LUISS

Degree Program in Data Science and Management

Course of AI Frontiers: Large Language Models

Addressing the Gaps in Human Intelligence Assessment and GenAI Benchmarking: A Comparative Analysis of Evaluation Frameworks and Practical Implementation with SparkBeyond.

Prof. Simone Di Somma

SUPERVISOR

Prof. Giuseppe Francesco Italiano

CO-SUPERVISOR

Joshua Brauner 778931

CANDIDATE

Academic Year 2024/2025

Table of contents:

Chapter 1: Introduction

1.1 Fundamentals of Artificial Intelligence

Artificial Intelligence (AI) refers to the discipline within computer science concerned with developing systems capable of performing functions typically associated with human cognition. Although its definition has evolved over time, it remains as the design of systems capable of executing tasks that, if performed by humans, would require intelligence, replicating it in a computational framework.

According to the U.S National Institute of Standards and Technology (NIST), AI systems are those that "perceive, learn, reason, and act to achieve specific goals" [1]. AI is a broad field that includes some main specific areas like machine learning (ML), deep learning (DL), and generative AI (GenAI), characterized by a very fast evolution in the last few years.

As shown by Fig. 1.1, it could be represented through a multi-layered approach that reflects the real relationships between these technologies, each of which is developed based on the capabilities of the previous one.

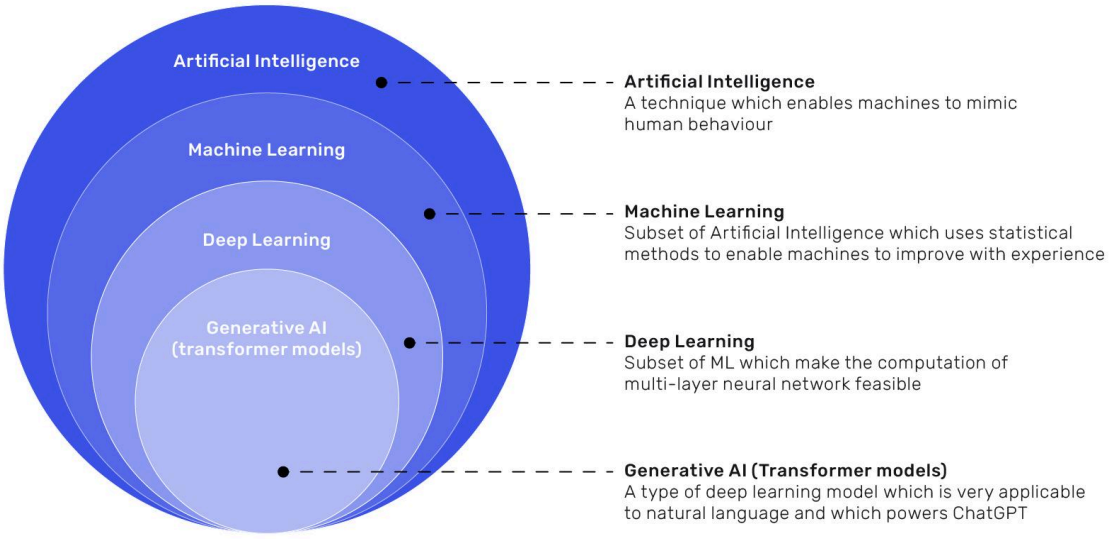


Figure 1.1: Layered structure of the AI field; Artificial Intelligence (AI) represents the broadest domain, and encompasses all forms of machine based intelligence. Within it, Machine Learning (ML) refers to techniques that enable systems to learn from data and recognize patterns. Deep learning (DL) is a specialized subset that leverages multi layered neural networks for complex tasks. At the core, Generative AI and transformer based models represent advanced deep learning architectures at the core of today's most widely used tools.

The origins of AI as a formal academic discipline date back to the 1950s, when researchers began to explore the concept of machine based intelligence in a

structured way. The Dartmouth Conference (1956) is commonly considered as the official founding event that established AI as a research field: at the time, artificial systems operated based on what came to be known as symbolic AI, systems that tried to replicate human reasoning by manually programming knowledge through logical rules and symbolic representations, later referred to as Good Old Fashioned AI (GOFAI). Despite the trials to emulate human thinking, these early AI models had difficulties dealing with the complexity and unpredictability of real world situations, leading to the exploration of alternative approaches.

Not all AI systems are built the same, and it's important to first recognize the different types of intelligence that machines can demonstrate, some created to handle very specific tasks, and others to mimic the broader thinking abilities of humans. This distinction leads to one of the key ways through which AI is categorized: the difference between Narrow AI and General AI.

Narrow AI is designed to perform particular and clearly defined tasks with high accuracy. For instance, voice assistants like Siri or Alexa are built to understand and respond to spoken commands, while recommendation systems on platforms such as Netflix or Amazon suggest content based on users' past behaviour. These systems work within a fixed set of capabilities and cannot perform tasks outside their original purpose. This is why narrow AI is sometimes called "weak AI", not for lack in power, but because it doesn't have real understanding or flexible reasoning, not being able to go beyond what it was trained to do, or apply its knowledge in different contexts.

In contrast, General AI, also known as Artificial General Intelligence (AGI), refers to a still hypothetical form of intelligence that would enable a machine to perform any intellectual task that a human can do, including abstract reasoning and learning across diverse domains, but as of 2025 it remains an open research goal and has not yet been realized in practice.

However, some of today's advanced models, like OpenAI's Large Language Models, can be seen as early steps toward this goal, being able to produce fluent, human like content and handle a wide range of language related tasks. Despite this, they are still not capable of general intelligence, not having real understanding or self awareness in the same way people do.

Much of what today is commonly known as AI is instead more precisely described as Machine Learning (ML), a subfield of AI focused on developing algorithms that enable systems to improve performance on a task through exposure to data, without being explicitly programmed for every scenario [2]. This idea remains central also in modern machine learning: a computational system is said to learn when it improves its performance on a given task as a function of experience.
Unlike traditional rule based software, which operates strictly according to instructions manually encoded by developers, machine learning systems derive decision rules from examples, allowing models to adapt to new data distributions and refine their predictions or classifications without manual reprogramming.
Machine learning techniques have their roots in statistical methods: in fact many foundational models, such as Linear Regression, Logistic Regression, and Decision Trees, are built on principles from traditional statistics. However, these two fields differ completely in their primary objectives: traditional statistics focuses on explanation, understanding the relationship between variables, while machine learning emphasizes prediction, so the ability to make accurate guesses even if that means to sacrifice some level of interpretability.

The learning process is generally distinguished into three main types: Supervised, Unsupervised, and Reinforcement learning.
In supervised learning, models are trained using labeled data, where each input is paired with a known output. Here, the objective is to learn a mapping function that minimizes the error between predicted and actual outputs. Over time, the model adjusts its internal parameters to reduce errors, gradually learning how to match inputs with the correct outputs, until the model reaches a satisfactory level of accuracy.
This technique is mainly used for two types of tasks: classification, where the goal is to assign inputs to predefined categories or labels, and regression in which the objective is to predict a continuous value based on input features. Despite being regression one of the most widely used, its effectiveness depends heavily on the availability of high quality labeled data: as will be later demonstrated with synthetic data usage, without enough well annotated examples the model may struggle to generalize and make accurate predictions.

