studied in SoundsRight with sound masking study [35]. Constant hiss (CH) noise is also utilized in audio CAPTCHAs. The grammatical noise was utilized in *jCAPTCHA*. With the HearSay model, speech-

Table 1 CAPTCHA for persons with visual impairments—features

Study title/feature	HIPUU	jCAPTCHA	HearSay	ReCAPTCHA (A11Y)	HIPUU 1.0	SoundsRight	SoundsRight (SM)
Recognition type	CSR	LBR	WR	DR	CSR	RTRA	RTRA
Response matching	Fuzzy	Fuzzy	Exact	Exact	NR	Exact	Exact
Noise type	None	Grammatical	Speech	CH	None	None	Multiple
Entry method	FFT + DL	FFT	Speech	Numeric	FFT	TSKP	TSKP
User count	74	272	48	6	10	20	40
Repo. building	Manual	Manual	Automatic	Automatic	Manual	Manual	Manual
Sighted user inclusion	Yes	Yes	Yes	No	Yes	No	Yes
Success rate	≥ 90%	80–90%	83%	46%	93%	96%	83.75%
Preemption	No	No	No	No	No	No	No
Personalization	No	No	No	No	No	No	No
Polymorphic	No	No	No	No	No	No	No

based noise is added. The proposed *HuMan* model incorporates *ambient noise* which refers to the natural ambience-based noise present in the environment in which the challenge audio is recorded. The environments are chosen in such a manner that noise is well mixed with the actual audio. For example, the real-time recordings of announcements made in the railway stations include the ambience noise generated by passengers, passing trains and vendors.

The *entry method* parameter refers to the mode of response entry by the user. The response shall be entered in free-form text (FFT) mode or drop-down list. The HearSay approach adopts speech-based response entry. Time-specific key press (TSKP) is another entry method which requires the users to press specific keys in response to the contents of challenge audio. The TSKP method is adopted by the SoundsRight CAPTCHA model. The proposed *HuMan* model utilizes the FFT mode of entry as it is more suitable for the nature of challenges presented to the user.

The *user count* refers to the number of users participated in the experimental setup for the corresponding CAPTCHA model. It shall be observed that the studies involving persons with disabilities generally employ fewer participants compared with other typical user-based experiments. The experiments on proposed *HuMan* model were conducted with 140 participants (86 persons with visual impairments and 54 sighted persons). The *sighted user inclusion* indicates whether experiments were conducted only with visually impaired or a mixture of sighted users and persons with visual impairments.

The *repository building method* indicates whether the challenges are generated automatically or manually. Table 1 shows that five out seven methods fall under the *manual* category. Though it would be desirable to build the challenge repository using automatic methods, the design considerations of the accessible audio CAPTCHA models

require manual processing in building the challenge repository. The challenges for proposed *HuMan* model are also built with the manual process.

Preemption indicates the ability to stop the audio as soon as the user finds out the answer. *Personalization* allows the user to solve the challenges which might interest him/her. The proposed *HuMan* model incorporates both these novel dimensions of preemption and personalization in presenting and solving the CAPTCHA.

This paper proposes a personalized model for accessible CAPTCHA based on user's preferences. The proposed *HuMan* incorporates a semantic challenge–response model which fits into the comfort zone for the humans and complex zone for the bots.

3 The HuMan model

This paper presents a model entitled *HuMan* (*human or machine?*) which aims at enhancing the accessibility of CAPTCHA for persons with visual impairments. The *HuMan* model exploits the convenience of identifying semantic components by a human without much effort. The architecture of proposed model is illustrated in Fig. 1. The formal algorithmic representation of the model is given in Algorithm 1 (given in Appendix I).

The *HuMan* CAPTCHA model consists of three layers namely *HuMan: preference*, *HuMan: builder* and *HuMan: interfacier* as shown in (1) where ρ represents the preference, β represents the builder and α represents the interfacier.

$$H = \{\rho, \beta, \alpha\} \quad (1)$$

The proposed model incorporates the capability to handle spelling errors in the answers typed by the user by adopting a fuzzy comparison with the help of Jaro–Winkler

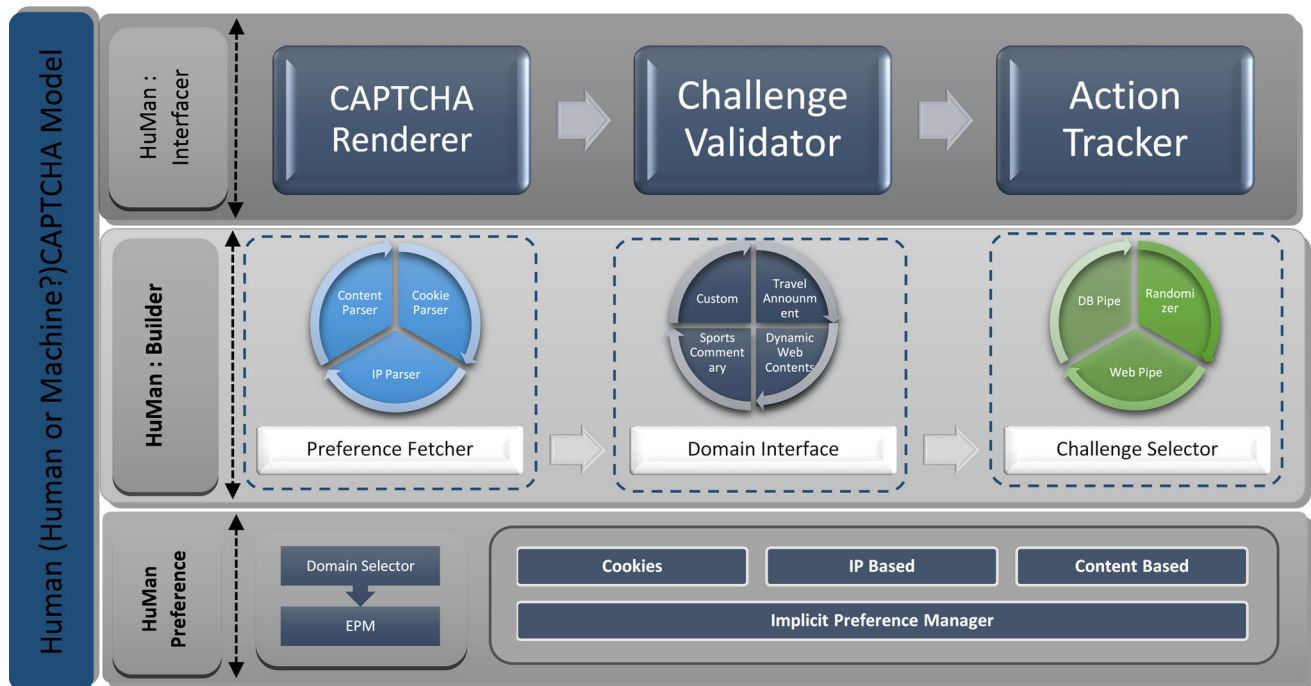


Fig. 1 HuMan model block diagram

edit distance. This feature makes the model efficient, in identifying human users with a degree of error tolerance in the answer verification.

3.1 Preference layer

The preference layer is responsible for capturing the user’s preferences which function as the source for incorporating personalization in the HuMan model. The preference component has two major building blocks: (a) explicit preference manager (EPM) and (b) implicit preference manager (IPM) as shown in (2) where δ, ϵ represent EPM and IPM, respectively, and \oplus denotes the combination operation.

$$H = \{\rho[\delta \oplus \epsilon], \beta, \alpha\} \tag{2}$$

When the user is interacting with the HuMan CAPTCHA model for the first time, the explicit preference managers role is to receive the users interest choice explicitly through the options provided in the interface. These options would be later harnessed by the HuMan model through the implicit preference manager for providing domain-specific CAPTCHA. The implicit preference manager handles the user’s preference using three different parameters as shown in (3).

$$H = \left\{ \rho \left[\delta \oplus \begin{bmatrix} \epsilon_c \\ \epsilon_i \\ \epsilon_t \end{bmatrix} \right], \beta, \alpha \right\} \tag{3}$$

1. For repeating users, the cookies set through the HuMan model in the earlier accesses shall function as the preference source (ϵ_c). When the user revisits a page, preferences need not be selected each time explicitly by the user. The cookies set shall be used to auto-set the preferences. Let us assume user “A” visits a ticket reservation site which has implemented HuMan CAPTCHA. User A selects the preferences as *Sports*. When this page is visited again from the same device, the site would automatically render HuMan CAPTCHA belonging to the *Sports* category, which is identified with the help of cookies. This arrangement is made to make the interaction smoother by automatically selecting the preference which the user opted in the last visit. Nevertheless, users are given the option to change the choice, as per their wish. Strictly speaking, this is not user identification, rather revisit identification from a particular device. Here the revisit is identified with the cookies from the machine. The assumption made is that the user is utilizing his personal device for accessing the web page. If more than one user is utilizing the same device then the last selected preference from that machine, if any, would be chosen. However, CAPTCHAs are provided only in those sites which handle sensitive information. It is always better not to access these sites from shared devices.
2. The client machine’s Internet Protocol (IP) address shall also be used as a parameter for identifying user’s

preferred domain for CAPTCHA (ϵ_i). Both the cookies and IP addresses can be used only for the repeating users. The cookies and IP-based options would be activated if and only if the user permits to do so. Otherwise, the user shall simply select the preferences explicitly on each occasion.

- Based on the contents of the page in which the CAPTCHA is placed, the preferred domain shall be chosen (ϵ_i). For example, the CAPTCHA rendered in a sport web site shall render a CAPTCHA challenge which is based on the sports domain. If the CAPTCHA is placed in an empty page then the domain shall be chosen based on the title of the page, keywords if any specified through *meta*-tag. For extracting keywords from the source web page, a Python-based implementation of *automatic keyword extraction from individual documents* [37] was adopted. The textual representation of the web page was fed as input to the keyword extractor to fetch the relevant keywords. This functionality of embedding CAPTCHA related to the content of the page incorporates the context sensitiveness in the HuMan model. Moreover, CAPTCHA matching the contents of the site would provide a *thematic appeal* to the user which shall be treated as an additional benefit of using HuMan CAPTCHA challenge.

3.2 Builder layer

The next layer in the proposed model is *HuMan: builder* which shall be treated as the pivot element responsible for building the CAPTCHA. The builder layer has three major components as shown in (4).

$$H = \left\{ \rho \left[\delta \oplus \begin{array}{c} \epsilon_c \\ \epsilon_i \\ \epsilon_t \end{array} \right], \beta [\mu \quad v \quad \pi], \alpha \right\} \tag{4}$$

The preference fetcher component $\beta[\mu]$ interfaces with the earlier layer and gathers the preferences. In parallel with the three approaches provided in the implicit preference manager, the preference fetcher also has three respective parsers as shown in (5).

$$H = \left\{ \rho \left[\delta \oplus \begin{array}{c} \epsilon_c \\ \epsilon_i \\ \epsilon_t \end{array} \right], \beta \left[\begin{array}{c} \mu_c \\ \mu_i \\ \mu_t \end{array} \right] \quad v \quad \pi, \alpha \right\} \tag{5}$$

The IP parser is for handling the IP-based preference identification. The content parser is responsible for analyzing the contents to choose the matching CAPTCHA domain. The cookie parser receives the cookies through

their counterpart in the implicit preference manager and chooses the corresponding CAPTCHA domain.

