**The Turing test and AI Benchmarking: from its Inception to the Present Day.**

Caoimhe Cahill [21331308] Olan Healy [21318204]

Aaron Maher [21337918] and Kevin Collins [21344256]

Computer Science Building, University Of Limerick, Castletroy, Limerick, Ireland V94 NX93

Module: CS4006 Intelligent Systems

**ABSTRACT**

"At some stage, therefore, we should have to expect the machines to take control." (Turing, 1996, p. 260) Alan Turing, one of the founders of artificial intelligence (AI), said these haunting words. Turing was one of the first to recognise the vast potential of machines to exceed human intelligence, a concept still relevant today. One of the most difficult challenges in AI development has been measuring machines' intelligence levels. The question "Can machines think?" posed by Turing in his 1950 paper has since become a significant question in AI. In response, Turing devised the Turing Test, which analyses a machine's capacity to demonstrate intelligent behaviour comparable to a human's. Since then, the Turing Test has been a widely used benchmark for assessing AI intelligence. However, its weaknesses and flaws have led to the development of new benchmark tests. This essay examines the inception of the Turing Test, its influence on AI benchmarking, and the first tests to pass. It will explore the limitations of the Turing Test and how it has paved the way for the development of other benchmark tests. Additionally, it will discuss different cases of AI benchmark tests and modern-day AI testing.

**1 The Turing Test**

The Turing Test is one of the most well-known benchmarks in AI. Also known as the Imitation Game, the Turing Test is a method of determining whether or not a machine can think like a human being. The test is named after its founder, Alan Turing, an English computer scientist, mathematician, cryptanalyst, theoretical biologist, and philosopher. Turing released his 1950 paper titled "Computing Machinery and Intelligence", initially published in the leading philosophy journal "Mind", which proposed his idea of a test, now known as the Turing Test, that aims to solve the question "Can machines think?" (Turing, 1950, p. 433).

**1.1 The Imitation Game: Turing's Test for Machine Intelligence**

He begins by presenting this problem in terms of a game known as the "Imitation Game". It involves no use of machines or AI but a man, a woman, and an interrogator whose gender is unimportant. The interrogator is in a room, away from the man and woman. The interrogator's objective is determining which of the two participants is a woman. The man's aim is to try to convince the interrogator that he is the woman and the other is not, whilst the woman's aim is to guide the interrogator to the correct choice. The interrogator may ask any question that they like. Turing suggests the questions and answers should be in natural language and written or, better yet, teleprinter communicating between the two rooms – the 1950s equivalent to electronic messaging such as instant messaging and email. Then Turing asks :

"'What will happen when a machine takes the part of A in this game?' Will the interrogator decide wrongly as often when the game is played like this as he does when the game is played between a man and a woman? These questions replace our original, 'Can machines think?'" (Turing, 1950, p. 434)

The test is successful if the machine convinces the interrogator that it is the woman. It's clear that neither the man in the gender-based test nor the machine in the machine-woman test is a female. It can be seen that Turing proposes comparing the machine's success to that of the man rather than whether it 'beats' the woman. They can be reached as they try to imitate something they are not. (Saygin et al 2000, p. 464 - 468)

He believed this change would draw a fairly sharp line between a man's physical and intellectual capacities. He thought this question would distinguish the mind from the man rather than comparing an intelligent machine to a man. "We do not wish to penalise the machine for its inability to shine in beauty competitions, nor to penalise a man for losing in a race against an aeroplane." (Turing 1950, p. 435).

It is possible to suggest that Turing's concept of thinking is that the computer in the test could impersonate a woman. However, it is likely that Turing intended to put forward a trial in which the computer must simulate a woman instead of a test that measures its ability to impersonate a human being.

Later in the article, Turing described a different situation, saying

'Let us fix our attention on one particular digital computer C. Is it true that by modifying this computer to have adequate storage, suitably increasing its speed of action, and providing it with an appropriate programme, C can be made to play the part of A satisfactorily in the imitation game, the part of B being taken by a man?' (Turing 1950, p. 442).