Unsupervised learning by contrast, involves training models on data that has no labels or predefined outcomes. The objective is not to predict a specific result, but to discover hidden patterns or relationships within the data. Since correct answers are not provided in advance, the algorithm must learn to make sense of the data on its own.
Reinforcement learning differs from both: it's a domain where agents learn to make sequential decisions by interacting with an environment, improving their actions over time based on a system of rewards and penalties known as feedback loop.
The objective here is to discover an optimal policy, a strategy that guides an agent's actions, and that maximizes the total reward it receives over time. Instead of being given correct answers (supervised), or discovering hidden patterns without labels (unsupervised), the agent is learning from outcomes of actions taken, following a dynamic trial and error method. Its great utility is in contexts where the optimal action policy cannot be predefined and must instead be learned through continuous experience and feedback, like when training agentic AI systems.

In this landscape, data plays a fundamental role by providing the ground on which algorithms are trained and evaluated. The quality and quantity of data are directly correlated with the model efficacy: poor quality data can significantly impact performance, no matter how sophisticated the algorithm is. This principle is frequently summarized by the concept garbage in, garbage out, highlighting the dependency of learning systems on reliable input data.
Beyond its role in training, data is also critical for evaluating model performance, with datasets that in statistical modeling are typically divided into three subsets: the training set, the validation set and the test set.

- The training set is used to teach the model, usually making up around 70% of the total data and including many examples where each input is matched with the correct output. By studying these examples, the model learns to identify patterns and relationships relevant to the task.
- After training, the validation set is used to adjust the model's parameters and improve its performance, preventing overfitting when a model becomes too closely tailored to the training data and performs poorly on new, unseen examples. It's used to monitor performance and adjust hyperparameters like

the learning rate or the structure of a neural network, helping to determine which version of the model performs best on data it hasn't seen before.

- Finally, the test set is used to evaluate how well the model can generalize to completely new data. Unlike the previous ones, this is kept separate during the entire training process, providing an objective way to measure the model's real world performance.

1.2 The evaluation gap

The progress toward more intelligent, human like artificial systems increasingly requires defining and evaluating intelligence in a way that enables a complete comparison between two systems and with humans. Yet most advances have focused on task specific assessments, standardized tests and performance metrics that, while convenient for benchmarking, often fail to measure intelligence itself. Without a formal definition the field remains fragmented, with dictionary based notions of intelligence lacking the clarity and measurability needed for scientific foundations. Similarly, evaluation frameworks in artificial systems like the Turing Test, which rely on human judgment, are not suitable for measuring cognitive capabilities in a reproducible way.

Historically, two broad conceptions of intelligence have dominated, contrapposing the ability to achieve specific goals using task specific skills and the capacity to learn new skills and adapt to a new and wide range of environments. These perspectives correspond to established psychological theories that will be discussed in detail in the following sections, such as Cattell's differentiation between crystallized intelligence (knowledge based abilities) and fluid intelligence (adaptive reasoning across unfamiliar situations) [3], and reflect broader philosophical discussions on the nature of cognition.

The first perspective aligns well with systems that encode fixed and predefined abilities for specific tasks, while the second emphasizes learning, adaptability, and the acquisition of new competencies in previously unseen contexts.

Early research in artificial intelligence frequently adopted a modular approach to intelligence, seeing it as a collection of domain specific capabilities with an emphasis on solving predefined problems through programmed rules and structured knowledge bases, leading to evaluation frameworks typically focused on success in narrowly

defined tasks, with performance judged by whether a system could replicate human level behavior in a specific context. While effective in domains requiring formal logic, of which early symbolic reasoning is a very successful example, this approach did not provide systems the capabilities of broad generalization or adaptive behavior across diverse tasks, with learning that was often treated as a peripheral function, secondary to the execution of explicitly encoded knowledge. As a result, systems that achieved high performance did so without displaying genuine intelligence.

An alternative framework defines intelligence as the general ability to acquire new skills and solve unfamiliar problems, emphasizing the importance of learning and adaptability, a starting point for many relevant researchers.
With the rise of machine learning, followed by the development of deep learning, this perspective gained even more popularity: AI systems began to be seen as general purpose learners capable of extracting knowledge from data, rather than relying on explicitly programmed rules.
However, this learning based approach also introduced its own set of assumptions, in particular the idea that intelligence can be compared to the performance on a set of training data, despite still lacking generalization capabilities.
Consequently, evaluation practices have traditionally emphasized task specific performance, relying on benchmarks that quantify accuracy or success within narrow defined problems. While such benchmarks are effective for monitoring progress in constrained settings, they fail to capture broader cognitive abilities such as the reasoning steps through which these models reached a certain solution, or their capacity to generate novel insights beyond the scope of the training data.
As AI systems are increasingly deployed in open ended environments, there is a growing demand for evaluation frameworks that measure robustness, adaptability, and generalization, essential for systems that must operate under uncertainty in dynamic and real world contexts.
The shift from skill to ability evaluation requires a new class of benchmarks that don’t simply reward task performance but assess the capacity to learn, adapt, and generalize across tasks and domains.

According to Chollet [4], so far traditional AI evaluation has relied on four primary methods: human judgment, white box analysis, peer confrontation, and benchmark

testing. Among these, benchmark tests, where systems are evaluated against a fixed set of inputs and expected outputs, have become the most widely used due to their scalability and reproducibility. Despite having played a central role in advancing the field by enabling to compare performances, they also present important limitations since many successful models are strongly tailored to specific datasets, and therefore fail to generalize on real world settings, relying on skill based evaluation even with models that increasingly show steps toward general intelligence.

On the human evaluation side, the development of the Binet Simon scale [5] in the early 20th century marked the beginning of psychometrics, a field totally dedicated to the systematic measurement of cognitive abilities. Shortly after, Spearman [6] observed that individual performance across diverse and apparently unrelated intelligence tests was instead positively correlated, suggesting the existence of a single factor of general intelligence, often referred to as the g factor. Today, psychometrics is a well established subfield of psychology, known for producing highly reproducible findings, allowing modern intelligence assessments to use strict methodological criteria to ensure reliability, validity, standardization, and efforts to minimize cultural and systemic bias.
A fundamental notion in psychometrics is the distinction between broad cognitive abilities and narrow, task specific skills. Theories of intelligence structure that evolved simultaneously to psychometric testing often picture cognitive abilities hierarchically, which parallels the concept of generalization levels observed in artificial intelligence. An ability is an abstract construct, as opposed to a directly measurable and objective property of an individual mind, like a score on a specific test, and broad abilities in AI, which are also constructs, fall into the exact same evaluation problems. Psychometrics approaches the quantification of abilities by using broad batteries of test tasks rather than any single task, and by analysing the result via probabilistic models. A fundamental requirement is that the task should be previously unknown to the test taker, not being able to prepare for it.

