Robots are coming,
or who needs to defend against robots, anyway?

Evgeniy Gabrilovich gabr@cs.technion.ac.il
Alex Gontmakher gsasha@cs.technion.ac.il

Humans attack computers on a routine basis, while machines attacking humans are still science fiction. The new trend today involves computers attacking their own kin, and we better have ample defense against this threat.

Introduction

We live in a networked world, and many services that used to be provided by humans are now rendered by computers. Most of them are intended to be *used by humans only*, and their designers give little or no consideration to the fact that these services may also be used by automated programs. And here lies one of the most troubling vulnerabilities of today's online services, since a malicious hacker can easily write a program to impersonate a human and thus abuse a service for his own dark purposes.

The first example that comes to mind in this regard is an attack directed at Internet-based email systems, such as Hotmail or Yahoo. Spammers use massive numbers of email accounts to avoid being detected by spam filters. However, opening email accounts manually would cost spammers money, and will thus invalidate the very reason for profitability of spamming, namely, that sending the messages is virtually free. Fortunately (for the spammers), the open nature of the Web allows one to automate the creation of new accounts with minimal programming effort.

Naturally, online service providers do not like to waste their resources and reputation on spammers, and so they endeavor to develop techniques to protect themselves from such an abuse. One possible protection against scripters is to introduce a complex and time-consuming registration process, and keep changing it regularly to make hackers' lives harder. The problem is, this method does more harm than good. Complex registration process might well scare off legitimate users, while it is still easy to circumvent with modern programming tools.

The ultimate solution would make sure that services designed for humans are indeed used by humans only. To this end, we need to formulate a challenge that is very easy to solve for a human and very hard for a computer at the same time. Actual implementations of this approach will aim at activities where human intelligence by far surpasses the (infamous) artificial one. For example, the user can be presented with a picture containing a garbled image of a text string, and asked to type the contents of the string into a designated form field to proceed. Existing OCR programs can only recognize relatively clean images, so this effectively blocks scripters altogether.

But why would *you* need such protection? After all, not everybody routinely maintains online email systems. The truth is, there are plenty of other systems that need protection from robots.

Let us revisit the case for junk mail. The other thing spammers are thirsty for is access to long lists of email addresses (otherwise, whom will they bombard with their unsolicited letters?). As it happens, the liberal nature of the Internet gets routinely abused by robots that scan Web pages for potential targets. Not many users are aware of this risk, but those who care usually post their addresses in a garbled way, for example, "john [at] hotmail [dot] com" (instead of the more

convenient but easily captured "john@hotmail.com").[1] It is ironic that your email, one of the most important means of contact, must be kept a closely guarded secret!

Another example comes from the realm of online databases such as Yellow Pages. Malevolent users can download the entire contents of the database in order to perform reverse telephone lookups, which are considered illegal in many jurisdictions. This is not to mention that downloading and storing the whole database usually violates someone's IP rights.

The Solution – Reverse Turing Test

Alan Turing suggested the *Turing Test* (see sidebar) to distinguish between humans and computers, while he presumed that the judge is *human*. A variant of this test can be used to allow *computers* to discriminate human users from robots. The idea is to use riddles that can be easily answered by humans, but are still too challenging even for state of the art computer algorithms. Examples of such an approach involve modalities where human abilities by far exceed those of computers, for instance, interpreting heavily distorted images or understanding noisy speech recordings.

This new type of a quiz is referred to as the *Reverse* Turing Test (RTT). This notion was originally formulated back in 2000 by Udi Manber, then Chief Scientist of Yahoo!, in an attempt to solve the "chat room problem". This problem manifested itself in online discussion forums being plagued by chat-bots, which constantly hindered the discussion with aggressive advertisements and sales pitches. The idea was then picked up by the CAPTCHA[2] project at the Carnegie Mellon University, where a group of computer scientists designed and implemented a number of such tests (often called "captchas")

The most common type of RTT presents the user with a heavily distorted image of a character or digit string, and asks her to transcribe this sequence (see Figure 1 for examples). This line of defense relies on the (current) inability of computers to perform character recognition under deliberately hostile conditions.[3] Another option is to ask the user to complete a very simple series of images or figures. Using other modalities is also possible: yet another system developed within the CAPTCHA project superimposes two speech fragments in different languages, making the outcome only intelligible for the human ear. The crucial observation behind these approaches is that tests can be easily *generated* by a computer, but cannot be easily *solved* by any automated procedure (at least for the time being).

Robot defense based on RTT is used today in a variety of applications. Hotmail and Yahoo employ it to prevent unattended registration of email accounts. PayPal – a popular Internet payments provider – offers a small (but very real) incentive of $5 for opening an account with the company. To prevent its cash resources from evaporating all too quickly, PayPal uses RTT

---

[1] Observe that even this simple encoding carries a price tag, as it makes sending email slightly less convenient to other humans, who can no longer click on a mailto: link to have the email program open it automatically.

[2] CAPTCHA stands for Completely Automated Public Turing test to tell Computers and Humans Apart. The home page of the project can be reached at www.captcha.net.