This contradicts the idea that the computer is supposed to imitate a woman.

In addition, in his lecture 'Can Digital Computers Think?', which was broadcasted on BBC Radio on the 15th of May 1951, Turing presents the point of the test is to determine whether or not a computer can 'imitate the brain' (Turing & Copeland, 2013, p. 436).

The Turing Test today usually describes the woman as a person of either gender, and the interrogator must determine which one is the human being. It is also sometimes described as a single room containing either a person or a machine, and the interrogator must decide whether they are talking to a human being or a machine. Although these variations differ slightly from the original test, it is generally agreed that it doesn't change the primary purpose of the test. (French 2000, p. 116)

**1.2 Chatbots and Controversy: Assessing the True Success of the Turing Test**

Despite significant advances in AI, it is still argued that computers still need to pass the Turing test. However, there have been some strong contenders. Many chatbots were developed, inspired by the Turing Test. The first known chatbot, Eliza, was created by Joseph Weizenbaum in 1966 and is one of the earliest chatbots made. It was constructed to take the role of a therapist and was given handcrafted scripts. It did not understand conversations but instead searched for appropriate responses and keywords in the scripts. Eliza would repurpose the user's input as a question and relay it back if it did not understand. Its abilities were limited, and it could only chat with people in a restricted domain, so it did not pass the Turing Test.

1.2 A typical conversation with Eliza

Here is an example of the chat log provided by Wezenbaum:

User: Men are all alike.

ELIZA: IN WHAT WAY

User: They're always bugging us about something or other.

ELIZA: CAN YOU THINK OF A SPECIFIC EXAMPLE

User: Well, my boyfriend made me come here.

ELIZA: YOUR BOYFRIEND MADE YOU COME HERE

User: He says I'm depressed much of the time.

ELIZA: I AM SORRY TO HEAR YOU ARE DEPRESSED

(Weizenbaum 1966, p.36-37)

The Parry chatbot was developed by Kenneth Colby in 1972. Colby used a similar technique to Weizenaum, but instead of repurposing questions, it was designed to avoid questions and change the topic as much as possible. It was created to model the behaviour of a paranoid schizophrenic. It's said to have passed the Turing Test with psychiatrists and psychologists in certain situations. However, for the same reasons as Eliza, it did not officially pass the Turing Test.(Shum et al 2018, p 11-12)

Also inspired by Weizenbaum's Eliza, the Alice chatbot was developed by Dr. Richard Wallace in 1995. Alice is a natural language open-source AI chatbot which uses artificial intelligence markup language to respond to queries. It has won the Loebner Prize three times. The Loebner Prize was a prize given to the computer program considered the most human-like by judges in an annual Turing Test formatted competition. Although won three times and coming close to passing, it is still considered to have yet to pass the Turing Test as everyday users still often spot its faults in short conversations. It is also due to the fact that it is based on a set of pattern-matching rules rather than true AI. (Sharma et al. 2017, p.53)

Recently, in 2014, a computer program named Eugene Goostman, created by Vladimir Veselov and Eugene Demchenko was said to have passed the Turing Test. A series of test experiments were carried out by the Royal Society of London on the 6th and 7th of June 2014. In his 1950 paper, Turing stated:

"I believe that in about fifty years' time, it will be possible to programme computers,..., to make them play the imitation game so well that an average interrogator will not have more than 70 per cent chance of making the right identification after five minutes of questioning."