3.3 Domain interfaces

The HuMan model proposes a domain-based approach in providing the CAPTCHA. Three basic domain interfaces are introduced in the current version. The model is designed in such a manner that custom domains shall also be added by the web interface administrators, as shown in (6).

$$H = \left\{ \rho \left[\delta \oplus \begin{array}{c} \epsilon_c \\ \epsilon_i \\ \epsilon_t \end{array} \right], \beta \left[\begin{array}{c} \mu_c \\ \mu_i \\ \mu_t \end{array} \right] \quad \begin{array}{cc} v_s & v_t \\ v_w & v_c \end{array} \quad \pi, \alpha \right\} \tag{6}$$

Most of the CAPTCHA models include noise as they function as a barrier (may not be 100% fail safe) against automatic resolving by machines. At the same time, the presence of noise makes it inconvenient to solve the CAPTCHA for human users also. The challenge in developing CAPTCHA model is to find the right trade-off between the protection and usability with respect to noise. The proposed HuMan model has ambient noise in the CAPTCHA challenge audio. When compared with random algorithm generated noise, the ambient noise would be comparatively less difficult for the users, which is validated with the results of system usability survey (SUS).

3.3.1 Sports commentary

The audio commentary of the sporting events serves as the CAPTCHA challenge in this domain. Short commentary audio which might vary in length from 10 to 35 s is rendered to the user. Before rendering the audio, a question is read out to the user. The questions may range from identifying the sport to identifying a specific event happening in that sport, using the provided audio. The two major reasons for selecting the sports commentary as a CAPTCHA medium are the presence of ambient noise in the commentary audio and the possibility of raising many semantic questions. The stadium crowd noise functions as the inseparable noise in the rendered audio for automated algorithms, whereas the human can segregate the noise from the content with less difficulty in comparison with algorithm generated random noise. The answers for the questions are near impossible for the automated bots to identify, whereas a human can answer them without much effort. For example, in a cricket commentary audio, if the question is *identify the mode of wicket*, the answer shall be *bold, catch, lbw*, etc. These types of questions would be obvious for the user interested in that domain, to answer.

3.3.2 Travel announcements

The audio clips containing announcements made at railway and bus stations are used as the CAPTCHA medium in the *travel announcement* domain. The nature of questions shall be to identify the destination station, train number, etc. These semantic challenges would not pose much effort for the human, whereas for the automated bots it would be a very complex one. In both of the above domains, for a single audio medium, the various numbers of questions shall be associated. Hence, the same audio shall be used for multiple challenges which make the CAPTCHA Polymorphic. The answer to the CAPTCHA would depend not only on the rendered audio but also on the associated question. This one-to-many relationship between a single audio and multiple questions facilitates the *kaleidoscopic* behavior for the challenge. The automated bots cannot associate the answer for the CAPTCHA only with the rendered audio as there exist multiple questions.

3.3.3 Dynamic web contents

Based on the interest identified by the user, the contents of the web pages related to the interest of users functions as the CAPTCHA in this domain. From the DOM (Document Object Model) tree, a random page element with more than three words is chosen. This word bag would be spoken out to the user as an audio clip. To make them secure against automatic speech recognition tools, random phoneme sequences were added in between words. This approach has been established as an important mechanism in making audio CAPTCHA stronger [32]. The user has to type in the first characters of each legitimate (leaving out the extra phonemes added) word from the spoken word bag. The minimum threshold for the size of the word bag is set as four. The HuMan model allows the web interface administrators to customize this threshold value. The most important advantage of this domain is that infinite number challenges shall be composed with this approach, as the source web pages chosen are dynamic in nature with respect to their contents. For example, newspaper web sites shall function as an excellent source for this type of challenge as their contents would be getting updated in fine-grained intervals of time.

The customized domains (v_c) shall also be added to the HuMan model which facilitates the extensions of the proposed idea. For example, domains such as music with identification-based questions shall function as a good candidate for the choice of domain. The core idea of the HuMan model is providing a challenge which would align with the user’s interest and incorporate semantic questions in the challenge.

The third component of the builder layer is *challenge selector* (π) as shown in (7).

$$H = \left\{ \rho \left[\delta \oplus \begin{matrix} \varepsilon_c \\ \varepsilon_i \\ \varepsilon_t \end{matrix} \right], \beta \left[\begin{matrix} \mu_c \\ \mu_i \\ \mu_t \end{matrix} \right] \left| \begin{matrix} v_s & v_t \\ v_w & v_c \end{matrix} \right| \left[\begin{matrix} \pi_r \\ \pi_w \\ \pi_d \end{matrix} \right], \alpha \right\} \tag{7}$$

The role of this component is to select or build the challenge audio from the repository. The challenge selector component consists of web pipe (π_w), DB pipe (π_d) and randomizer (π_r). The web pipe is for interfacing with the web sources in case the selected domain is dynamic web contents. The DB pipe is for interfacing with the database holding the audio clips and their multiple associated questions. The randomizer is responsible for selecting both the challenge and its associated question in a random manner through either web pipe or DB pipe. Randomization is one of the important security aspects of CAPTCHA. The presence of two layers of randomization, one for selecting the audio challenge and another for selecting the associated question, makes it stronger.

3.4 Interfacer layer

The next layer in the HuMan model is the *interfacer* (α) which is responsible for rendering, validating and tracking activities, as shown in (8).

$$H = \left\{ \begin{matrix} \rho \left[\delta \oplus \begin{matrix} \varepsilon_c \\ \varepsilon_i \\ \varepsilon_t \end{matrix} \right], \\ \beta \left[\begin{matrix} \mu_c \\ \mu_i \\ \mu_t \end{matrix} \right] \left| \begin{matrix} v_s & v_t \\ v_w & v_c \end{matrix} \right| \left[\begin{matrix} \pi_r \\ \pi_w \\ \pi_d \end{matrix} \right], \\ \alpha \left[\begin{matrix} \alpha_r \\ \alpha_v \\ \alpha_t \end{matrix} \right] \end{matrix} \right\} \tag{8}$$

The CAPTCHA renderer (α_r) facilitates the rendering of the challenge in the web interface. The renderer announces the question before the audio clip is played. The ordering of question preceding the audio makes the user to focus on the corresponding semantic components of the audio, to provide the answer. The challenge validator (α_v) is for checking whether an answer provided by the user is correct or not. In the case of a correct answer, further access to the web interface is provided and in the case of a wrong answer, another HuMan challenge is rendered via the interface. The validator shall be customized to check the correctness of the answer with an allowed level of distortions in the answer. For example, the edit distance shall be

used as a parameter for measuring the actual answer and the user provided response [31]. A threshold value shall be set for this edit distance in computing correctness of the answer. The measure adopted in the HuMan model for fuzzy string matching is Jaro–Winkler distance [54]. The reason for choosing Jaro–Winkler is its appropriateness for comparing smaller strings which are the case with CAPTCHA answer comparison.

The action tracker (α_t) component is used to gather information regarding the user's interactivity with the HuMan interface. The collected data shall be used further for enriching the models performance. The tracker shall collect information such as whether the user plays the audio completely or stops it before it reaches the final point. The HuMan model has preemption capabilities which allow the user to stop the audio as soon as the answer is identified. The tracker is also employed to collect details regarding number of times a particular CAPTCHA challenge is failing. With these data, the following scenarios are handled:

- If the failure rate of a particular challenge is critically higher, then either that challenge shall be totally removed or modified accordingly;
- If for a specific question in a CAPTCHA challenge, incorrect answers are provided most of the time, then such questions were modified keeping the CAPTCHA challenge audio intact;
- Another scenario is to update the answer itself. For example, if for a specific question most of the users are providing the same answer, then the actual answer itself was modified [43]. This step was accommodated so that the core objective of HuMan CAPTCHA, i.e., differentiating human and machines is satisfied.

AES 128-bit encryption was applied to questions and answers before they were stored in the database to prevent unauthorized leakage of these data. If these details were hacked by someone by attacking the database, then the encrypted form of questions and answers would leave it unusable. With respect to the security of AES 128 bit encryption, it is reported by security studies that the best possible attack on AES-128 requires 2^{88} bits of data storage (≈ 38 trillion terabytes of data)². Due to the impracticality of such a mammoth storage requirement, it can be treated as an acceptable mechanism to protect the HuMan CAPTCHA challenge.

The design goals for HuMan are accessibility, semantic challenge, CAPTCHA preemption and personalization of the CAPTCHA challenge. Through the aforementioned components, all these four design goals are achieved. Moreover, the model is designed in such a manner that it is

flexible enough to incorporate the future requirements such as custom domains, localization by providing the CAPTCHA challenge in the user preferred language.

4 The experiments and results analysis

This section explores the design and analysis of experiments carried out with HuMan. For experimentation purpose, a prototype implementation of HuMan was developed using PHP as server side scripting, JavaScript in the client side, MySql for database storage and Apache as the web server. For tasks such as keyword recognition, Python 2.7 was also used. With respect to the hardware Quad Core processor systems with 4 GB main memory and 128 Mbps leased line Internet connectivity were used. For non-visual access, the screen reader NVDA (NonVisual Desktop Access) was utilized in the client machine³. The three major reasons for choosing the NVDA are the ease of use, free access and availability of trained users in and around our campus.

Experiments on the proposed HuMan model were carried out with 140 participants which included both persons with visual impairments and sighted users. The demographic details of the participants are illustrated in Table 2. The *YoE* refers to *years of experience* in the table.

Three different domain interfaces were incorporated in the current implementation of HuMan. They are sports, travel announcements and dynamic web contents. For the sports audio commentary, the clips from the cricket matches were utilized. The presence of stadium noise in these clips made them a suitable option for purpose of CAPTCHA challenge. For travel announcements, the recorded clippings from the railway station announcements were utilized as CAPTCHA medium. The presence of noise due to crowd, passing vehicles and vendors in these announcements made them suitable for CAPTCHA challenge. The feature set of HuMan CAPTCHA base is shown in Table 3.

The HuMan CAPTCHA model has built polymorphism into the CAPTCHA challenge. The term polymorphism is adopted to represent the ability of using the same CAPTCHA medium with more than one challenge. For a single audio, there would be more than one associated question. The randomizer component would select the candidate question to be announced to the user from the list of available questions to that challenge.

4.1 Mean polymorphic index

One of the unique features of proposed HuMan model is the ability to establish 1:*N* relationship between the

² <https://blog.agilebits.com/2011/08/18/aes-encryption-isnt-cracked/>.

³ <http://www.nvaccess.org/>.

Table 2 Participants demographic details

Participants details	Persons with visual impairments	Sighted users
Number of participants	84	56
Male	48	30
Female	36	26
Mean age	31.03	26.6
Standard deviation	8.17	8.6
Low-vision users	40	NA
Blind users	44	NA
Average screen reader YoE	6.83	
Average computer YoE		7.86

Table 3 HuMan CAPTCHA audio features

Feature	Value
Challenge min length	8 s
Challenge max length	36 s
Average duration	15.7 s
Mean polymorphic index	3.48
Character set	Natural language
Voice type	Human: male/female
Noise type	Ambient
Repeat	Manual
Bit rate	256 kbps

challenge audio and questions. The traditional CAPTCHA models adopt 1:1 relationship between the challenge audio and the answer. HuMan introduces polymorphism in the challenges. The term polymorphic refers to the ability to associate various answers with respect to a single audio challenge. The answer would be dependent on both the challenge audio and the current question posed to the user. In order to measure the polymorphic ability, this study proposes a metric termed mean polymorphic index (MPI) which is computed as the mean of number of challenges associated with a HuMan CAPTCHA, as shown in (9).

$$MPI = \frac{\sum_{i=1}^n |oi|}{n} \tag{9}$$

In (9), $|oi|$ indicates the number of question choices for the challenge i and n represents the total number of audio challenges. The possible range of values for mean polymorphic index is from 1 to infinity. The value of MPI cannot be less than one as there should be at least one challenge for any audio. MPI measures the degree of polymorphism for a HuMan CAPTCHA implementation. The higher the value of MPI, the better would be the strength of the system.