This approach is highly relevant also in AI evaluation, since recent work has increasingly adopted multi task evaluation benchmarks designed to test systems across a wide range of tasks, with the goal to move beyond measuring isolated skills, assessing more general capabilities [7]. However, a key limitation of these

benchmarks is that the full set of tasks is often known in advance to developers, allowing systems to be explicitly trained for them, often including the use of task specific pretraining on related data, and other forms of optimization that compromise the ability to evaluate true generalization.
As a result, many of these benchmarks remain vulnerable to overfitting and continue to assess performance on task specific skills rather than underlying cognitive flexibility. Although they give valuable insights into progress within bounded areas, they do not represent a qualitatively different evaluation paradigm from traditional testing: this does not mean that such benchmarks are not useful at all, but only that such static multi task benchmarks are not useful to assess flexibility or generalization.

Bridging the gap between psychometric and AI evaluation requires a shift in focus, from static performance metrics to the measurement of underlying learning and adaptability.
This involves the design of benchmarks that:

- Assess skill acquisition efficiency rather than fixed task performance.
- Check for prior exposure, pretraining, and system specific biases.
- Incorporate new and previously unseen tasks to prevent overfitting.
- Ensure reproducibility and cross system validity.

Such benchmarks should avoid overreliance on crystallized knowledge and instead target the mechanisms that enable systems to acquire new skills in unfamiliar environments, focusing on testing the foundations of generalization.

1.3 Research project overview

In response to the limitations of conventional AI evaluation frameworks highlighted so far, and in collaboration with SparkBeyond, an Israeli company specializing in AI powered data analytics, this research seeks to build a dynamic benchmarking framework able to successfully evaluate an AI agentic system, not considering only its output accuracy or performance, but comprehensively judging the core dimensions that have led to generate a specific output, with the optimal goal of self improvement. This is understood through the assessment of Large Language Models and Agentic AI in their ability to uncover strategic insights from structured data, so

natural language statements that explain the underlying drivers of a specific outcome or performance metric.

Insight discovery, intended as the ability to identify new, non obvious patterns and relationships within complex datasets, remains largely under measured in both human and machine intelligence, despite being a task that is cognitively complex but essential in real world decision making.

This benchmark is based on the generation of synthetic problems, each simulating a realistic analytical context, where for every problem are created structured tabular datasets along with contextual descriptions of a business like scenario. These datasets are carefully constructed to include one or more insights, meaningful relationships between variables that explain changes in a predefined Key Performance Indicator (KPI). Each problem is designed such that the target variable is strongly associated with these embedded insights, enabling precise evaluation of whether a model can identify the correct underlying patterns.

This methodology overcomes many of the limits in traditional benchmarking:

- The use of synthetic data allows for scalable and controlled variation across domains, complexity levels, and data structures, making it possible to test a model's generalization capabilities, as each problem may present previously unseen relationships or table structures.
- Since the insights are expressed in natural language, the benchmark aligns more closely with real world applications where interpretability and communication of findings are critical.
- The open ended nature of the task shifts the focus away from simple classification and toward generating meaningful hypotheses grounded in data.

2. Chapter 2: Defining Intelligence

2.1 Intelligence as a concept

The definition of intelligence remains controversial, being subject to strong debates. Historically, it has been associated with cognitive capacities such as reasoning, abstract thinking, comprehension, and memory, but despite its central role in human life, there's still not a universally accepted definition of what intelligence truly is, and there seem to be almost as many conceptualizations and definitions of it as there are experts to write them.

The main research questions involve whether intelligence is defined by the ability to solve problems, to learn and adapt to an environment, even by its role in driving performance, or maybe as something measured to evaluate it.

This great ambiguity has influenced those questions and distorted the way intelligence has been assessed, particularly through standardized testing, and rather than clarifying the nature of it, these attempts often reveal its complexity and resistance to simple categorization.

Psychology has adopted multiple kinds of approaches to understand intelligence, with psychometric as the most common, both in academic research and applied settings. It emphasizes standardized testing, particularly through instruments such as Intelligence Quotient (IQ) tests, which aim to quantify cognitive performance.

The origins of modern intelligence testing began thanks to the work of French psychologists Binet and Simon in the early 20th century, which between 1903 and 1905 developed the Simon Binet Scale [5], a diagnostic tool intended to identify schoolchildren who required additional instructional support, becoming a significant advancement in educational psychology by introducing the concept of mental age as a driver for cognitive development.

However, from the very beginning Binet has been aware about the potential misuse of his test, repeatedly emphasizing that it was not designed to rank children or to define them by a fixed level of intelligence, but as a practical tool for educators, believing that intelligence could not be measured in the same way as physical traits like height or weight since mental abilities are complex, dynamic and not directly comparable between individuals. Assigning a single, fixed score could unfairly label

a child to a lifetime of being considered less capable, and damage their educational opportunities and life outcomes.

Despite these warnings, the test quickly gained popularity particularly in the United States, where its original intent has been reinterpreted. The American psychologist Goddard [8] translated the Simon Binet Scale in English and promoted the misleading idea that intelligence was a single, fixed, and heritable trait, possible to be measured through a single numerical score that represented an individual's overall cognitive capacity. This led to the creation of the new Stanford Binet Intelligence Scale [9] in 1916, which became the standard intelligence assessment used across American schools and institutions, and introduced the formula for the Intelligence Quotient (IQ), obtained by dividing the mental age by the chronological age, and multiplying by 100.

The term "IQ" itself was coined by the German psychologist William Stern [10], who introduced the idea of dividing mental age by chronological age to create this quotient. Nowadays, following the guidelines of the American Psychiatric Association [11] , a score below 70 often indicates intellectual disability, while scores above 130 are typically considered indicative of being exceptionally talented. An average score is 100, with most people scoring between 85 and 115.

In the decades that followed, growing evidence challenged the assumptions behind IQ testing: particularly after the horrors of World War II and with the rise of the Civil Rights Movement, a new wave of research began focusing on the environmental influences on intelligence. One of the most notable findings is the Flynn effect [12, 13], which describes the observed improvements in standardized IQ scores across successive generations throughout the 20th century, that being observed in both fluid and crystallized intelligence, has progressed at a pace too rapid to be attributed to genetic evolution.

Since IQ tests are periodically recalibrated to preserve a mean score of 100, researchers observed that newer generations consistently outperformed earlier ones on older test versions. This pattern suggests that environmental factors associated with modernization play a substantial role in shaping cognitive abilities, underscoring that intelligence is not fixed but strongly influenced by social conditions.

Today, these IQ tests continue to be used for specific purposes such as identifying intellectual disabilities and determining eligibility for educational support, although there are now better methods to detect potential biases in questions and scoring. However, most psychologists agree that reducing an individual's intelligence to a single numerical score is misleading, reductive and ethically problematic, failing to capture the full spectrum of human potential, like creativity, emotional depth, adaptability, and more.

Theories of multiple intelligences have played a transformative role in reshaping how cognitive diversity is interpreted, and at the same time, critiques of IQ testing have grown stronger.