Based on this, the Royal Society of London stated that to pass the test, the interrogator must make the wrong choice over 30% of the time after 5-minute long conversations. (Warwick & Shah 2016, p. 990) The bot successfully convinced 33% of the judges that it was human, being the first machine to pass the Turing Test. However, there is a lot of speculation surrounding this. One of the main problems being the persona they gave the bot may have made it easier for it to pass as a human. They gave it the persona of a 13-year-old Ukrainian boy who speaks English as a second language, which would allow sentences that did not make sense and obvious mistakes to slip by. Some might ask, is that not the point? To make it seem like they are human? Wouldn't a 13-year-old Ukrainian boy who speaks English as a second language make those mistakes? While this is a valid argument, you also have to ask if Eugene was 'intelligent' enough to make those mistakes on purpose or were actual mistakes made in the program itself. If the latter, can you say that program was intelligent enough to pass the Turing Test? We also believe that this is different from what Turing had in mind. Turing only talks about a general ability to impersonate a human, not the ability to impersonate a specific person, which could make it easier. The same problem goes for the Parry Chatbot mentioned above.

It is also important to note that although that quote came directly from Turing, he did not specifically say that this was the criteria for passing. It was just his prediction for the year 2000. So is 5 minutes a sufficient amount of time to determine if it is a human or a machine? Is 30% a high enough percentage to aim for, or should we aim higher?

Overall, while some say we have passed the Turing Test, it is not universally agreed, and we also consent that we have not passed the test in all situations.

**2 The Chinese Room Experiment**

Without a doubt, Alan Turing's 'Turing test' has played a significant role in determining whether a machine can exhibit intelligent behaviour, which is indistinguishable from a human. However, there are also limitations to his test, and many other computer scientists do too. The Chinese room argument against the Turing test was proposed by philosopher John Searle. The thought experiment demonstrates that an AI system could pass the Turing Test by manipulating symbols (language) without actually understanding the underlying concepts.

In his experiment, John pretends to lock himself in a room with a box of characters (in Chinese) that he can't understand with a book of instructions which he can. He then, obviously like the Turing test, has the Chinese speaker outside the room, passing in messages in Chinese which Searle can follow with his book of instructions to deliver an appropriate response. The person outside the room would clearly believe she is talking with a Chinese speaker. However, if a robot were to do the same task, it too wouldn't understand Chinese. It is simply following the instructions like Searle himself. The robot is only simulating the knowledge of actually understanding Chinese. This goes against the Turing test, as simulating knowledge is not actually thinking of its own accord. This limitation highlights the need to evaluate AI systems based on their comprehension of concepts rather than just their linguistic capabilities.

Speaking about intelligence, we would strongly agree with a quote from 20th US/Indian Philosopher Jiddu Krishnamurti who states that intelligence is:

"The ability to observe without evaluating is the highest form of intelligence".

When thinking about the Turing test, the test encourages the AI system to deceive test evaluators by replicating human behaviour. Although Jiddu has nothing to do with AI, his quote correlates to the Turing test. The AI systems are not showing the ability to "observe without evaluating"; they are really just simulating intelligence like the Chinese Room argument. The Turing test promotes trickery over genuine intelligence. This can lead to the development of AI systems that don't actually show intelligence like what Alan wants the test to demonstrate. Instead, they are adept at producing human-like responses but need more proper understanding and problem-solving abilities. This limitation highlights the need for evaluation methods that focus on the authenticity of intelligent behaviour rather than its mere appearance.

As the Turing test has been used as an evaluation method for AI, it has long been criticised for its lack of clear objective criteria. It relies on a subjective judgement of human evaluators, which can lead to inaccuracies in the evaluation process, making it difficult to compare the performance of various AI systems fairly. The Turing Test is an existence proof, not a performance test, as Moor (2001) notes (p. 78), emphasising the need for more reliable and impartial AI benchmarking techniques. Future AI benchmarking techniques should include unbiased, quantitative criteria that can precisely assess an AI system's performance in order to get beyond this constraint. With the use of these metrics, assessments might be made in a more consistent and reliable manner, allowing researchers and developers to compare and contrast different AI systems effectively and eventually fostering the development of AI technology.

A "more systematic and complete method to defining and assessing intelligence" (p. 22), as suggested by Legg and Hutter (2007), is required. This calls for the creation of alternative assessment techniques that consider unbiased, quantitative measurements across several aspects of intelligence, including problem-solving, reasoning, learning, and adaptation. AI benchmarking can offer a more thorough knowledge of an AI system's capabilities by including these objective criteria, opening the door for the creation of brilliant machines.