The polymorphic nature of HuMan functions as an additional layer of resistance against attacks. In traditional

CAPTCHA model, the challenge functions as an independent entity which is sufficient to find out the answer. Solving HuMan CAPTCHA requires both the challenge and current question being posed. The MPI functions as a factor to increase the number of different combinations of challenges that can be posed. For example, in conventional models if there are 1000 audio files are possible in a system then the total number of CAPTCHA challenges is also 1000. In the case of HuMan, the possible number of CAPTCHA challenges is determined by both number of challenge audio and the total number of questions (oi). For example, in a CAPTCHA system with 1000 audio files and a MPI of 8, total number of challenges that can be generated would be ≈ 8000 . As there is no upper limit is set for MPI, the permutation of challenges shall be made very large. This multifold increment in the count of possible number of challenges makes the proposed HuMan CAPTCHA comparatively stronger. For a automatic bot to break the HuMan model, it has to capture both the audio repository and question, and answer mapping database. Another layer of defense introduced here is the AES-128 bit encryption of questions and answers.

4.2 Domain interface sample challenges

With respect to the sports domain, the transcript of a sample challenge and its four associated challenges are shown in Table 4 (this transcript is from the television broadcast of the cricket world cup 2015 match between India and Australia).

The spectrogram of the audio clip utilized for the HuMan challenge explained in Table 4 is shown in Fig. 2.

Similarly, another sample from the travel announcements domain with its associated five challenges are shown in Table 5. The spectrogram of the audio clip is illustrated in Fig. 3.

4.3 The HuMan model prototype

A prototype implementation of the HuMan model as shown in Fig. 4 was developed to carry out the experiments and analysis.

Table 4 Sample HuMan challenge with sports domain interface

Item	Description
HuMan audio	This could be the wicket that India wanted. It has come late nonetheless. It is a huge wicket. Smith has gone for 105 from 93
Challenge #1	Identify the bowling team from this clip?
Challenge #2	What did the team India expected?
Challenge #3	What was total number of runs the batsman scored?
Challenge #4	What was the number of balls the batsman faced?

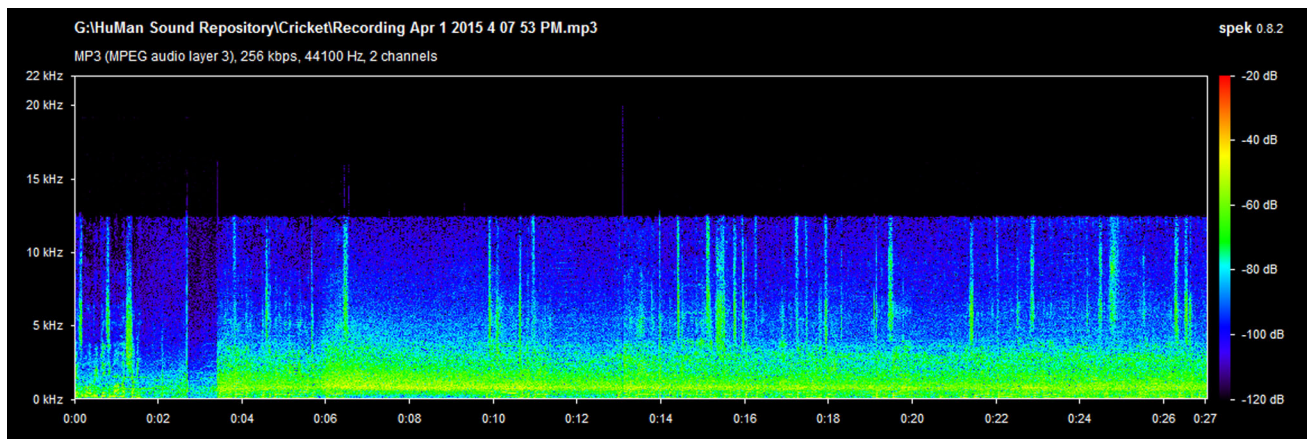


Fig. 2 Spectrogram of sample HuMan challenge with sports domain interface

Table 5 Sample HuMan challenge—travel announcements domain interface

Item	Description
HuMan audio	May I have your attention please? 2415 Indore–Hazrat Nizamuddin express from Indore via Ujjain to Nizamuddin is expected to departure 16 h 20 min from Platform number 3
Challenge #1	What is the train number mentioned in the audio?
Challenge #2	The train is leaving from which platform?
Challenge #3	What is the source station for the train?
Challenge #4	What is the type of train mentioned in the audio?
Challenge #5	When is the train departing from the station?

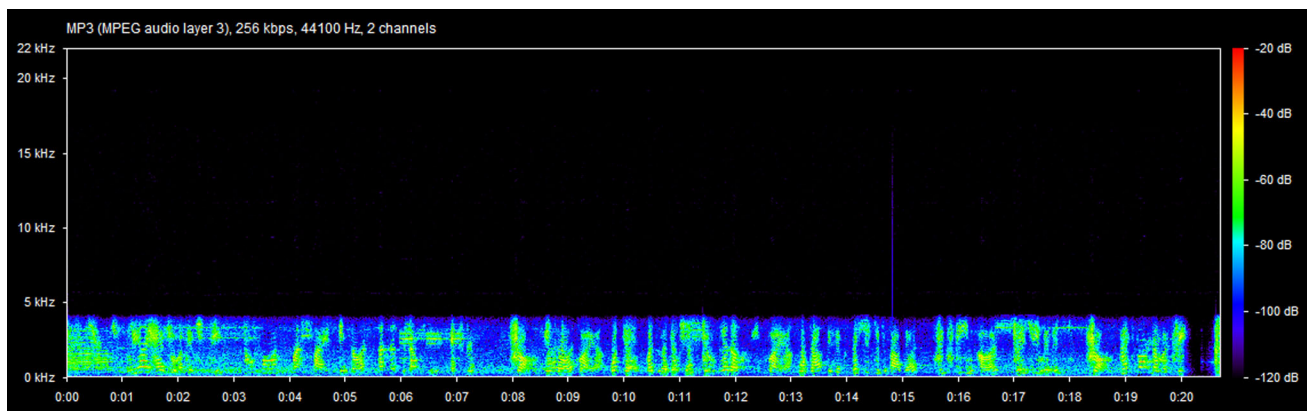


Fig. 3 Spectrogram of sample HuMan challenge with travel announcements domain interface

The sample screenshot shows the inclusion of HuMan CAPTCHA block in a demo sign-up form. For the visual indications given for the purpose of validation, the corresponding audio alerts were also provided for non-visual access.

4.4 The procedure

The experiments involving 140 participants were set up as fourteen sessions. Each session involved ten participants.

Out of them six were persons with visual impairments and four were sighted users. It was made sure that in each session there would be both low-vision and blind users. Each session began with a demonstration of the proposed HuMan model using the prototype implementation. Exact set of instructions were given across all the fourteen sessions. During the experimental sessions, no additional clarifications were encouraged in order to maintain consistency across all the fourteen sessions. In each session, twelve different HuMan CAPTCHA challenges were

Fig. 4 HuMan model prototype implementation

presented to the participants. These twelve challenges are selected in such a manner that it consisted of equal number of personalized and non-personalized challenges. The presentation of personalized and non-personalized challenges was made in a random manner so as to avoid any effect due to sequential presentation. In the case of personalized challenges, three were explicit and three were implicit challenges.

The quantitative metadata of experiments conducted on the proposed HuMan model are listed as follows:

- In each session, 10 users participated. Out of these, 6 were persons with visual impairments and 4 were sighted users. Among the six persons with visual impairments, a mix of low vision and blind was maintained proportionately based on the availability;
- Each user had to solve 12 CAPTCHA challenges presented to them. So in each session $10 \times 12 = 120$ HuMan challenges were solved;
- The total number of sessions was 14, which makes the overall number of HuMan challenges solved in the experiments to $120 \times 14 = 1680$.

To maintain uniformity across the sessions, the reading speed of screen readers was set at a constant level. This decision was to make sure that the tasks completion times are not influenced due to the reading speed. The participants were allowed to interact with the system for 5 min at the beginning of the session to make them feel comfortable with the screen reader's speed. At the end of each session, an exit-experiment questionnaire was given to the participants and the feedback was collected. The exit-experiment questionnaire involved two major sections: (a) The HuMan CAPTCHA-specific, six different measures proposed in Sect. 4.12 were collected in Part A and (b) to measure the validity of the proposed model with respect to user satisfaction, the standard SUS (System Usability Study) survey was carried out [8].

4.5 Metrics

The metrics adopted are mean solving time (MST) and mean success rate (MSR) which deal with the time required to solve the CAPTCHA and the percentage of time success was achieved, respectively. The reasons for adopting these

Table 6 Mean solving time

Session ID	P-MST-VI	P-MST-SU	MST-VI	MST-SU
1	24.25	26.18	35.95	35.78
2	24.4	25.73	34.13	36.33
3	23.65	26.28	34.43	36.08
4	23.22	24.43	34.73	36.6
5	23.1	24.88	35.63	38.03
6	22.6	24.88	36.58	36.65
7	22.77	24.6	35.32	36.1
8	23.15	25.5	34.67	36.85
9	24.72	26.58	33.65	36.48
10	22.27	24.23	33.83	34.33
11	23.8	24.15	35.72	36.28
12	22.82	25.23	35.5	37.63
13	22.88	25.25	35.42	35.88
14	23.88	24.1	34.73	37.28

Table 7 MST—summary values

Metric	P-MST-VI	P-MST-SU	MST-VI	MST-SU
Mean	23.394	25.144	35.021	36.450
S.DEV	0.735	0.821	0.849	0.893

metrics are their proven efficiency through large-scale studies in CAPTCHA research domain and the ability to capture these metrics without disturbing the normal flow of the user. The values of MST are given in seconds. The personalized MST is indicated as P-MST. The SD

(Standard Deviation) metric is also specified to indicate the intra-session measure for the associated metric.

Table 6 presents the MST values for the experiments conducted in fourteen sessions. The summary of MST values is given in Table 7. Overall mean solving time value was observed as 23.39 s for personalized CAPTCHA rendered by HuMan and 35.02 s for non-personalized challenges with respect to persons with visual impairments. For sighted users, corresponding values were observed as 25.14 and 36.45, respectively. The box plot for mean solving time is plotted as shown in Fig. 5. The box plot is generated using an online tool called BoxPlotR [48]. The mean values are marked with + sign in the box plot. The statistical measures of mean solving times are shown in Table 8. It shall be inferred from the box plot that the median values in all sessions are not deviating significantly, which indicates that the solving time is consistent across all sessions. It shall also be observed that the quartile values are also consistent in a range across all ten sessions, which is a preferable behavior.