Gardner's theories [14] defined intelligence not as a single and unitary ability, but rather a constellation of building blocks, each representing a unique way in which individuals can excel and demonstrate intellectual capability. This perspective challenged the dominance of psychometric approaches and broadened the lens through which educators and psychologists view human potential.

Goleman's definition of emotional intelligence [15] includes the capacity to recognize and manage emotions, both one's own and of others, highlighting dimensions previously not considered in traditional models. Five core components have been identified: self awareness, self regulation, motivation, empathy, and social skills, not only influencing psychology but also transforming educational practices and training, linking emotional skills to personal and professional success.

Taleb [16] has described IQ measurement as a form of pseudoscience, noting its weak correlation with real world outcomes like creativity or, to cite something focal in nowadays society, even income. He recently argued that IQ is more effective at detecting cognitive deficits than predicting excellence, echoing Binet's original intention, criticizing also the statistical assumptions behind IQ tests, particularly their reliance on linear models to capture what is in reality a nonlinear and context dependent phenomenon.

This ongoing debate has implications far beyond human cognition, with the difficulty of defining and measuring intelligence is strongly mirrored in the study of artificial intelligence too. Just as early psychologists struggled to define a complete description of human intelligence, AI researchers today have similar challenges: artificial systems often demonstrate notable skills in specialized tasks but struggle

with generalization and adaptability, qualities that lie at the heart of what makes intelligence so hard to define.

2.2 Definition of artificial intelligence

I. Early models

The idea of creating artificial entities has long captured human imagination, and although the term artificial intelligence may seem like a product of the digital age, this ambition has deeper origins.

Intelligence has been mostly studied in humans, but signs of it have been recognized also in animals, plants, and more recently in machines. AI is the latest and most effective effort to recreate this complex capability using computer systems, and this link between natural and artificial intelligence is focal, not just for technological progress, but also for understanding the nature of intelligence itself.

AI already plays a central role in everyday life, influencing how people interact and make choices, but to avoid misunderstandings and inflated hopes, it's essential to explore where it comes from and what core ideas it's based on.

Although often associated with science fiction or futuristic ideas, the theoretical roots are dated to the mid 20th century, with the introduction of the first computers designed for basic calculations. Since then, its development followed alternating cycles of optimism and doubts: after initial optimism came periods of limited progress and declining support, until the 21st century when interest has grown again, driven by improvements in computational power and the rise of deep learning techniques.

The progress of AI reflects a continuous effort to understand and replicate cognitive processes through computational systems. A major milestone was achieved when McCulloch and Pitts [17] introduced the first formal model of an artificial neuron: known as the McCulloch Pitts (MP) neuron, it processed binary inputs and produced a binary output, so either 0 or 1, based on a threshold activation function, allowing to mimic decision making through logical operations. Even though it was quite simple, the MP neuron could already perform basic logical tasks, demonstrating the potential of using interconnected artificial neurons for computations.

Built on this foundation, one of the earliest trainable neural architectures was the Perceptron, developed by Frank Rosenblatt [19]. Unlike the MP neuron, the

Perceptron could process real inputs and adapt its internal weights through a learning rule aimed to reduce classification errors, making it a more flexible and adaptive model than the previous one.
In the following decades, experiments with digital computers showed the potential of artificial systems in solving simple problem tasks generating huge enthusiasm, but soon facing significant technical challenges.
During the 1960s and 1970s, progress in AI research had considerable limitations especially due to insufficient computational power and a lack of large scale datasets, with optimistic predictions that further amplified the field's challenges.
The inability of early models to solve non linearly separable problems highlighted the need for multi layer network architectures, and the introduction of multi layer perceptrons, along with the formalization of the backpropagation algorithm, provided an effective solution for training deep networks [20].
In parallel, attention shifted toward Artificial Neural Networks (ANNs), inspired by the structure and function of the human brain: these networks, made of layers of artificial neurons, were designed to imitate the brain's ability to learn from data and recognize patterns. However, progress slowed once again due to limitations mentioned above, leading to temporary decline in interest in the field despite reaching a turning point with the rediscovery and formalization of the backpropagation algorithm.

The late 1990s and early 2000s marked a period of rapid global growth in computing and internet use, with people around the world starting to produce massive amounts of data and computer processing power advancing at an extraordinary rate, leading to the development of scaling laws for neural networks models. In fact when increasing computing power, dataset size and parameters, the result is an exponentially decrease in test loss and smooth improvement in model's performance (Fig. 2.1). To reach the optimal performance, all three factors must be scaled up in tandem.

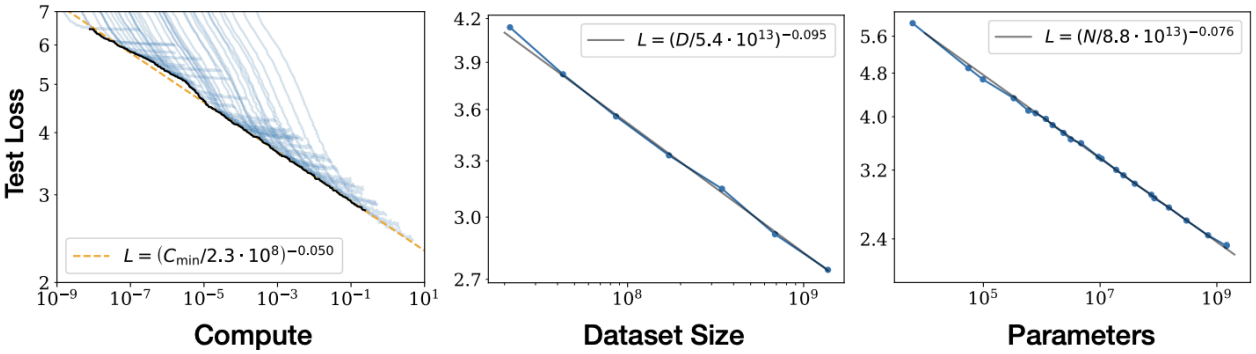


Figure 2.1: Scaling law for Neural Models: performance improves by increasing the model size, dataset size, and computing resources used for training [22].

Thanks to this convergence of several key advancements, including the rise of effective deep learning architectures and significant improvements in hardware, neural networks saw a major resurgence in the late 2000s and early 2010s, making training large models much faster and more efficient.

The current one represents an especially dynamic and promising era in the development of AI, driven by the convergence of advanced hardware, unprecedented access to large scale data, and increasingly effective training techniques, and alongside these technical factors, growing public interest and substantial global investment have further accelerated progress. Together, these conditions are enabling the creation of more powerful models and the widespread accessibility of AI tools, leading to rapid innovation and deployment of its real potential across sectors.

II. The deep learning era

The real jump in AI capabilities came with the rise of deep learning [21], which has transformed many fields by offering powerful tools to analyze complex data, relying on training deep neural networks (DNNs) with many layers to recognize detailed patterns in data.