An address made to improve on the limitations of the Turing test was introduced by Hanard in 1991 and became known as the total Turing test. This was a significant benchmark in the world of AI. The total Turing test aim was to consider an AI's system ability to process and produce not only linguistic but also perceptual, auditory and motor responses.

"The Turing test is not just about what the candidate can say, but about what the candidate can do" ~ (Harnad, 1991, p. 43).

Going off on talking about the Chinese Room Experiment, to genuinely exhibit human-like intelligence, an AI system must do more than engage in "conversation". An AI system should also be able to interact with the world like humans do, such as perceiving and manipulating objects.

**3 The Total Turing Test: Assessing AI's Linguistic, Visual, Auditory, and Motor Abilities**

The Total Turing test evaluates an AI system based on a number of different abilities. Similar to the Turing Test, the same type of conversation takes place with the AI system and an actual human in a room separate from the evaluators. They are asked a series of questions and are asked to respond. Same as the Turing Test, this assesses the system's linguistic capabilities. Improving on the Turing test, it also tests the AI system by examining how it understands visual and auditory inputs. To show 'human-like intelligence', you should be able to act like a human does. This part of the test can be split into visual and audio recognition. For visual, if an AI system was presented with an object, it should be expected to recognise what the thing is (or be fairly accurate), and you could also show it a scene (e.g. busy street), and the system should be able to describe what is happening. It should be able to identify human faces and should be able to recognise human faces (same as humans once they actually recognise the face). "Understanding a scene, recognising objects and faces, and navigating through the world are among the most basic human abilities.

If a machine is to pass the Total Turing test, it must be able to do these things as well as humans can (Russell and Norvig, 2020, p. 726). Now for the audio side, in this test, we can present an AI system with various audio clips, and the system should be able to identify and classify these sounds from the clips. The Total Turing test also looks at speech recognition and understanding. It should be able to understand the chosen language spoken to them in different types of accents etc. The Total Turing test should be done in a controlled environment and is considered successful in passing the total Turing test if its performance is indistinguishable from that of humans. It should also be tested on its motor capabilities, such as performing tasks a human can do, e.g. navigating through an environment or assembling a piece of furniture.

"A machine that passes the Total Turing Test should be able to do anything a human can do because the test requires the machine to be indistinguishable from a human" ~ (Russell and Norvig, 2020, p. 1029).

There are drawbacks to using the Turing Test to assess AI systems, such as the China Room argument and its emphasis on deceptive imitation. The test has drawn criticism for lacking objective standards, which prompted the creation of substitute techniques. These problems are addressed by the Whole Turing Test, which Harnad suggested in 1991. It evaluates AI systems based on their linguistic, perceptual, auditory, and motor responses. Even though this test represents a substantial advancement, the search for really intelligent robots must continue, necessitating continued research and improvement of AI evaluation methods.

**4 Chess Computing**

When it comes to the topic of AI benchmarking and evaluating the intelligence of an AI compared to that of a human, one of the most interesting cases to look at is that of Deep Blue and chess-computing.

Chess-computing first became a point of discussion when Claude E. Shannon published his paper "Programming a Computer for Playing Chess". In it, he theorises how one could go about creating a chess-playing computer.

"Although no practical importance, the question is of theoretical interest, and it is hoped that a satisfactory solution to this problem will act as a wedge in attacking other problems of a similar nature and of greater significance. ~ (Claude E Shannon, 1949)

Since in chess, there are no "right" or "wrong" moves. Moves can only be evaluated by how advantageous of a position they put you in. He proposes two types of strategy, type A and type B.

- Type A is a computationally intensive design, exploring every possible combination of moves. While, in theory, a strong option, he figures that it would need to be faster, as the number of available options becomes exponentially bigger the further you analyse the game.