4.6 Impact of personalization on MST

A comparison was made with non-personalized rendering which generated the CAPTCHA challenge without incorporating the user preferences, as illustrated in Fig. 6. The mean solving time for non-personalized CAPTCHA model was observed as 35.02 s, which indicates 33.02% overall improvement in the solving time with the incorporation of personalization.

The solving time for the HuMan CAPTCHA challenge with personalization and preemption is observed as 23.39 s

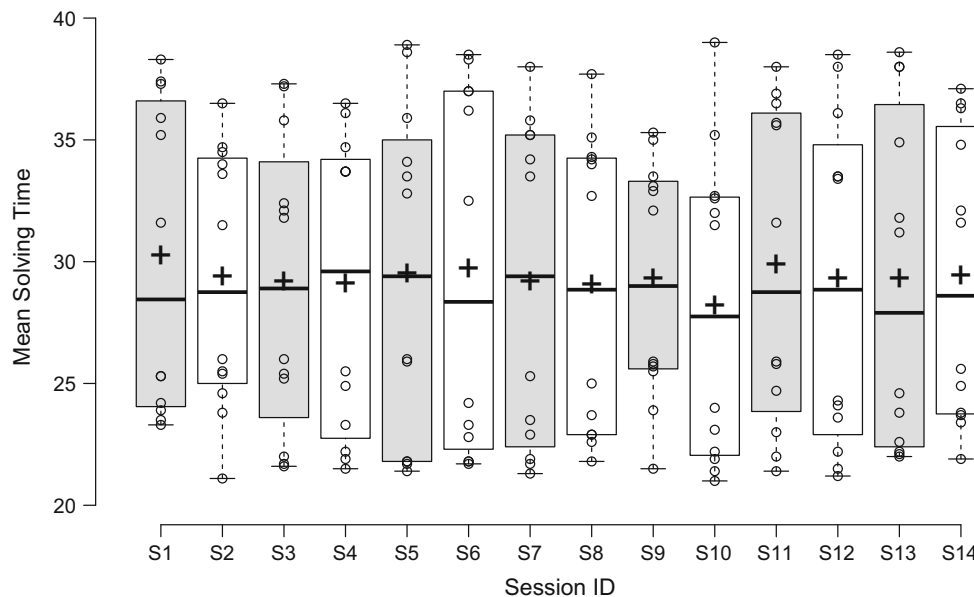


Fig. 5 Mean solving time box plot

Table 8 MST statistical measures

Session ID	Upper whisker	3rd quartile	Median	1st quartile	Lower whisker	Mean
S1	38.3	36.6	28.45	24.05	23.3	30.1
S2	36.5	34.25	28.75	25	21.1	29.27
S3	37.3	34.1	28.9	23.6	21.6	29.04
S4	36.5	34.2	29.6	22.75	21.5	28.98
S5	38.9	35	29.4	21.8	21.4	29.37
S6	38.5	37	28.35	22.3	21.7	29.59
S7	38	35.2	29.4	22.4	21.3	29.04
S8	37.7	34.25	28.85	22.9	21.8	28.91
S9	35.3	33.3	29	25.6	21.5	29.18
S10	39	32.65	27.75	22.05	21	28.05
S11	38	36.1	28.75	23.85	21.4	29.76
S12	38.5	34.8	28.85	22.9	21.2	29.16
S13	38.6	36.45	27.9	22.4	22	29.15
S14	37.1	35.55	28.6	23.75	21.9	29.31

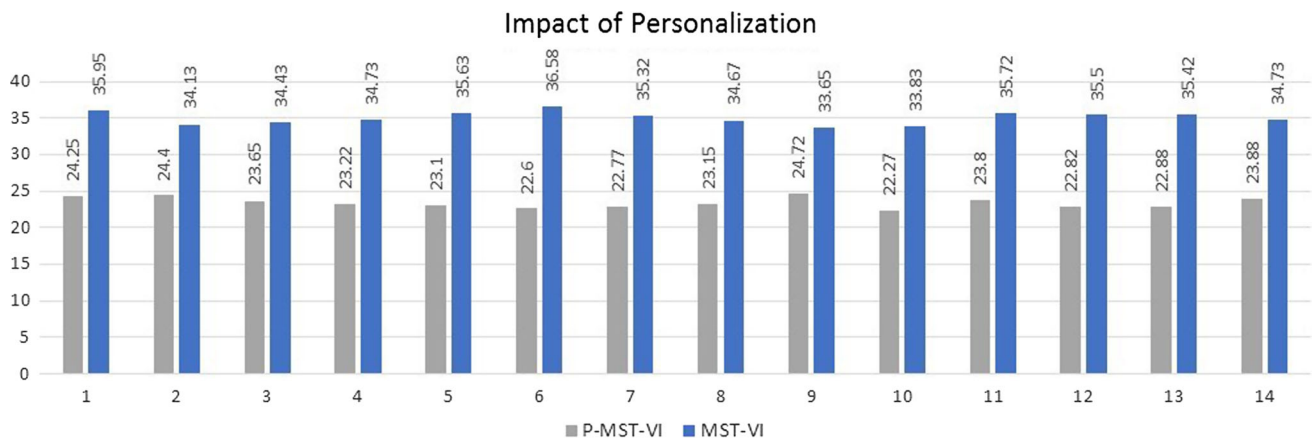


Fig. 6 Impact of personalization in solving CAPTCHA

which is better than the solving time reported for other popular services such as ReCAPTCHA audio (30.1 s), Yahoo audio (25 s) [10].

4.6.1 Wilcoxon signed-rank test

In order to validate the positive impact of personalization on the solving process, *Wilcoxon signed-rank* test was set up [55]. The hypotheses formulated are as follows:

- **Null Hypothesis H0** *The incorporation of personalization has no impact on mean solving time of the CAPTCHA challenge rendered by HuMan model;*
- **Alternate Hypothesis H1** *The incorporation of personalization has a positive impact on mean solving time of CAPTCHA challenge rendered by HuMan.*

The results of the Wilcoxon signed-rank test projected the Z value as -9.778 . The p value is 0. The result is significant at p 0.05. Hence, the null hypothesis is rejected and it

is statistically proved that inclusion of personalization has a positive impact on the mean solving time. This improvement shall be considered as load and increased involvement for the user while solving the personalized CAPTCHA rendered through the HuMan model.

4.7 Mean success rate

The session-wide mean success rate for persons with visual impairments and sighted users is shown in Figs. 7 and 8, respectively. With respect to the persons with visual impairments, it was observed that 91.04% instances were solved at the first attempt itself. For the second attempt, the MSR was observed as 92.19 and for the attempt three it was 94.15. For the sighted users, corresponding values were observed as 91.85, 92.3 and 94.32. It shall be noted that there is no significant differences in MSR among both categories. Though the audio CAPTCHA are generally considered to be tougher to solve, it was observed that the

VI - Mean Success Rate Attempt I, II and III

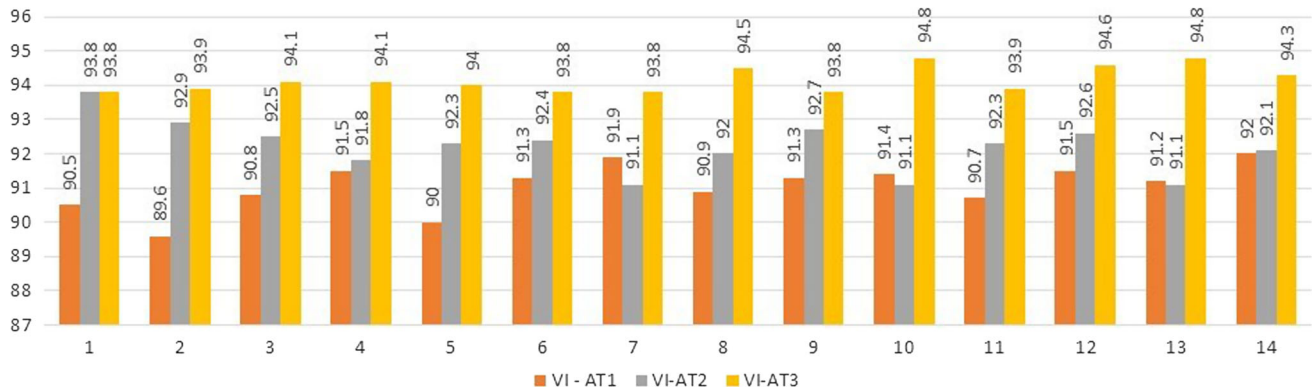


Fig. 7 Persons with visual impairment—mean success rate in Attempts I, II and III

Sighted Users - Mean Success Rate Attempt I, II and III

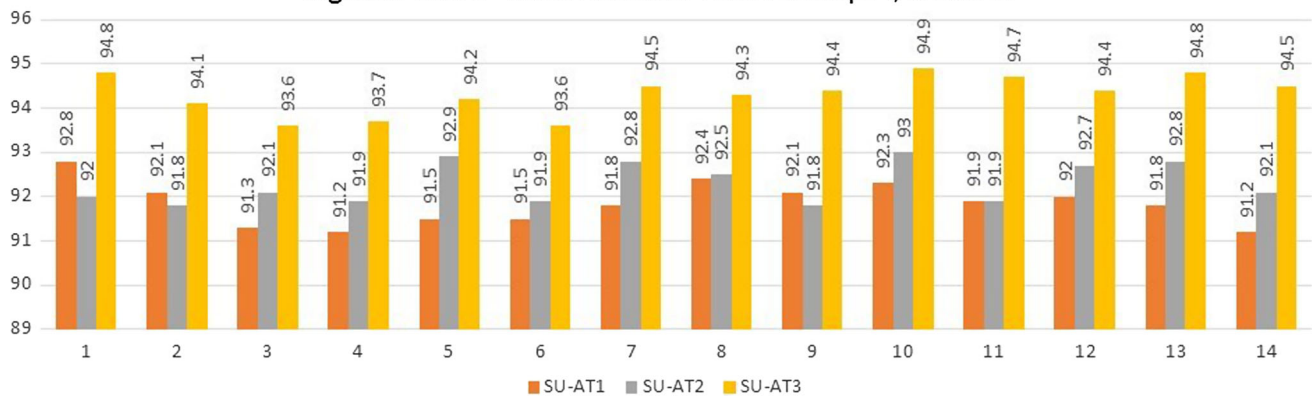


Fig. 8 Sighted users—mean success rate in Attempts I, II and III

Table 9 Impact of personalization on MSR

Metric	Attempt I	Attempt II	Attempt III	Mean
VI-MSR	91.04	92.19	94.15	92.46
VI-MSR (WoP)	85.08	87.07	89.98	87.38
SU-MSR	91.85	92.3	94.32	92.82
SU-MSR (WoP)	87.45	88.78	90.35	88.86

inclusion of semantic challenge and personalization has a positive impact on solving them. The impact of personalization on MSR is presented in Table 9. The MSR values for persons with visual impairments and sighted users are given along with their *without personalization (WoP)* counterparts. It shall be inferred from the table that for both sighted and visually impaired users’ personalization has a positive impact with respect to MSR. The mean of MSR values for all three attempts for persons with visual impairments was 92.46. The respective counterpart without personalization (VI-MSR (WoP)) was 87.38 which

indicates personalization has improved the MSR by 5.08 %. Similarly for sighted users, the improvement was observed as 3.96 %.