These networks are inspired by how the brain works and are built from units called neurons, each processing input data and passing the result to the next layer, forming a layered structure that allows the network to learn from experience in a way that recalls human learning. Although influenced by this idea of neurons and synapses in the brain, deep learning models are mathematical systems designed for computational speed and accuracy, not for replicating biological processes.

As already discussed, before deep learning, machine learning models faced two main challenges: they required manual feature engineering with experts that had to design the right inputs for the model, and they struggled to capture complex, non linear patterns in the data.
Deep learning overcame both limitations by allowing models to automatically learn useful features directly from raw data. In fact, one of the key strengths of deep learning is its ability to extract relevant features without human intervention, being also highly effective at handling large, high dimensional datasets. Thanks to their flexibility, deep learning models can be used in many types of tasks, but they require a significant amount of data to perform well and produce reliable results.

Among the most influential deep learning architectures of this period, recurrent neural networks (RNNs) were specifically designed to process sequential data, where the order and temporal dependencies between elements are crucial.
Unlike traditional feedforward networks, RNNs maintain an internal hidden state, or memory, that is updated at each step in the sequence, allowing them to capture temporal patterns and making them well suited for tasks like language modeling and time series prediction, where past information influences the interpretation of future inputs.
RNNs are trained using a method called Backpropagation Through Time (BPTT), which is an extension of the already discussed regular backpropagation algorithm used in feed forward networks. It works by unfolding the network over time, treating each time step as a layer, and in the forward pass the entire input sequence is processed, with the error being computed at the final output layer.
This error is then propagated backward through time, from the last time step to the first, allowing the model to adjust its parameters based on the full sequence.

However, they struggle significantly when trying to learn long term dependencies, due to what is known as the vanishing and exploding gradient problems. In the case of vanishing gradients, the gradient values diminish exponentially as they are propagated backward through time, eventually becoming too small for the model to learn effectively. Contrarily, exploding gradients occur when gradients grow uncontrollably large, leading to unstable training and convergence failures.

In addition, while they are very effective at processing sequential data, they struggle to handle large amounts of sequential data because they process inputs one at a time, making it difficult for them to efficiently process long sequences.
For example, in a simple language model trying to predict the next word in a short sentence, the model can easily rely on the immediate context, and RNNs generally perform well. In such cases, the distance between important information and where it is needed is short, so the model can learn the pattern effectively. There are cases in which more context is needed, and RNNs could become very ineffective when the gap between relevant information and where it's needed becomes very large. The problem is that information is passed forward one step at a time, and the longer the sequence, the more likely that key information gets lost along the way.

The Transformer architectures partially solve those issues. Introduced in the paper "Attention is all you need" [24], this model marked a significant advancement in the field of deep learning and natural language processing by eliminating the need for recurrence, and instead relying entirely on attention mechanisms within sequences. This has been a central innovation, particularly the later developed self attention mechanism and its multi headed extension, enabling it to weigh the importance of different elements in the input sequence relative to each other. This allows for simultaneous processing of all tokens in the sequence, contrarily to recurrent neural networks which process inputs sequentially and are therefore less efficient in capturing long range dependencies.
Formally, self attention mechanism is defined as:

$Attention(Q, K, V) = Softmax(\frac{QK^T}{\sqrt{d_k}})V$, where Q, K, V are Query, Key, and Value matrices, and d_k is the key dimensionality.

Its multi headed extension applies several independent attention mechanisms in parallel, with each head focusing on different parts or aspects of the input, allowing the model to generate richer and more comprehensive contextual representations. The outputs of the individual heads are then concatenated and linearly transformed, resulting in an integrated representation of the sequence from multiple perspectives. By effectively addressing the limitations of earlier models, such as the difficulties in learning long range dependencies and the lack of parallelism during training, this

architecture has become the standard approach for training large scale language models.

III. Generative AI and Large Language Models

This development process gave rise to the field of Generative Artificial Intelligence (GenAI), AI systems that can create original content, such as text, images, code, audio, or video, by learning patterns from large datasets. Unlike traditional AI models, which are usually discriminative models designed to classify or recognize input data, generative models can produce new data that mimics the structure and style of the data they were trained on.

The field of generative modeling gained significant attention following the development of Generative Adversarial Networks (GANs) by Ian Goodfellow et al. [25]. GANs use a two network architecture consisting of a generator, which produces synthetic data, and a discriminator, which evaluates the authenticity of the data by distinguishing between real and generated instances, resulting in highly realistic outputs particularly in image generation tasks and encouraging the generator to produce increasingly realistic outputs.

In natural language processing, language models are systems designed to process, understand and then generate human like language, learning patterns and structures from large datasets that enable them to produce coherent and contextually relevant text.

Although the terms language models and large language models are often used interchangeably, LLMs specifically refer to neural language models characterized by a very large number of trainable parameters, typically hundreds of millions or even more.

Most LLMs are trained using an autoregressive modeling framework, in which the model learns to predict the next token in a sequence based on the previous ones, allowing the model to learn syntactic and semantic dependencies, and making them very effective for tasks requiring to capture complex language patterns and dependencies.

Formally, an autoregressive language model estimates the conditional probability of a token w_{t+1} given the sequence of preceding tokens $(w_1, w_2, \ldots, w_t)$:

$$P(w_{t+1}|w_1, w_2, \ldots, w_t)$$

During generation, LLMs use decoding strategies to determine the next token at each step. It selects the token with the highest probability at each step, and randomly samples it from the predicted probability distribution, introducing variability that makes the output unpredictable and diverse as human language.

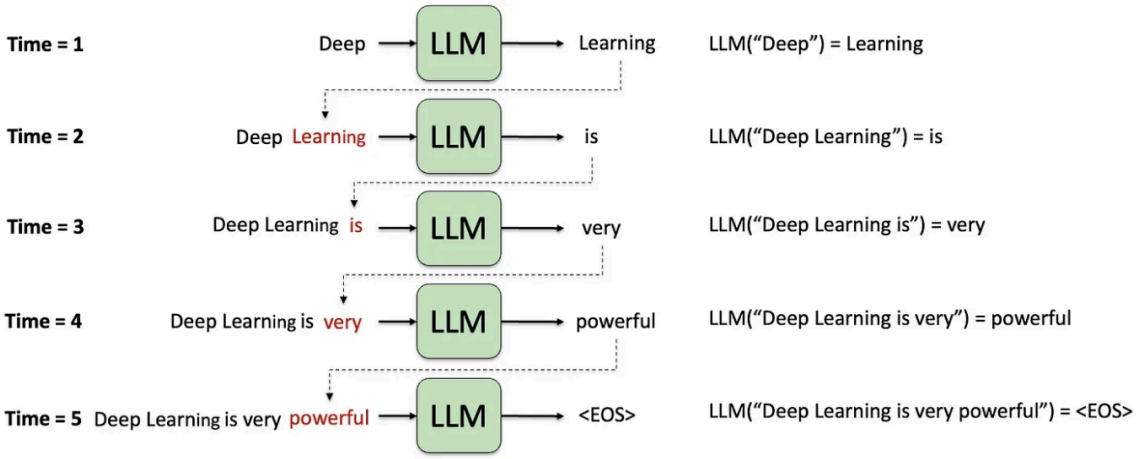


Figure 2.2: Auto-regressive token generation in a Large Language Model (LLM). At each time step, the model predicts the next token based on all previously generated tokens, continuing until it outputs an explicit end of sequence (<EOS>) token.