- Type B would be a more sophisticated design, determining which moves gave the greatest opportunity and discarding those that had little to offer. By doing this, it could ignore many possible moves that held little value and save on computational power. Furthermore, the moves that did offer value could be deeply analysed, allowing the computer to play much further into the future.

In a way, this type B strategy would mimic how professional human chess players would play, only evaluating certain states and ignoring meaningless ones. It is then valid to ask if a computer can outclass man at this. Could it be considered intelligent?

"Chess is generally considered to require "thinking" for skilful play; a solution of this problem will force us either to admit the possibility of a mechanised thinking or to further restrict our concept of "thinking"." ~ (Claude E. Shannon, 1949)

Many would try their hand at creating a chess-playing computer, but it was when Feng-Hsiung Hsu entered the frame that things started to take off. Hsu was introduced to chess computing when he was approached by Hans Berliner, an AI faculty member at Carnegie Mellon, where he was a student. He asked him to help with the chess computer he and his team were working on, HiTech. Hsu suspected that the design of HiTech's evaluation chips could have been more efficient and began work on his own custom design.

Upon presenting his work to Berliner, he rejected it, wanting to stay with his original design. Hsu, realising the potential of his design, began working on his own chess-computing project at Carnegie Mellon with the help of a few other graduate students.

"I had the basic blueprint to build the Mother of all Chess Machines, a machine that could defeat the World Champion. In other words, I had a chance to pursue one of the oldest holy grails in computer science and make history. ~ (Feng-Hsiung Hsu, 2002)

Hsu and his team would go on to create ChipTest, which despite having less than a year behind its development and being much smaller than every other computer at the time, was able to give a surprising performance at ACM's Computer Chess Championship (CCC). The CCC was an annual chess tournament where chess computers would compete against each other to find the ultimate chess computer, and Hsu's team would go on to win the event undefeated the following year in 1987 with an overhauled version of ChipTest, ChipTest-M.

**4.1 Deep Blue**

Over the next few years, Hsu's team would develop Deep Thought and continue winning chess-computing tournaments before being picked up by IBM. Now with big funding behind them, they would use their knowledge to develop Deep Blue. In 1996, Deep Blue would face Garry Kasparov, the undisputed world chess champion at the time. Although Kasparov would win with a score of 4-2 across six games played, it would mark the first time he had been beaten by a computer. In 1997, the two would rematch. At that time, Deep Blue had been extensively developed and upgraded to be able to perform 200,000,000 evaluations per second. The competition would be close, and going into the final game, they were tied at 2.5 points each.

Deep Blue was not trained the same way modern chess programs are. It was designed using specialised hardware and a carefully curated set of rules and heuristics. For example, the "opening book" of Deep Blue - which is a database of pre-analysed chess openings, allowing chess algorithms to quickly determine a strong opening - was carefully crafted to be able to respond to moves that the team speculated Kasparov would make. In one notorious moment during the final game, Kasparov made an incredibly risky move that would seem stupid to an avid chess player. "What Garry played in game six on the move seven was a very risky anti-computer chess move." (Hsu, 2002) He put Deep Blue in a position where they could sacrifice their knight for a pawn to gain an advantageous position, believing that the computer would be unable to see the benefit of the move.

"Several of the top commercial chess programs at the time were explicitly prohibited in their opening books from playing the knight sacrifice that Deep Blue played. So, apparently, lots of the commercial chess programmers knew that their programs could not play the sacrificial line." ~ (Hsu, 2002)

In a move that surprised Kasparov, Deep Blue sacrificed their knight for the pawn and won the game and event as a whole. Deep Blue defeated Garry Kasparov over six games, 3.5 points to 2.5. While there was some controversy surrounding some of the circumstances of the event, the bigger picture portrayed a huge milestone in computer science. A machine was able to defeat a world champion in a game of intelligence. Having achieved its goals, Deep Blue would be retired. However, Deep Blue was evidently far from perfect, and the world of chess-computing continued moving forward.