A comparative analysis among the three domains, sports commentary, travel announcements and dynamic web contents was carried out in terms of mean solving time and mean success rate. The results are shown in Table 10. The overall mean across all fourteen sessions, for three domains was 23.67, 24.24 and 21.85 s, which indicates no significant differences (Fig. 9). Similarly the success rate values were 91.04, 91.74 and 91.15 for the domains in the same aforementioned order (Fig. 10).

4.7.1 Wilcoxon signed-rank test for MSR and personalization

In order to validate the positive impact of personalization on the mean success rate (MSR), a Wilcoxon signed-rank test was set up [55]. The hypotheses formulated are as follows:

Table 10 Comparison among three domain interfaces

Session ID	SC MST	TA MST	DWC MST	SC MSR	TA MSR	DWC MSR
1	23	24.2	22.3	91.9	89.7	90
2	23.7	23.5	22.2	90.8	86.9	94.6
3	23.4	24.4	21.4	90.9	96	89.2
4	23.9	24.4	21	94.1	95.8	89
5	23.2	23.6	21.2	88.9	92	91.3
6	23.8	23.8	22.1	90.1	94	89.3
7	23.6	24.2	21.5	94.4	92.8	93.9
8	23.5	24.2	22.2	93.5	89.2	89.6
9	24.3	24.9	22.1	89.9	87.1	94.5
10	24.3	24.9	21.3	87.5	96	89.8
11	23.2	24.5	21.9	91.7	86.7	93.8
12	23.1	23.8	22.2	90.1	91.1	89.2
13	24.5	24.9	21.6	87.3	93.4	89.5
14	23.9	24.1	22.9	93.5	93.7	92.5

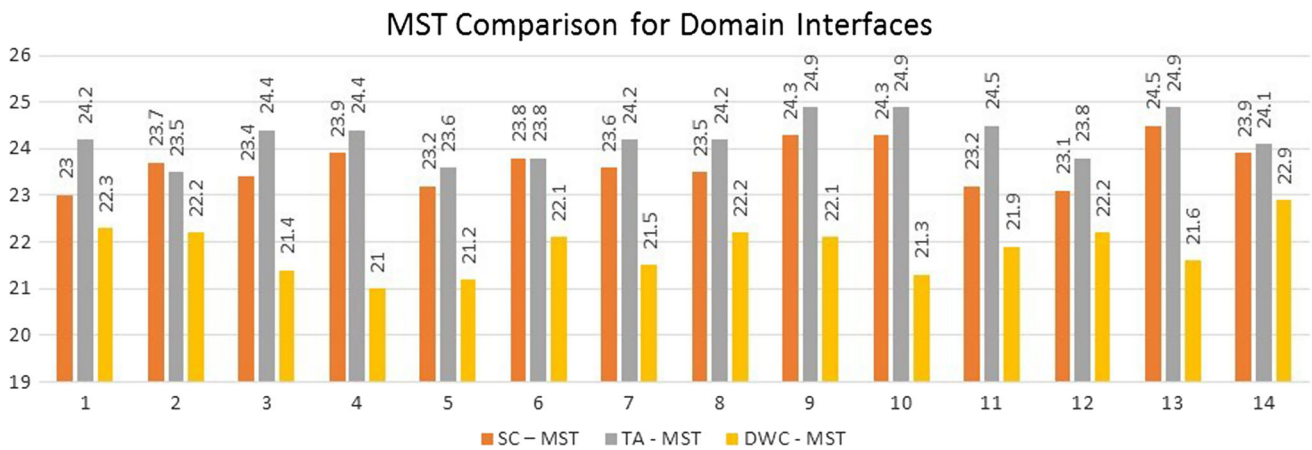


Fig. 9 Comparison of MST across domains

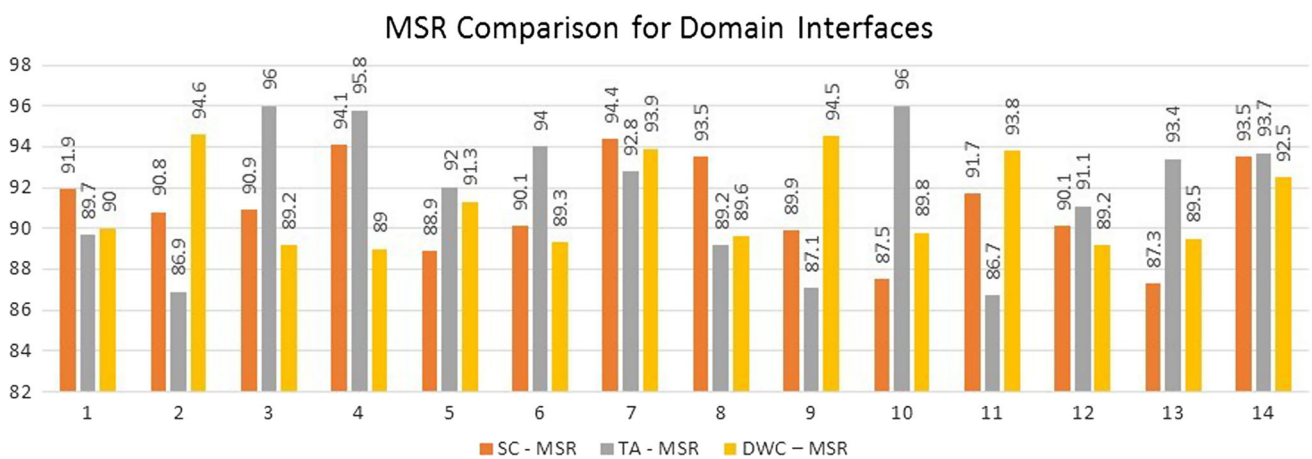


Fig. 10 Comparison of MSR across domains

- **Null Hypothesis H0** *The incorporation of personalization has no impact on mean success rate of CAPTCHA challenge rendered by HuMan model;*
- **Alternate Hypothesis H1** *The incorporation of personalization has a positive impact on mean success rate of CAPTCHA challenge rendered by HuMan.*

The results of the Wilcoxon signed-rank test projected the Z value as -8.658 . The p value is 0. The result is significant at $p < 0.05$. Hence, the null hypothesis is rejected and it is statistically proved that inclusion of personalization has a positive impact on the mean success rate (MSR).

4.8 CAPTCHA preemption index

Another important attribute of the HuMan CAPTCHA model is its ability to preempt it as soon as the user identifies the answer. Unlike most of the audio CAPTCHA models where the user has to listen to the complete audio to answer the challenge, HuMan has the preemption feature which would facilitate the user to solve the CAPTCHA in a quicker manner. The proposed metric *CAPTCHA preemption index* for a session s (CPI_s) is computed as shown in (10) where $\ell(\omega_i)$ indicates the total length of CAPTCHA audio ω_i , $\bar{\ell}(\omega_j)$ is the preemption point and $|Sp|$ indicates the total number of HuMan preempted in that session.

$$CPI_s = \frac{\sum_{j=1}^{|Sp|} (\ell(\omega_j) - \bar{\ell}(\omega_j))}{\sum_{i=1}^{|S|} \ell(\omega_i)} \quad (10)$$

The CAPTCHA preemption index is applicable only to *travel* and *sports* domain CAPTCHAs. For the dynamic web contents domain, user has to enter the first character of each word, and hence, it cannot be preempted. However, the *dynamic web contents* domain was incorporated into the HuMan implementation due to the ability to provide personalized challenges which were built from sources identified by the user. As contents of the web resources are dynamic, CAPTCHA built with *dynamic web contents* domain exhibits improved dynamism.

The mean CAPTCHA preemption index across all sessions (without considering the CAPTCHA belonging to dynamic web contents domain) is 0.514 for persons with visual impairments and 0.520 for sighted users which indicates that more than half of the length of CAPTCHA audio is skipped by both categories of users while solving the challenge.

4.8.1 CAPTCHA preemption impact

In order to validate the CAPTCHA preemption impact on the HuMan model, the *Wilcoxon signed-rank* test was set up by comparing the MST with and without preemption

feature [55]. The hypotheses formulated are as given below:

- **Null Hypothesis H0** The incorporation of preemption has no impact on mean solving time of CAPTCHA challenge rendered by HuMan model;
- **Alternate Hypothesis H1** The incorporation of preemption has positive impact on mean solving time of CAPTCHA challenge rendered by HuMan.

The results of the Wilcoxon signed-rank test computed the Z value as -4.9781 . The p value is 0. The result is significant at $p \leq 0.05$. Hence, the null hypothesis is rejected and it is statistically proved that inclusion of preemption has a significant impact on the mean solving time.

4.9 Jaro–Winkler measure

Jaro–Winkler measure was adopted for the CAPTCHA result verification purpose. The possibility of typographical errors is high as the challenges rendered are semantic and their answers may include entities such as names of persons and places. Hence, an exact matching between the actual answer and the user typed answer would undermine the original objective of differentiating between human and machine. The objective is to simply check whether the user is capable of recognizing the semantic challenge and identifies the answer. Thus it was decided to include fuzziness into the answer validation process, and hence, the Jaro–Winkler measure was used. The Jaro–Winkler threshold value was set as 0.7. During the experiments, a comparison was made between validation strictly based on exact answer and distance measures. It was observed that the inclusion of Jaro–Winkler distance measures in the validation process increased the MSR by 51.34%.

4.10 Validity analysis

This section explores the validity of the proposed HuMan CAPTCHA model. The validity is analyzed in three major dimensions: (a) internal validity, (b) external validity and (c) ecological validity using the standard factors [12].

4.10.1 Internal validity

With respect to the Internal Validity all of the eight standard influencing factors are analyzed as described below

- *History* It has been established that longer duration studies would have great influence on this history parameter. The HuMan model experiments were conducted in short sessions which spanned less than an hour, which functions as a barrier for this influence;

- *Maturation* The risk of participants getting tired or entering into a mechanical mode was avoided by two factors: (a) Many challenges were presented in a domain in which they are interested and (b) the total number of challenges that a participant has to solve was kept at a manageable level (12 CAPTCHA challenges for each user);
- *Testing* The participants were given a demonstration of the system before the experiment session began. Across all sessions, an *exact* set of instructions were delivered to nullify any possible bias. During the experimental session, any detailed clarification to specific participants was avoided;
- *Instrumentation* The measurements were carried out using scripts monitoring the user actions, and hence, no human observation was adopted to measure. This step was taken to avoid observer-related bias. As the scripts for measurement of time, etc. were exactly the same, the instrumentation influence was avoided. The computer systems utilized were also exactly the same across all fourteen sessions. The screen reader reading speed was also kept at a constant level to nullify the instrument-related bias;
- *Statistical regression* Negligibly small numbers of outliers in the experiments were identified and eliminated to handle this factor. For example, two challenges identified with maximum number of wrong answers were not included in the mean score computation;
- *Selection* The selection of participants for the experiments was carried out keeping in mind that equal number of persons with visual impairments and sighted users are involved in each session. With respect to the visually impaired, the low vision and blind were proportionately mixed at all sessions;
- *Experimental mortality* This issue did not arise with the experimental design of HuMan model. The experimental sessions were completed in a shorter time. So the possibility of subjects dropping out of experiments did not arise in the HuMan experiments.
- *Selection interactions* As the participants were selected by following a uniform treatment, this factor was kept minimal.