As shown by Fig. 2.2, this autoregressive nature of LLMs allows them to generate text sequentially one token per time, using previously generated words as context. Starting from an initial prompt or token, they iteratively predict the next word until a complete sequence is formed or a predefined stopping condition is met, allowing LLMs to produce coherent and contextually relevant text, which make them powerful tools for tasks that require linguistic flexibility and adaptability, much like humans do.

The introduction of this transformer architecture marked the beginning of a new era in natural language processing, characterized by a rise of pretrained models and an unprecedented focus on scalability.
This period has seen the emergence of two influential groups of models: BERT and GPT.

In 2018, Google introduced BERT (Bidirectional Encoder Representations from Transformers) [27], a model that used the transformer's encoder architecture to achieve state of the art performance across a wide range of natural language

processing tasks. Unlike earlier models that processed text in a single direction, either left to right or right to left, BERT uses a bidirectional training approach, allowing context to be captured simultaneously from both directions, thanks to which it has significantly improved performance on various language understanding tasks like text classification, sentiment analysis, and many more.
Instead of predicting the next word in a sequence, BERT was trained to predict randomly masked words within a sentence, forcing the model to consider both the words before and after the masked token, and encouraging a better understanding of the full sentence context.
In addition, it has also been trained to determine whether one sentence naturally follows another in a text, helping the model to learn relationships between sentences, and making it especially effective in tasks like question answering or semantic understanding [28].

a. The GPT series revolution

Between 2018 and 2020, advances in computational infrastructure and the availability of large scale training data enabled the development of significantly larger language models. In this period, model architectures were scaled from hundreds of millions to billions of parameters, leading to measurable improvements in generalization across a wide range of tasks, and this scaling effect demonstrated that increasing model size, dataset volume, and training duration usually leads to enhanced performance across many natural language benchmarks.

While BERT focused on understanding bidirectional context, the Generative pretrained transformer (GPT) series introduced by OpenAI followed a different strategy, prioritizing generative capabilities through autoregressive pretraining. Differently from BERT, which is based on the transformer encoder, the first GPT model introduced a unidirectional autoregressive training objective with approximately 117 million parameters, where each token was predicted based only on the preceding sequence. This allows it to capture sequential patterns in language, making it strong in generative tasks.

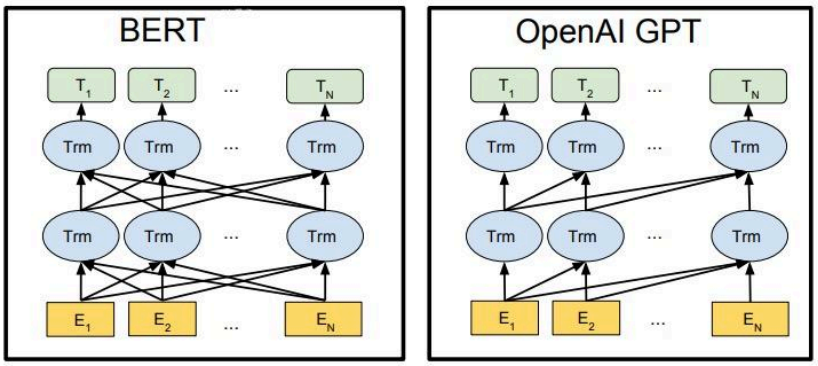


Figure 2.3: Differences in pretraining model architectures. BERT uses a bidirectional Transformer, OpenAI GPT uses a left to right Transformer. Source: Devlin et al. (2018).

The release of GPT-2 [29] represented a significant advancement over its predecessor. Built on the original GPT architecture, which had already demonstrated that pretraining on large text data could significantly improve task performance, it maintained core transformer decoder only architecture from Vaswani et al. [24] and employed a multilayer autoregressive model that predicts each token based on its preceding context. It demonstrated strong performance on zero shot learning, predicting the answer given only a natural language description of the task (prompt), without examples.

GPT-2 was the first LLM to exhibit elements of commonsense reasoning, capable of performing a range of natural language processing tasks, including question answering and reading comprehension, achieving state of the art performance on seven out of eight language modeling benchmarks (Fig. 2.4).

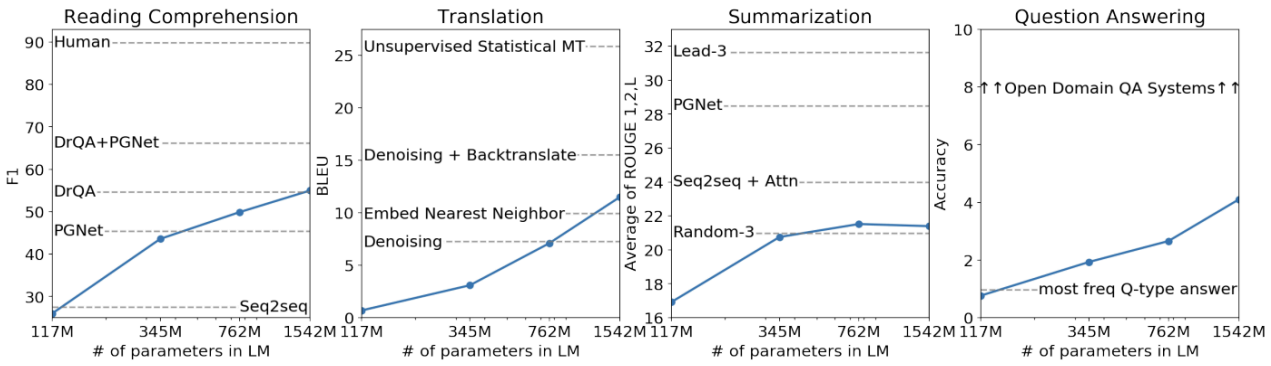


Figure 2.4: Zero shot task performance of WebText LMs Benchmarks as a function of model size on many NLP tasks [29].

GPT-3 [22, 26] advanced generative AI to the next level, becoming the largest language model of its time, but while being trained on a massive and diverse corpus of text despite keeping the same transformer based architecture as the previous one, it introduced substantial improvements particularly in scale and performance.
A major innovation introduced was its ability to perform tasks without fine tuning, relying instead only on prompt based learning, a capability demonstrated through three approaches:

- Zero shot learning, where the model generates answers based solely on a natural language description of the task (the prompt), without examples.
- One shot learning, where the model is given a single demonstration along with the task description.
- Few shot learning, where the model receives a small number of demonstrations (typically fewer than 100) but no weight updates, enabling it to generalize effectively from minimal supervision.