[3] It has been reported in the literature that certain kinds of captcha images can be automatically recognized with fairly high precision (see http://www.cs.berkeley.edu/~mori/gimpy/gimpy.html), but undoubtedly the general method can be made to work if implemented properly. Interestingly, scientists view this situation in a positive light, since competition in this field will also promote machine vision algorithms to a decent level.

technology to safeguard opening new accounts. Altavista uses a similar protection to prevent automatic submission of URLs to be indexed by the search engine, and after the defense was put in place the number of submission was reported to drop by almost 90% virtually overnight. More recently, Spam Arrest LLC (www.spamarrest.com) proposed an interesting way to use RTT against junk email. Whenever somebody you don't know sends you an email for the first time, the system *quarantines* the letter and sends this person a riddle she must answer. When the riddle is solved correctly, the system considers the letter a legitimate one, and delivers it to the addressee.

Interestingly, some people use similar protection mechanisms even without knowing anything about Reverse Turing Tests. For instance, in order to prevent automatic collection of email addresses from their Web pages, people encode their addresses as <first four letters of last name> [at] mydomain.com, while most advanced users even prepare a small GIF image with their email address (see Figure 2).

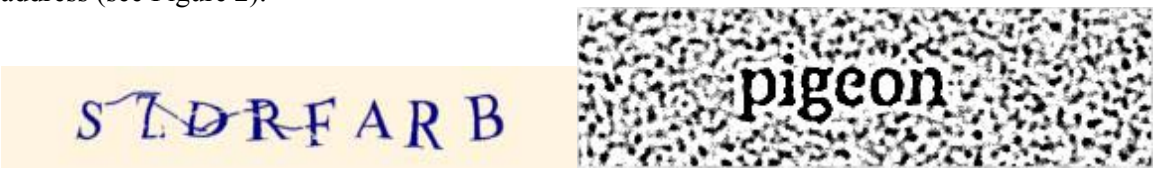

Figure 1: Sample captchas (required for registration for a new account with Hotmail and Yahoo!, respectively).

Figure 2: Garbled email address.

Conclusion

The future of RTT will inevitably become a race for arms, where good guys invent more sophisticated kinds of tests, and the bad guys upgrade their robots. Whether this contest will continue indefinitely, or finish with computers performing on par with humans, nobody knows. Whatever be the case, we will surely end up with some good technology at hand, and gain some security in the process.

Sidebar – Turing Test

The so-called Turing Test was conceived by the British mathematician Alan Turing, considered by many one of the forefathers of the modern computer science. Among his other achievements, Turing defined a prototypical computational model – the Turing Machine – that laid the foundation for contemporary theories of computation and complexity.[4]

The question whether machines can think started bothering people soon after the advent of computers in early 1950s. In an attempt to address this question, Alan Turing proposed a test designed to distinguish a computer from a human being, which works like this. A human Judge is sitting in room A before a computer monitor, which is connected to two additional monitors in rooms B and C. One of these rooms is occupied by another human Subject, while the other is occupied by a Computer pretending to be a human. The Judge poses a series of questions to be answered by the residents of rooms B and C. The Judge's aim is to eventually tell where the Computer is.

[4] Outside the computer science world Turing is most famous for breaking the German code machine Enigma during WWII.

This setting looks deceptively simple, as the human Subject can apparently scream "I'm a real person!" right from the outset. But then the Computer can do so too. A smart Judge would ask questions that can be easily answered by humans, but cause the computer to reply with gibberish, or otherwise make its presence obvious.

Several obstacles remain in place that make the Turing Test insurmountable even by today's best computers. These include natural language understanding skills and the scope of world knowledge possessed by people. (Naturally, the human Subject doesn't have to know everything, but she does know how to gracefully address a difficult question.)

The Turing Test is actually not a fixed recipe, but rather a template for a verification procedure that can be implemented in a variety of ways. In real life, there is a Loebner Prize Contest[5] conducted annually since 1990, which intends to award the Grand Prize of $100,000 and a Gold Medal for the first computer program whose answers would be indistinguishable from those of a human. To date, the Grand Prize has not yet been awarded, while the organizers routinely award a lesser prize of $2,000 and a Bronze Medal to the best contender.

A comprehensive survey of the past and present of the Turing Test is available at http://cogsci.ucsd.edu/~asaygin/tt/ttest.html.

About the authors

Evgeniy Gabrilovich is a Ph.D. student in Computer Science at the Technion – Israel Institute of Technology. He is a member of the ACM and the IEEE. His interests involve computational linguistics, information retrieval, and machine learning. He can be contacted at gabr@cs.technion.ac.il.

Alex Gontmakher is a Ph.D. student in Computer Science at the Technion – Israel Institute of Technology. His interests include parallel algorithms and constructed languages. He can be reached at gsasha@cs.technion.ac.il.

[5] The home page of the Loebner Prize is accessible at http://www.loebner.net/Prizef/loebner-prize.html.