4.10.2 External validity

Boosting the external validity of results is attempted with session-based experiments. Basically these sessions serve as replication tools. Each session has 10 participants and the mean of the parameters is calculated for individual sessions. The consistency of a session finding is checked by comparing with other sessions. This mechanism of repeating the experiments with different sets of participants

is used as an important factor for external validity of the results of proposed HuMan model. Moreover, the following steps were taken to increase the external validity:

- The participants for each session were selected in a random manner. This randomization reduces the *interaction* between subject selection and the findings;
- Each session has different sets of participants with no overlapping. *Pretesting* was not carried out with any participant to avoid bias due to pretesting;
- *Experimental setting*-related bias was avoided by maintaining consistency across all the sessions. The users were specifically instructed to work at their normal speed. For persons with visual impairments, this factor was controlled by the screen reader reading speed. All participants were informed that their completion time and success rate are measured by automated scripts so as to avoid any bias caused by some participants knowing these details;
- The *multiple treatment intervention* was kept minimal as the complete HuMan CAPTCHA solving is considered as an atomic unit. The randomization in presenting personalized and non-personalized challenges also assisted in controlling this parameter, thereby increasing external validity.

4.10.3 Ecological validity

Measures for ecological validity of the results were also incorporated in the design to the extent possible. The HuMan CAPTCHA challenges were presented to the users in *pseudo-web pages* to mimic real-time environment. Here the pseudo-web pages refer to pages specifically built for the experimental purpose. For example, pages such as online train ticket booking, cricket match information, student information portal were utilized to present the CAPTCHA challenges. Moreover, CAPTCHA solving task is not very complex and involves only two major steps of recognizing the challenge and entering the answer, the environmental factors would be comparatively lesser.

4.11 Security aspects

The primary objective of the proposed HuMan CAPTCHA model is to provide accessible an alternative to the traditional audio CAPTCHA. However, the resistance against attack on the HuMan model also need to be considered carefully.

The major security requirements identified for CAPTCHA by research studies ([44, 58]) are analyzed with respect to the design of the proposed HuMan CAPTCHA model as follows.

- **Media security** One of the primary security requirements identified for CAPTCHA is media security which refers to the obfuscations added to the media before presenting them to the user. The distortions of textual representations, addition of noise to audio the measures that fall under the media security category; the CAPTCHA challenges presented by HuMan are obfuscated with ambient noise in which CAPTCHA audio challenges are recorded. In contrast to the constant, uniform type of noise present in various existing audio CAPTCHA, the ambiance noise of HuMan is neither uniform nor constant. Another characteristic of this ambiance noise is that for human it would be relatively simpler to ignore them, as we face such circumstance in real-life scenarios and the human brain is well trained to do this task effortlessly. It has been already established by research studies that CAPTCHAs containing phrases are better suited for humans than CAPTCHAs containing isolated digits or alphabets. Moreover, these type of CAPTCHA are identified as strong against automatic speech recognition (ASR) tools [49]. As the HuMan model inherits the characteristics of sentence-based approach with ambient noise, it is stronger.
- **Script security** refers to the strength of CAPTCHA against algorithmic breaking. For a traditional audio CAPTCHA, the only major task involved is the recognition of a digit or letter after the removal of noise from the audio challenge. The stages involved in breaking the proposed HuMan CAPTCHA would involve the following steps:
 1. Transcribe the audio into textual format;
 2. Understand the meaning of the question;
 3. Extract concepts from the transcribed text and map them with concepts present in the question;
 4. Derive or identify the answer to the question by analyzing this *concept link map*, with potential inclusion of specially constructed, domain-specific ontology.

Theoretically, if we assume the development of an ASR which shall recognize the audio challenges 100% correctly, then it leaves the remaining three steps of breaking the HuMan CAPTCHA unsolved. It shall be noted that conversion from text to speech is simpler than the reverse. The speaker-independent speech to text recognition requires powerful hardware resources and training process.

A Python script was developed, which utilized the *Sphinx* speech recognition system that is considered to be one of the most frequently adopted systems in similar pioneering studies to break the CAPTCHA [9, 51]. The standard *pocketsphinx* implementation of

Table 11 Challenge text recognition

Domain	WRP
Travel	8.6
Sports	7.3
Dynamic web contents	9.2

Python was adapted in performing the recognition tasks without any specific training. The output of the aforementioned script was compared with the original transcriptions of input challenges. The results are presented in Table 11. *WRP* indicates *word recognition percentage*. It was observed that the automatic scripting was capable of identifying mere 8.36% of words in the CAPTCHA challenge audio. The inference derived here is not about the capability of Sphinx, the nature of the audio files was not favorable for ASR, which confirms the friction against automatic transcribing.

Steps 2, 3 and 4 require domain-specific knowledge bases to be built and real-time mapping has to be established between questions and transcribed text to generate the answer. We were unable to detect any major studies with the potential for carrying out all the tasks listed from step 1 to step 4, as of writing of this paper. The AI-based complete question answering systems [53] are still evolving and are not mature enough to be employed for solving HuMan CAPTCHA challenges efficiently.

Moreover, it has to be noted that CAPTCHA is not going to be used as a stand-alone authentication service such as a password or biometrics which protects critical interfaces such as e-banking. CAPTCHA functions as a filter to detect whether the access is by a human or machine. Hence, the cost–benefit analysis of building such a complex, resource-heavy system to break the CAPTCHA would not be favorable for any potential hacker, in comparison with breaking of aforementioned authorization services.

- **Randomness** The CAPTCHA selection process should always include randomness. Most of the audio CAPTCHA services include only one layer of randomness in selecting the audio. In contrast, the HuMan model includes two layers of randomness: one for selecting the CAPTCHA audio and another for selecting the question to be presented in the current instance from a set of predefined questions associated with that corresponding audio challenge.

Hence, obfuscation of media, complexity involved in script level breaking and double-layer randomness increase the security of the proposed HuMan model to an acceptable level.

Table 12 Sample HuMan challenge with polymorphic response

Item	Description
HuMan audio	Passengers Attention please. One Six Seven Zero One, Chennai Rameswaram Express from Chennai via Trichy to Rameswaram is expected to departure 21 h 40 min from Platform number 5
Challenge #1	Identify the train number mentioned in the audio?
Challenge #2	What is the destination station of the train?
Challenge #3	From which platform is the train leaving?
Challenge #4	What is the type of train mentioned in the audio?
Challenge #5	When is the train departing from the station?
Challenge #6	Identify the intermediate station?

The human proxy-based attack is another form of attack wherein the CAPTCHA shall be redirected to human workers employed particularly for the purpose of breaking the CAPTCHA. As the HuMan CAPTCHA incorporates the personalization element, the presented audio with a semantic challenge would require additional attention than breaking the non-semantic counterpart for a CAPTCHA relay worker. Moreover, relaying of the text-based CAPTCHA is trivial as it requires only a screen shot of the CAPTCHA image, whereas in the case of HuMan audio CAPTCHA sophisticated methods such as streaming need to be employed in redirecting the challenge to a human proxy. Nevertheless, designing a CAPTCHA system which is 100% fail safe against human proxy would be violating the very purpose of incorporating a CAPTCHA (i.e., to differentiate a human from a machine).

For example, Table 12 shows that the specified audio challenge has six possible questions, and hence, at various instances the answer to the CAPTCHA would depend on the challenge thrown at the current instance. Another human-friendly feature present in the proposed model is the dependence of common sense knowledge while answering the questions. For example, answering the question “What is the type of train mentioned in the audio?” requires the common sense knowledge that trains are of different types such as express and passenger. For an automated attack to crack the above challenge, even if the audio is recognized fully and converted to text, the dependency of human-friendly common sense knowledge makes it hard for the bots. Similarly challenge 6 requires the human knowledge that the first sequence of digits announced is the train number, while answering it. In many audio CAPTCHA systems, the challenges are unique across (identifying a particular sound or letter or digit) all provided samples where as in the proposed HuMan CAPTCHA each associated question with challenge requires different types of inferences need to be applied.

It has been accepted by pioneering studies in the field of CAPTCHA for visually impaired that the security and usability of CAPTCHA have an inverse relationship among

them [43]. Hence, if the security aspect of the CAPTCHA is fully optimized, then it would become harder for the visually impaired users to solve them. However, the presence of real-time noise which is not easily separable, the semantic nature of the challenges, polymorphic response nature makes the HuMan CAPTCHA model resistant to bots and friendlier to the human user, which is the primary objective.

4.11.1 Real-time checks

Apart from the aforementioned measures, in widespread real-time implementations of the proposed HuMan model, the following bot detection techniques shall be adopted:

- Inclusion of *response time boundary (RTB)* which poses a condition that after the presentation of CAPTCHA audio challenge, the response has to be given within a time limit (set to a minimal value), shall be adopted. If the HuMan CAPTCHA needs to be broken automatically, then the possibility of total time needed to relay the CAPTCHA and to perform aforementioned four steps exceeding the RTB is significantly higher. Repeated requests violating the RTB shall be identified as bots;
- The CAPTCHA preemption index (CPI) was observed around 50% in the experiments. Hence, if large numbers of requests are originating from same IP address or a geographic region, and no preemption is applied (all audio is played completely), then such requests shall be identified as bots;
- There is a strong possibility that answers provided by the human user would not match exactly with the result. This is the reason for inclusion of fuzzy comparison for answers. The repeated requests with no preemption and exact matching answer shall be suspected as bots.

As the design of the HuMan model allows to perform the aforementioned checks without much effort, the model shall be considered for providing thematic CAPTCHAs in web pages.

Table 13 HuMan model satisfaction measures

Measure	Description
HM.STS	Solving time satisfaction
HM.SRS	Success rate satisfaction
HM.PUS	Preemption usage satisfaction
HM.PRS	Polymorphic relation satisfaction
HM.ICC	Interest in CAPTCHA challenge
HM.EoU	Overall ease of use

4.12 User satisfaction analysis

To measure the satisfaction of the users with the proposed HuMan CAPTCHA model, it was decided to gather inputs across six different measures as shown in Table 13 (the prefix HM in all five measures represents the HuMan model).

The data with respect to the aforementioned six measures were gathered from all users after the experiments, in order to get an insight into the satisfaction of the users with the HuMan CAPTCHA model. The data are gathered in a scale of 1 to 5 (Likert scale). The higher the value on the scale, the better the satisfaction level of user. The mean and standard deviation of the data gathered are shown in Table 14.

4.12.1 System usability survey (SUS)

At the end of each session, the users were asked to fill in a system usability survey questionnaire [8]. The SUS consisted of ten questions and users feedback was received in a

scale of 1 to 5. The compiled results of SUS after the completion of all fourteen sessions are tabulated in Table 15.

The SUS consists of both positive and negative response category questions as indicated in Table 15 as P and N, respectively. For P-type questions, the objective is to maximize the response value, and for the N-type questions, it is to minimize the value. The overall results of the SUS feedback are measured in a range of 0–100. The SUS results for persons with visual impairments was observed as 82.44 which indicates better usability of the proposed model with visually impaired users. Similarly, the overall SUS results for sighted users were observed as 82.63 which confirms the user satisfaction of the proposed HuMan model with the sighted users.