Its ability to generalize across a wide range of domains demonstrated the emergent capabilities of large scale pretrained models, and its capacity to produce human like language set a new benchmark for generative AI, despite still raising strong ethical concerns about bias, hallucinations, misinformation and misuse. In fact, despite these impressive capabilities, pretrained LLMs often show important limitations by generating incorrect outputs, and frequently struggling with logical reasoning over long contexts, since an LLM form of reasoning that predicts the next token based on statistical patterns in large datasets is different from the one of a human.
Therefore, pretraining alone does not ensure reliable reasoning or alignment with human intent, and post training techniques [23] emerged as crucial steps to address these gaps, trying to align the model's outputs with human values and task specific requirements.
Two of the most widely adopted techniques are:

- Supervised Fine Tuning (SFT), which involves training the model on carefully curated datasets to improve accuracy and ensure compliance with guidelines.
- Instruction Alignment, where models are trained to better understand and follow user prompts, effectively reducing cases of hallucinations, improving the model's ability to follow instructions, and aligning the model's behavior more closely with human values.

However, supervised fine tuning alone has limitations in both scalability and performance, being also very labor intensive. Simply replicating human behavior does not ensure that the model will exceed human examples or generalize effectively to unknown tasks.

To address this challenge, OpenAI introduced Reinforcement Learning from human feedback (RLFH) [31, 32, 33] as part of the training pipeline for instruction following models. Unlike SFT, which relies on manually written outputs, RLHF involves ranking multiple model generated outputs based on their quality, simplifying data labeling while improving efficiency and alignment with user preferences. This involves two main stages: training a reward model, which learns from human ranked outputs to estimate which responses are preferred, and fine tuning the LLM using reinforcement learning specifically with the Proximal Policy Optimization (PPO) algorithm, that iteratively updates it based on the reward model's feedback, encouraging the generation of outputs that better reflect human expectations and values. This combination helps the model not only to accurately follow instructions, but also to adapt to new tasks and improve continuously. Overall, models trained with post training alignment techniques demonstrate improved alignment and reduced hallucinations compared to base GPT and purely prompted models (Fig. 2.5).

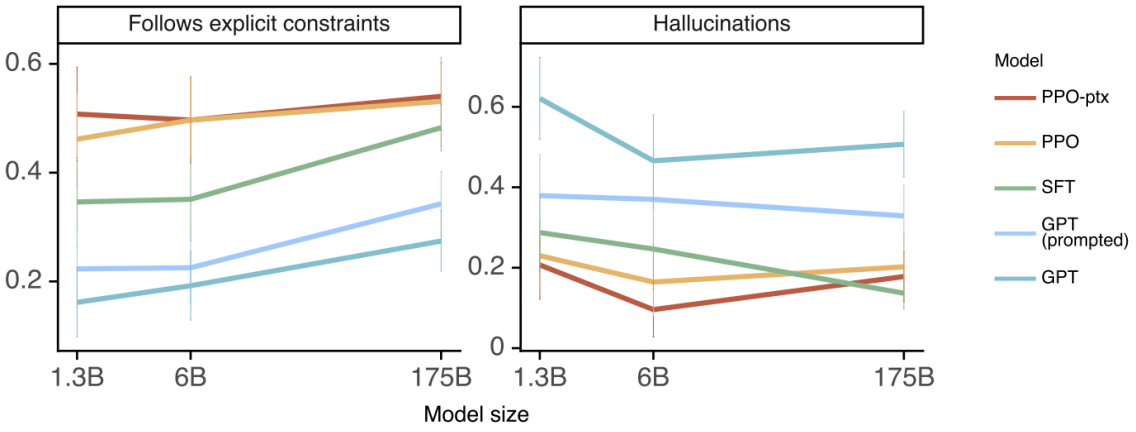


Figure 2.5: Impact of model size and post training techniques on constraint adherence and hallucination rates in LM [33].

With the release of ChatGPT by OpenAI (2022), human machine interaction has experienced a complete transformation: this revolutionary conversational AI model optimized for natural language dialogues was trained on large scale conversational

data and fine tuned using RLHF, contributing to generate outputs that were more helpful, harmless, and honest as much as possible.
The field has recently undergone further expansion with the emergence of multimodal large language models such as GPT-4 and GPT-4o, which mark a new era in AI by integrating multiple data types, including text, images, audio, and video, within a single unified system. This has significantly extended the capabilities of traditional language models and revolutionized a wide range of sectors by enabling innovative applications across previously unconnected domains.

b. Large reasoning models

By 2023, the research in AI development began to prioritize models with enhanced reasoning capabilities, shifting beyond basic pattern recognition and toward architectures designed for structured, multi step problem solving.
This transition was influenced by cognitive psychology's dual process theory, which distinguishes between System 1, meant as fast and intuitive, and System 2, characterized by slow and analytical reasoning. While earlier models such as GPT-3 and GPT-4 have demonstrated strong performance on System 1's tasks like fluent text generation, they instead showed limitations in handling more complex reasoning and problem solving tasks typically associated with System 2 [34].

The introduction of reasoning models, starting with OpenAI's o1-preview (2024), marked the emergence of a new class of reasoning focused models that employ Long Chains of Thought (Long CoT), meaning internal reasoning steps that allow the model to decompose problems, critique its own solutions, and explore alternatives before arriving at a final answer. These CoT are typically hidden from users, who only see a summary output.
In addition to Long CoT, these models incorporate techniques for improving reasoning quality and managing computational efficiency. Although early reasoning models like o1-preview were sometimes less capable in general language tasks compared to standard LLMs, they outperformed them in reasoning tasks, often even reaching human level performance. In fact, o1-preview has shown higher performance than GPT-4o on tasks involving advanced mathematics, code synthesis, and PhD level academic evaluations, as reported in OpenAI's technical documentations.

Large language models with advanced reasoning capabilities typically require substantial computational resources for both training and inference, something that combined with the closed source nature of many state of the art systems, restricts the access to high performance AI.

In response, the increasing availability of open weight and open source LLMs is playing a crucial role in democratizing access to advanced AI technologies: with open weight models the parameters are publicly accessible with minimal restrictions, enabling fine tuning and adaptation to each personal application although the training data and underlying architecture remain proprietary. They are particularly useful for rapid deployment, with Meta AI's LLaMA series representing a prominent example. By contrast, open source models release both the code and the architecture, offering full transparency and customizability for the goal of collaborative innovation.

The open source AI ecosystem is currently facing a particularly dynamic phase, fueled by advances in alignment techniques that drive the release of an increasing number of high performing open weight models, narrowing the gap with closed source alternatives. Rather than relying only on frozen, general purpose systems such as GPT-4o, alignment methods now make it possible to fine tune pen weight models for specialized use cases, leading to improved performance and customizability at lower cost (Fig. 2.6).