**4.2 Stockfish**

It's only possible to talk about chess computing by also mentioning Stockfish. Stockfish was first released in 2008, and to this day has amassed 13 Top Chess Engine Championship wins and 19 Chess.com Computer Chess Championship wins. It is by far the most successful chess-playing program to date, so much so that it itself is used as a benchmark for other chess-playing programs.

For example, AlphaZero is a computer program developed by Google DeepMind that uses a neural network to take in the current state of the game and determine the best move to make. It was trained to play chess by playing millions of games against itself and refining its strategies from each game. During training, it was periodically matched against Stockfish to test its progress. Within 4 hours of training, it had already passed this benchmark. The two would then fight 100 games against each other, in which AlphaZero would win convincingly with a score of 28 wins and 72 draws.

As far as the game of chess goes, determining a benchmark for AI intelligence is relatively simple, and it is safe to say that AI has reached a level of intelligence exceeding that of a human when it comes to chess. However, as we continue to create more advanced AI programs, benchmarking can become much more varied and complicated.

**5 Contemporary Testing Approaches in Today's Era**

Innovative breakthroughs in areas such as reinforcement learning, computer vision, and natural language processing have emerged due to recent progress in artificial intelligence (AI). As AI systems grow more advanced, assessing their performance becomes increasingly important. Modern AI evaluation is designed to measure the effectiveness, accuracy, and overall efficiency of AI systems, paving the way for further advancements in the field.

Quantitative metrics for AI systems' performance allow researchers to compare them with other systems and monitor their progress over time. Top tech companies, academic institutions, and AI researchers have established the MLPerf collaboration with the purpose of creating standardised performance guidelines for assessing machine learning models, systems, and components. MLPerf offers a collection of strict, representative, and widely-accepted benchmarks that enable impartial performance comparisons across various AI models and hardware configurations. The initiative's aim is to boost transparency and cooperation within the AI community.

MLPerf benchmarks include a diverse range of AI tasks, such as object detection, reinforcement learning, image categorisation, and natural language processing. They are customised to address the particular needs of different AI subdomains. By providing a uniform and extensive framework for evaluating AI performance, MLPerf plays a crucial role in promoting innovation, efficiency, and transparency in the rapidly evolving field of artificial intelligence.

**5.1 Generalisation Capabilities**

The assessment of generalisation ability is a critical component of AI testing. The ability of AI models to generalise these patterns to fresh, unstudied data must be evaluated before they are allowed to learn patterns from huge datasets. The generalisation skills of a model are typically assessed using cross-validation and out-of-sample testing, two popular techniques (Kohavi, 1995). Researchers may assess how successfully an AI model generalises its learning to new data points by dividing the dataset into training and validation sets, which is crucial for real-world applications.

The use of autonomous vehicles is one instance where generalisation abilities are essential in AI. In this setting, artificial intelligence (AI) systems must reliably perceive and respond to a variety of real-world circumstances, including spotting pedestrians, reading traffic signals, and dodging obstacles. Large datasets with photos, videos, and sensor data gathered from a variety of road conditions and environments are often used to train these systems.

An autonomous car might, however, run with situations during real-world deployment that weren't present in the training data, such as fresh road signs, strange weather, or unanticipated obstructions. The AI system needs to be able to successfully and securely generalise its learning from the training data and apply it to these unique scenarios.

By exposing the AI system to fresh, previously unexplored data and assessing its performance, generalisation skills are, in this case, put to the test. A piece of the dataset, for instance, might be kept back during training and utilised as a validation set later. Researchers can evaluate the model's capacity to generalise its learning to new circumstances by contrasting the performance of the AI system on the validation set to its performance on the training set. When presented with new circumstances, a model with great generalisation capabilities will maintain high accuracy and efficiency, assuring the reliable and safe operation of autonomous cars (Geiger et al., 2012).

**5.2 Modern Safety Testing**

Modern AI testing must also take AI safety and robustness into account. As AI systems are used more frequently in crucial applications, it is crucial to ensure their security and resistance to hostile attacks. Artificial intelligence models can be put to the test using adversarial examples, which are deliberately produced inputs intended to trick AI systems. This kind of testing aims to find weaknesses and strengthen the model's resistance to hostile attacks.