4.13 Limitations of the HuMan CAPTCHA model

Though the proposed HuMan model exhibits significant improvements with the incorporation of five novel dimensions, it has certain limitations as listed below:

- The current implementation of HuMan requires the challenges to be built manually. Efforts need to be taken for automatic (or semiautomatic) generation of questions for the audio clips utilized as CAPTCHA challenges;
- In the current implementation, challenge audio, question and answers are provided in English. To enhance the user experience of non-native speakers, regional language-based challenges shall be presented

Table 14 Mean and standard deviation of user satisfaction measures

Measure	HM.STS	HM.SRS	HM.PUS	HM.PRS	HM.ICC	HM.EoU
Mean	4.35	4.25	4.49	4.17	4.32	4.14
Standard deviation	0.66	0.73	0.58	0.68	0.71	0.66

Table 15 HuMan CAPTCHA model—system usability survey

S. no.	Cat.	Survey question	VI.MR	SU.MR
1	P	I think that I would like to use this system frequently	3.976	4.017
2	N	I found the system unnecessarily complex	1.452	1.589
3	P	I thought the system was easy to use	4.083	4.071
4	N	I think that I would need the support of a technical person to be able to use this system	1.988	2.053
5	P	I found the various functions in this system were well integrated	3.964	3.928
6	N	I thought there was too much inconsistency in this system	1.488	1.446
7	P	I would imagine that most people would learn to use this system very quickly	4.476	4.571
8	N	I found the system very cumbersome to use	1.571	1.428
9	P	I felt very confident using the system	4.476	4.392
10	N	I needed to learn a lot of things before I could get going with this system	1.50	1.410

- In its present form, the HuMan CAPTCHA challenges are presented only in audio format which makes it inaccessible to persons with hearing disabilities. To accommodate those users, the textual representation of the challenges with some sort of noise shall be accommodated.
- The HuMan model mandates the user to key in the answer to the presented challenge via the keyboard. The future implementation shall allow users to *say* it, which would reduce the solving time and entry errors.

5 Conclusions and future directions

CAPTCHA, which serves as the entry check mechanism in web interfaces, has generated friction in access for majority of users in general and visually impaired in particular. Among the various CAPTCHA modes, audio CAPTCHA are comparatively better accessible to visually impaired. This paper has proposed a model for providing enhanced audio CAPTCHA with specific features for the web interfaces. The proposed HuMan CAPTCHA is designed for the aural channel making it best suited for non-visual access.

CAPTCHA are primarily created to provide security for the web resources with minimal friction for the legitimate users interacting with the system. This requirement is incorporated into the HuMan CAPTCHA model, with the idea of personalization. The basic idea employed here is that the users would rather prefer to face the challenges in those domains in which they have interest than in a random domain.

The HuMan model provides personalization based on implicit and explicit preference gathering mechanisms. For the prototype implementation three different domain interfaces, sports commentary, travel announcements and dynamic web contents is built. Using these domain interfaces, various challenges were generated. The model is flexible enough to accommodate customized domain interfaces. The five dimensions associated with the HuMan model are (a) accessible, (b) polymorphic, (c) semantic, (d) personalized and (e) preemptive.

The CAPTCHA challenges set using the HuMan model are semantic in nature. The answers to the challenges are identifiable without much disturbances to the human users, whereas for the automated bots this would require higher levels of artificial intelligence mechanisms to come closer toward breaking them.

Moreover, the polymorphic nature of the HuMan CAPTCHA makes the automated solutions much more difficult. Each challenge in the HuMan model is associated with more than one question which is measured using a simple metric proposed by this paper, called mean polymorphic index (MPI).

The HuMan CAPTCHA has another significant advantage of preemption which means that it is not always required to listen to the complete CAPTCHA audio for answering the challenge. As the question is announced prior to playing the audio, the user can skip the remaining portions of the CAPTCHA, as soon as the answer is identified. The CAPTCHA mean preemption index was observed as 0.514 during the experiments.

The combinatorial effect of the five dimensions makes the HuMan model both easier and effective in providing CAPTCHA challenges to persons with visual impairments and sighted users, which is supported with the data gathered through experiments and the feedback received from the users.

Though the HuMan CAPTCHA model has shown encouraging results in the form of MSR and user satisfaction levels, there is scope for further improvements. The requirement of human involvement in generating the polymorphic challenges shall be considered as a bottleneck in the proposed model. The future directions for this research work shall include the following:

- Extending the HuMan model to incorporate the specialized requirements for people with other disabilities such as motor impairments and multiple disabilities;
- Inclusion of interfaces such as music, product advertisement domains and incorporation of localization features with support for regional languages;
- Enhancing the HuMan model by focusing on specific CAPTCHA interfaces for mobile web rendering in smartphones;
- Developing a CAPTCHA rating mechanism by users based on the failure rate associated with individual challenges.

Along with the features incorporated in the HuMan model, the aforementioned future directions would further enhance the usability of the proposed model. The proposed HuMan model makes it easier for the visually impaired users in solving a CAPTCHA by making the process of solving interesting and enjoyable.

Appendix I: HuMan CAPTCHA Algorithm

Algorithm 1 HuMan Captcha Algorithm

```

1: procedure HUMAN(Page p)
2:   if detectTrace(p) then
3:      $\mu \leftarrow I$ 
4:   else
5:      $\mu \leftarrow E$ 
6:   if Equals( $\mu, E$ ) then
7:      $\kappa \leftarrow \text{renderUItoGetPref}()$ 
8:      $\iota \leftarrow \text{fetchDomain}(\kappa)$ 
9:   else if Equals( $\mu, I$ ) then
10:    if userPerm(cookies, ip) then
11:      if isSet(H.cookies) then
12:         $\epsilon \leftarrow \text{parse}(H.cookies)$ 
13:         $\iota \leftarrow \text{domain}(\epsilon)$ 
14:      else if ipFound(TrackDB, ip) then
15:         $\rho \leftarrow \text{fetchValuesForIP}(ip)$ 
16:         $\iota \leftarrow \text{domain}(\rho)$ 
17:    else
18:       $tp \leftarrow \text{FetchText}(p)$ 
19:       $src \leftarrow tp \cup \text{getTitle}(p) \cup \text{getMeta}(p)$ 
20:       $kwrd\text{s} \leftarrow \text{FetchKeyTerms}(src)$ 
21:       $\iota \leftarrow \text{MatchDomain}(kwrd\text{s}, DI)$ 
22:     $maxCount \leftarrow \text{Count}(HuManDB, \iota)$ 
23:     $challengeID \leftarrow \text{genRandom}(1, maxCount)$ 
24:     $maxQnCount \leftarrow \text{Count}(Qn, challengeID)$ 
25:     $qnID \leftarrow \text{genRandom}(1, maxQnCount)$ 
26:     $RenderCaptcha(challengeID, qnID)$ 
27:     $uResponse \leftarrow \text{getUserResponse}()$ 
28:     $cAns \leftarrow \text{getCorrectAnswer}(challengeID, qnID)$ 
29:     $\tau \leftarrow \text{fetchThreshold}(\iota)$ 
30:     $\phi \leftarrow JW(uResponse, cAns)$ 
31:    if  $\phi \leq \tau$  then
32:       $\gamma \leftarrow \text{getPreemptionPoint}()$ 
33:       $ti \leftarrow \text{buildTrackingInfo}(\mu, \gamma)$ 
34:       $\text{addTrackingInfo}(challengeID, qnID, ti)$ 
35:      return(true)
36:    else
37:       $\gamma \leftarrow \text{getPreemptionpoint}()$ 
38:       $\lambda \leftarrow \text{getErrorMargin}()$ 
39:       $ti \leftarrow \text{buildTrackingInfo}(\mu, \gamma, \lambda)$ 
40:       $\text{addTrackingInfo}(challengeID, qnID, ti)$ 
41:      if  $fRate(qnID) \geq \nu$  then
42:         $\text{addToReview}(qnID)$ 
43:      else if  $fRate(challengeID) \geq \xi$  then
44:         $\text{addToReview}(challengeID)$ 
45:      if  $\text{count}(uResponse, TrackDB) \geq T$  then
46:        if approve(uResponse) then
47:           $\text{updateAns}(challengeID, qnID, uResponse)$ 
48:      return(false)

```

HuMan algorithm receives the url of a page as input. With *detectTrace* function, it checks for any trace of visit of the same page earlier from the same device. Based on this, it renders the preferences menu (*renderUItoGetPref()*) for the *explicit* scenario (*E*) and allows the user to choose the domain (*fetchDomain()*).

In the case of the *implicit* scenario (*I*), the domain is selected either from the *cookies* or *IP address* from *TrackDB*.

When both cookies and IP scenarios are not available, a thematic domain is chosen (*MatchDomain()*) by fetching

title (*getTitle()*), meta-keywords (*getMeta()*) and keywords (*FetchKeyTerms()*) from the page.

Once the domain is chosen, a random challenge (*challengeID*) is selected. After the selection of a challenge, one question (*qnID*) is selected randomly from the list of available questions for that challenge.

Then HuMan CAPTCHA constructed with specified *challengeID* and *qnID* is rendered (*RenderCAPTCHA(challengeID, qnID)*).

Response from the user is gathered in *uResponse* through *getUserResponse()*. The correct answer for the challenge (*cAns*) is fetched through *getCorrectAnswer()*.

As the HuMan model incorporates fuzzy answer matching to incorporate spelling errors, the Jaro–Winkler edit distance is computed ($JW(uResponse, cAns)$) between the user response and the correct answer. If it less than the threshold value (τ), then *true* is returned from the algorithm indicating successful CAPTCHA solving. Before this, the preemption point is recorded updated through *addTrackingInfo()*. The tracking information is used at the server side to check number of times this CAPTCHA is answered correctly.

If the edit distance is above the threshold value (τ), then preemption point (*getPreemptionpoint()*), error margin (*getErrorMargin()*) is computed and the tracking information is updated. The algorithm returns *false* indicating failure of CAPTCHA solving attempt.

If the failure rate for a challenge audio or specific question belonging to that challenge crosses the specified threshold values (ν, ξ), it is added to review (*addToReview()*) by the administrator.

If a response to a CAPTCHA receives a specific incorrect answer many times (above a threshold), then with the approval of administrator the answer is updated for that CAPTCHA.

References

- Baird, H.S., Coates, A.L., Fateman, R.J.: Pessimalprint: a reverse turing test. *Int. J. Doc. Anal. Recognit.* **5**(2–3), 158–163 (2003)
- Baird, H.S., Moll, M.A., Wang, S.Y.: Scattertype: a legible but hard-to-segment captcha. In: Eighth International Conference on Document Analysis and Recognition. Proceedings, pp. 935–939. IEEE (2005)
- Bauwens, B., Evenepoel, F., Engelen, J.: Sgml as an enabling technology for access to digital information by print disabled readers. *Comput. Stand. Interfaces* **18**(1), 55–69 (1996)
- Belk, M., Fidas, C., Germanakos, P., Samaras, G.: Do human cognitive differences in information processing affect preference and performance of captcha? *Int. J. Hum. Comput. Stud.* **84**, 1–18 (2015)
- Belk, M., Germanakos, P., Fidas, C., Holzinger, A., Samaras, G.: Towards the personalization of CAPTCHA mechanisms based on individual differences in cognitive processing. In: Holzinger, A.,

- Zieffle, M., Hitz, M., Debevc, M. (eds.) *Human Factors in Computing and Informatics, Lecture Notes in Computer Science*, vol 7946, pp. 409–426. Springer, Berlin (2013)
6. Berners-Lee, T.: Long live the web. *Sci. Am.* **303**(6), 80–85 (2010)
 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*, pp. 1829–1838. ACM (2009)
 8. Brooke, J., et al.: Sus-a quick and dirty usability scale. *Usability Evalu. Ind.* **189**(194), 4–7 (1996)
 9. Bursztein, E., Bethard, S.: Decaptcha: breaking 75% of ebay audio captchas. In: *Proceedings of the 3rd USENIX conference on Offensive technologies*, p. 8. USENIX Association (2009)
 10. 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*, pp. 399–413. IEEE (2010)
 11. Bursztein, E., Martin, M., Mitchell, J.: Text-based captcha strengths and weaknesses. In: *Proceedings of the 18th ACM conference on Computer and communications security*, pp. 125–138. ACM (2011)
 12. Campbell, D., Stanley, J.: Experimental and quasi-experimental designs for research. In: Gage, N.L. (ed.) *Handbook of research on teaching*, pp. 171–246. (1966)
 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*, pp. 711–720. ACM (2005)
 14. Converse, T.: CAPTCHA generation as a web service. In: Baird, H.S., Lopresti, D.P. (eds.) *Human Interactive Proofs, Lecture Notes in Computer Science*, vol 3517, pp. 82–96. Springer, Berlin (2005)
 15. Darejeh, A., Singh, D.: An investigation on ribbon interface design guidelines for people with less computer literacy. *Comput. Stand. Interfaces* **36**(5), 808–820 (2014)
 16. Datta, R., Li, J., Wang, J.Z.: Imagination: a robust image-based captcha generation system. In: *Proceedings of the 13th Annual ACM International Conference on Multimedia*, pp. 331–334. ACM (2005)
 17. Davidson, M., Renaud, K., Li, S.: jCAPTCHA: accessible human validation. In: Miesenberger, K., Fels, D., Archambault, D., Peñáz, P., Zagler, W. (eds.) *Computers Helping People with Special Needs, ICCHP 2014. Lecture Notes in Computer Science*, vol 8547, pp. 129–136. Springer, Cham (2014)
 18. Elson, J., Douceur, J.R., Howell, J., Saul, J.: Asirra: a captcha that exploits interest-aligned manual image categorization. In: *ACM Conference on Computer and Communications Security*, pp. 366–374 (2007)
 19. Gao, H., Liu, H., Yao, D., Liu, X., Aickelin, U.: An audio captcha to distinguish humans from computers. In: *2010 Third International Symposium on Electronic Commerce and Security (ISECS)*, pp. 265–269. IEEE (2010)
 20. Gossweiler, R., Kamvar, M., Baluja, S.: What’s up captcha?: a captcha based on image orientation. In: *Proceedings of the 18th International Conference on World Wide Web*, pp. 841–850. ACM (2009)
 21. Goswami, G., Powell, B.M., Vatsa, M., Singh, R., Noore, A.: Facedcaptcha: face detection based color image CAPTCHA. *Future Gener. Comput. Syst.* **31**, 59–68 (2014). doi:10.1016/j.future.2012.08.013. <http://www.sciencedirect.com/science/article/pii/S0167739X12001690>. Special Section: Advances in Computer Supported Collaboration: Systems and Technologies
 22. Goswami, G., Singh, R., Vatsa, M., Powell, B., Noore, A.: Face recognition captcha. In: *2012 IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pp. 412–417. IEEE (2012)
 23. 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*, pp. 245–246. ACM (2007)
 24. Kani, J., Nishigaki, M.: Gamified CAPTCHA. In: Marinos, L., Askoxylakis, I. (eds.) *Human Aspects of Information Security, Privacy, and Trust, HAS 2013. Lecture Notes in Computer Science*, vol 8030, pp. 39–48. Springer, Berlin (2013)
 25. Karshmer, A.I., Myler, H.R., Davis, R.D.: The architecture of an inexpensive and portable talking-tactile terminal to aid the visually handicapped. *Comput. Stand. Interfaces* **6**(2), 207–220 (1987)
 26. Kochanski, G., Lopresti, D.P., Shih, C.: A reverse turing test using speech. In: *Proceedings of the Seventh International Conference on Spoken Language Processing (ICSLP2002 - INTER-SPEECH 2002)*, Denver, Colorado, 16–20 Sept 2002, pp. 1357–1360. (2002). <http://www.isca-speech.org/archive/icslp02>
 27. Kuber, R., Yu, W.: Feasibility study of tactile-based authentication. *Int. J. Hum. Comput. Stud.* **68**(3), 158–181 (2010)
 28. 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*, pp. 2267–2276. ACM (2012)
 29. Lewthwaite, S.: Web accessibility standards and disability: developing critical perspectives on accessibility. *Disabil. Rehabil.* **36**(16), 1375–1383 (2014)
 30. Lupkowski, P., Urbanski, M.: Semcaptchauser-friendly alternative for ocr-based captcha systems. In: *International Multiconference on Computer Science and Information Technology. IMCSIT 2008*, pp. 325–329. IEEE (2008)
 31. Masek, W.J., Paterson, M.S.: A faster algorithm computing string edit distances. *J. Comput. Syst. Sci.* **20**(1), 18–31 (1980)
 32. Meutzner, H., Gupta, S., Kolossa, D.: Constructing secure audio captchas by exploiting differences between humans and machines. In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI ’15*, pp. 2335–2338. ACM, New York (2015). doi:10.1145/2702123.2702127
 33. Misra, D., Gaj, K.: Face recognition captchas. In: *Null*, p. 122. IEEE (2006)
 34. Moreno, L., González, M., Martínez, P.: Captcha and accessibility-is this the best we can do? *WEBIST* **2**, 115–122 (2014)
 35. Olalere, A., Feng, J.H., Lazar, J., Brooks, T.: Investigating the effects of sound masking on the use of audio captchas. *Behav. Inf. Technol.* **33**(9), 919–928 (2014)
 36. Pope, C., Kaur, K.: Is it human or computer? defending e-commerce with captchas. *IT Prof.* **7**(2), 43–49 (2005)
 37. Rose, S., Engel, D., Cramer, N., Cowley, W.: Automatic keyword extraction from individual documents. In: Berry, M.W., Kogan, J., (eds.) *Text mining: applications and theory*, Wiley, Chichester (2010). doi:10.1002/9780470689646.ch1
 38. Roshanbin, N., Miller, J.: A survey and analysis of current captcha approaches. *J. Web Eng.* **12**(1–2), 1–40 (2013)
 39. Roshanbin, N., Miller, J.: Adamas: interweaving unicode and color to enhance CAPTCHA security. *Future Gener. Comput. Syst.* **55**, 289–310 (2016). doi:10.1016/j.future.2014.11.004. <http://www.sciencedirect.com/science/article/pii/S0167739X14002386>
 40. Ross, S.A., Halderman, J.A., Finkelstein, A.: Sketcha: a captcha based on line drawings of 3d models. In: *Proceedings of the 19th International Conference on World Wide Web*, pp. 821–830. ACM (2010)

41. Sachdeva, M., Kumar, K., Singh, G.: A comprehensive approach to discriminate ddos attacks from flash events. *J. Inf. Secur. Appl.* **26**, 8–22 (2016)
42. Sauer, G., Holman, J., Lazar, J., Hochheiser, H., Feng, J.: Accessible privacy and security: a universally usable human-interaction proof tool. *Univ. Access Inf. Soc.* **9**(3), 239–248 (2010)
43. Sauer, G., Lazar, J., Hochheiser, H., Feng, J.: Towards a universally usable human interaction proof: evaluation of task completion strategies. *ACM Trans. Access. Comput. (TACCESS)* **2**(4), 15 (2010)
44. Schryen, G., Wagner, G., Schlegel, A.: Development of two novel face-recognition captchas: a security and usability study. *Comput. Secur.* **60**, 95–116 (2016)
45. Shirali-Shahreza, M., Shirali-Shahreza, S.: Advanced collage captcha. In: *Fifth International Conference on Information Technology: New Generations. ITNG 2008*, pp. 1234–1235. IEEE (2008)
46. Shirali-Shahreza, M.H., Shirali-Shahreza, M.: Multilingual captcha. In: *IEEE International Conference on Computational Cybernetics. ICC 2007*, pp. 135–139. IEEE (2007)
47. Shirali-Shahreza, S., Penn, G., Balakrishnan, R., Ganjali, Y.: Seesay and hearsay captcha for mobile interaction. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 2147–2156. ACM (2013)
48. Spitzer, M., Wildenhain, J., Rappsilber, J., Tyers, M.: Boxplotr: a web tool for generation of box plots. *Nat. Methods* **11**(2), 121–122 (2014)
49. Tam, J., Hyde, S., Simsa, J., Ahn, L.V.: Breaking audio CAPTCHAs. In: Koller, D., Schuurmans, D., Bengio, Y., Bottou, L. (eds.) *Proceedings of the 21st International Conference on Neural Information Processing Systems (NIPS'08)*, pp. 1625–1632. Curran Associates Inc., USA (2008)
50. Von Ahn, L., Maurer, B., McMillen, C., Abraham, D., Blum, M.: Recaptcha: human-based character recognition via web security measures. *Science* **321**(5895), 1465–1468 (2008). <http://www.sciencemag.org/content/321/5895/1465.short>
51. Walker, W., Lamere, P., Kwok, P., Raj, B., Singh, R., Gouvea, E., Wolf, P., Woelfel, J.: Sphinx-4: A flexible open source framework for speech recognition. Technical Report. Sun Microsystems, Inc., Mountain View, USA (2004)
52. Wei, T.E., Jeng, A.B., Lee, H.M.: GeoCAPTCHA: a novel personalized CAPTCHA using geographic concept to defend against 3rd Party Human Attack. In: *2012 IEEE 31st International Performance Computing and Communications Conference (IPCCC)*, pp. 392–399. IEEE (2012)
53. Weston, J., Bordes, A., Chopra, S., Mikolov, T.: Towards incomplete question answering: a set of prerequisite toy tasks. arXiv preprint [arXiv:1502.05698](https://arxiv.org/abs/1502.05698) (2015)
54. Winkler, W.E.: Overview of record linkage and current research directions. In: *Bureau of the Census. Citeseer* (2006). <http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.79.1519>
55. Woolson, R.F.: Wilcoxon signed-rank test. In: *Wiley Encyclopedia of Clinical Trials*. Wiley (2007). <http://onlinelibrary.wiley.com/doi/10.1002/9780471462422.eoct979/abstract>
56. Yang, T.I., Koong, C.S., Tseng, C.C.: Game-based image semantic CAPTCHA on handset devices. In: *Multimedia Tools and Applications*, pp. 1–16 (2013). <http://link.springer.com/article/10.1007/s11042-013-1666-7>
57. Zhou, J., Chin, W.Y., Roman, R., Lopez, J.: An effective multi-layered defense framework against spam. *Inf. Secur. Tech. Rep.* **12**(3), 179–185 (2007)
58. Zhu, B.B., Yan, J., Li, Q., Yang, C., Liu, J., Xu, N., Yi, M., Cai, K.: Attacks and design of image recognition captchas. In: *Proceedings of the 17th ACM Conference on Computer and Communications Security, CCS '10*, pp. 187–200. ACM, New York (2010). doi:[10.1145/1866307.1866329](https://doi.org/10.1145/1866307.1866329)