Testing for AI robustness and safety is crucial when it comes to facial recognition systems. These systems are used in a variety of contexts, including security, access management, and identity confirmation. It is crucial to ensure the security and reliability of facial recognition systems since they may be the subject of adversarial attacks that aim to trick the AI model or take advantage of its flaws.

The use of staged photographs or tangible items like spectacles or makeup that change a person's look in such a way that the AI system incorrectly recognises them is one type of adversarial attack. This might make it possible for unauthorised people to go around security precautions or accuse innocent people in the process. During the development process, researchers test the AI models against hostile cases to ensure the security and resilience of facial recognition systems. They produce adversarial inputs, such as distorted photos or other misleading adjustments, and then assess the model's capacity to successfully identify the subjects in spite of these manipulations. Researchers can utilise this knowledge to find flaws and strengthen the model's resistance to adversarial attacks if the model fails to detect the proper identity.

Researchers may create more robust models that are less vulnerable to adversarial assaults by using AI safety and robustness testing in facial recognition systems, ensuring improved levels of security and dependability in practical applications.

**5.3 Speech Recognition**

The crucial technology that enables AI systems to translate spoken language into written text is speech recognition, sometimes referred to as Automated Speech Recognition (ASR). The significance of thorough testing to verify ASR systems' performance and dependability has become crucial as their use in fields like transcription services, voice assistants, and accessibility aids grows. The main techniques and metrics for evaluating AI voice recognition systems, as well as pertinent sources, will be covered in this section.

Sets of data are essential for testing voice recognition models. The following are some well-known datasets for developing and testing ASR systems:

- Switchboard Telephone Speech Corpus: The Switchboard dataset consists of roughly 2,400 hours of English-language speech from a variety of speakers with various accents and dialects (Godfrey, Holliman & McDaniel, 1992).

- VoxForge: The open-source project VoxForge (http://www.voxforge.org/) intends to compile speech that has been transcribed from a variety of languages, offering a useful resource for developing and testing ASR systems in many languages.

The Word Error Rate is one of the most used measures used to assess the effectiveness of ASR systems (WER). By contrasting the transcriptions produced by the ASR system with a reference transcription, often written by humans, WER is determined. The statistic calculates the error rate as a % of the total number of words in the reference and accounts for the number of substitutions, insertions, and deletions necessary to match the reference transcription (Sak, Senior & Beaufays, 2014).

The Character Error Rate (CER), which evaluates error rates at the character level rather than the word level, is another metric used to assess ASR systems. For tasks involving sub-word components like phonemes or in languages where word segmentation is less distinct, CER can give a more detailed knowledge of the ASR system's performance (Morris, Maier & Green, 2004).

To make sure that ASR systems are reliable and can generalise effectively over a range of speech situations, such as different accents, dialects, and noise levels, it is crucial to assess them using a variety of datasets and many metrics. Researchers and practitioners can pinpoint areas for improvement and promote the creation of more precise and dependable ASR technology by extensively evaluating AI speech recognition systems.

Some important components of AI testing include interpretability and explainability. Understanding the decision-making process underlying AI predictions is essential for building confidence and guaranteeing ethical AI application as AI systems get more complicated. Researchers are developing a number of methods, such as Local Interpretable Model-agnostic Explanations (LIME) and Shapley Additive Explanations (SHAP), to evaluate the interpretability of AI models. These testing techniques give stakeholders insights into the rationale behind AI systems so they may make better-educated choices about their use.

**Conclusion**

In conclusion, since the field of artificial intelligence's birth, the Turing Test has been crucial in defining it. As a fundamental idea, it has inspired academics to investigate the potential and boundaries of AI systems' ability to emulate human behaviour and intelligence. The Turing Test has evolved beyond its initial intent over time, prompting the creation of several AI benchmarks that currently evaluate various facets of AI performance across numerous domains, including natural language processing, generalisation abilities, and contemporary safety testing.

In essence, the Turing Test will continue to serve as a philosophical reference point for AI researchers, serving as a reminder of their ultimate goal of building robots that can accurately understand and mimic human intelligence. As we advance, it is essential to keep improving and broadening AI benchmarking approaches to handle the difficulties posed by ever-more sophisticated AI systems, making sure that they are not just accurate and efficient but also consistent with human values and ethical issues. The AI research community can strive for continual improvement by carefully assessing AI systems across a variety of aspects, promoting the development of more dependable, trustworthy, and revolutionary AI technologies for the benefit of humanity.

**References**

Turing, A.M., 1996. Intelligent machinery, a heretical theory. Philosophia Mathematica, 4(3), pp.256-260.

Turing, A.M, 1950. I.—COMPUTING MACHINERY AND INTELLIGENCE, Mind, Volume LIX, Issue 236, pp. 433–460.

Pinar Saygin, A., Cicekli, I. & Akman, V, 2000. Turing Test: 50 Years Later. Minds and Machines 10, 463–518.

Turing, A.M. and Copeland, B.J. 2013 The essential Turing: Seminal writings in Computing, Logic, philosophy, artificial intelligence, and artificial life "Plus" the secrets of an enigma. Oxford: Oxford University Press.

French, R.M, 2000. The Turing Test: the first 50 Years, Trends in Cognitive Sciences, Volume 4, Issue 3, pp 115-122.

Weizenbaum, J., 1966. ELIZA—a computer program for the study of natural language communication between man and machine. Communications of the ACM, 9(1), pp.36-45.

Sharma, V., Goyal, M. and Malik, D., 2017. An Intelligent Behaviour Shown by Chatbot System. International Journal of New Technology and Research

Warwick, K. & Shah, H. 2016. Can machines think? A report on Turing test experiments at the Royal Society, Journal of Experimental & Theoretical Artificial Intelligence, 28:6, 989-1007.

Shum, Hy., He, Xd. & Li, D. 2018. From Eliza to XiaoIce: challenges and opportunities with social chatbots. Frontiers Inf Technol Electronic Eng 19, 10–26.

Searle, J. R. (1980). Minds, brains, and programs. Behavioural and Brain Sciences

Moor, J. H. (2001). The status and future of the Turing Test. Minds and Machines.

Legg, S., & Hutter, M. (2007). A collection of definitions of intelligence. Frontiers in Artificial Intelligence and Applications.

Harnad, S. (1991). Other bodies, Other minds: A machine incarnation of an old philosophical problem. Minds and Machines.

Russell, S. J., & Norvig, P. (2020). Artificial Intelligence: A Modern Approach (4th ed.). Pearson.References:

Shannon, C. E. (1950). XXII. Programming a computer for playing chess. The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 41(314), 256–275. https://doi.org/10.1080/14786445008521796

Hsu, F. (2002) Behind Deep Blue: Building the Computer That Defeated the World Chess Champion, Princeton, NJ; Princeton University Press.

Silver D, Hubert T, Schrittwieser J, Antonoglou I, Lai M, Guez A, Lanctot M, Sifre L, Kumaran D, Graepel T, et al. . A general reinforcement learning algorithm that masters chess and shogi and goes through self-play. Science. 2018;362(6419):1140–1144

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI), 1137-1143.

Geiger, A., Lenz, P., Stiller, C., & Urtasun, R. (2012). Vision meets robotics: The KITTI dataset. International Journal of Robotics Research (IJRR), 32(11), 1231-1237.

Godfrey, J. J., Holliman, E. C., & McDaniel, J. (1992). SWITCHBOARD: Telephone speech corpus for research and development. Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 517-520.

Rousseau, A., Deléglise, P., & Esteve, Y. (2014). TED-LIUM: An automatic speech recognition dedicated corpus. Proceedings of the Language Resources and Evaluation Conference (LREC), 125-