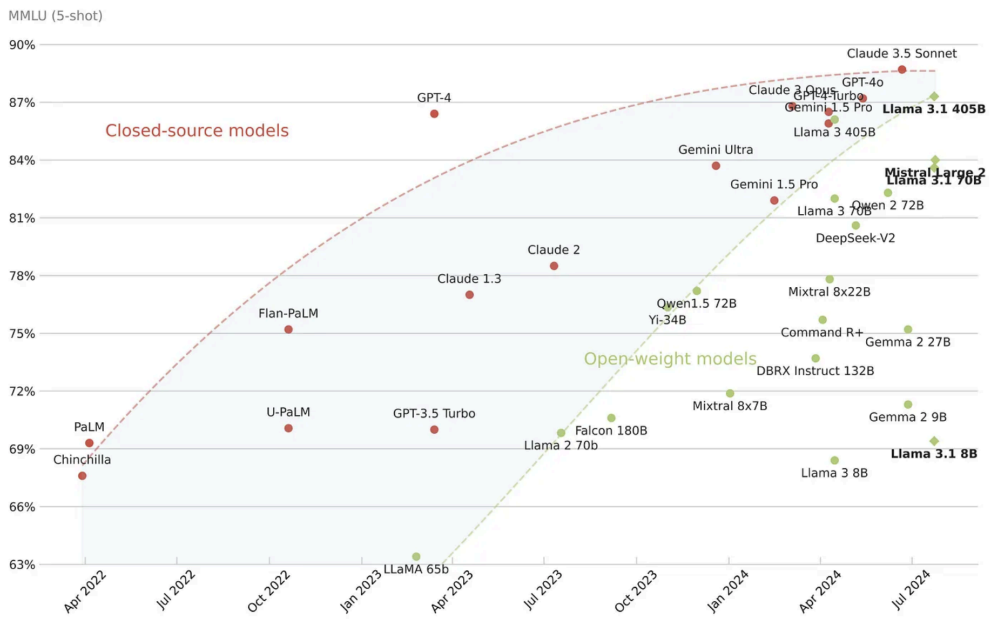


Figure 2.6: Closed source vs Open weight models. For the first time, with LLama an open weight model has significantly filled the performance gap with closed source models.

Among these new open weight models, DeepSeek-V3 established itself as a high performing one but with significantly lower development costs compared to OpenAI's ChatGPT, estimated at around $5.6 million, a fraction than typical investments made by Western tech firms. The model architecture is based on a mixture of experts (MoE) approach, with 671 billion total parameters of which 37 billion are selectively activated based on the input, dividing the model into specialized components that are reducing computational load while maintaining high performance.
Following its release, DeepSeek model series caused substantial disruptions in the technology sector. Market analysts attributed a temporary decline in market capitalization, estimated at up to $1 trillion and a 13% drop in Nvidia’s pre market stock price to the model’s disruptive potential. A key driver of this impact was its competitive pricing: at $2.19 per million output tokens, being up to 20 to 50 times cheaper than proprietary alternatives like OpenAI’s o1 model.
These game changing developments are not only technically impressive but also strategically revolutionary, democratizing access to advanced LLMs and fostering a more competitive ecosystem, with its affordability and open availability that are expected to accelerate adoption across a wide range of industries and applications in the next few years.

Recently, OpenAI has released the new version of its GPT models, named ChatGPT-5. This new model integrates the strengths of GPT-4 and the experimental o3 model, delivering strong performance across tasks typically assigned to neural networks. Its development introduces three principal enhancements: first, with improved reasoning and decision making have been introduced advancements in logical reasoning and multi step problem solving, making the model more effective for complex applications like strategic planning and scientific research.
Second, whereas GPT-4 enables both text and image inputs, but with limits across multimodal reasoning, GPT-5 provides broader and more flexible integration through multiple data sources, becoming more suitable for a wider range of real world use cases. Third, its capability for extended conversational memory enables longer interactions without losing coherence, improving information management and giving better user experience.

As LLMs continue to evolve, they are transforming into highly versatile, multimodal systems capable not only of generating text, but also interpreting and interacting a varied range of tasks, domains and data types, and as already showed this progression has been enabled by improvements in computational efficiency and data accessibility, as well as in model architectures, driving AI toward a future that is more accessible and aligned with real world use cases.

However, the growing general purpose nature and autonomous capabilities of these models present new challenges, especially for evaluation: conventional benchmarks, which often assess isolated tasks or narrow metrics, are insufficient for capturing the multifaceted cognitive functions that modern LLMs now exhibit. As they increasingly interact with users through dynamic environments, it becomes essential to evaluate not only their output quality, but also the processes by which they reach those outputs, including their alignment with human intent, contextual understanding, reliability, and adherence to ethical constraints.

This trajectory reflects a broader shift in AI evaluation paradigms, from measuring performance on static tasks to assessing dynamic and interactive forms of intelligence. In this context, the development of new generation evaluation frameworks is a critical priority, being necessary not only to characterize the full range of model capabilities, but also to ensure that these systems are deployed safely, responsibly, and in alignment with their intended roles in increasingly complex professional and social environments.

2.3 The problem in emulating human intelligence

Replicating human cognitive abilities has been one of the most ambitious goals in AI development, but despite the considerable progress already seen, achieving full human like intelligence in artificial systems remains an unresolved and complex challenge, not only due to technical limitations but also because human intelligence itself is not yet fully understood.

While advances in neuroscience have given detailed indications into the structural and functional organization of the brain, particularly associated with the neocortex believed to be responsible for cognitive functions, there is still a lack in a complete understanding of how the complex and adaptive behavior that defines humans has origin from neural activity.

This gap in knowledge has direct implications for AI, because without a clear understanding it's being difficult to design artificial systems that can replicate it fully. In fact, artificial systems today operate by simulating only specific functions of the brain, and often in narrow domains, but yet without demonstrating the broad generalizable intelligence that humans exhibit across diverse and often uncertain situations, and therefore even the most advanced neural networks remain today a simplification of biological reality.

Another reason why artificial systems struggle when emulating full human intelligence is indeed that it lacks all the other components that are built upon, like emotion and consciousness, since human thinking is not purely rational and emotions play a crucial role in human way of thinking. Current AI models do not possess such capabilities, processing inputs and producing outputs based on statistical correlations, without any form of self awareness or internal representational states similar to human consciousness.
Embodiment is another core function. Theories of embodied cognition [35, 36] suggest in fact that intelligence is grounded in sensorimotor experiences, where learning is facilitated through interaction with the physical environment, and that is in fact not possible with artificial systems at the current state.
That said, a new wave of agentic AI systems is beginning to push the boundaries of what artificial intelligence can do. Unlike static models that only generate outputs in response to prompts, agentic systems are designed to operate more like autonomous entities that pursue specific goals by interacting with their environment and refining their strategies based on feedback. However, while these agents show greater adaptability, they are still limited by their initial configuration, in terms of training data and goal definitions, and generally speaking agents do not formulate new objectives unless they are explicitly programmed to do so.
Adaptation remains a limited ability, which usually emerges only in strictly defined environments or through specific reward based training.

Intelligence itself is far from perfection and this is an often overlooked issue, being subject to a wide range of cognitive biases and emotional distortions. Being trained on human generated data, artificial systems might also lead to systematic errors, such as confirmation bias, overconfidence, or stereotype based reasoning, inheriting these flaws and often amplifying them. Numerous studies have shown that machine

learning models can replicate gender, racial or socioeconomic biases present in their training data, and this is particularly worrying when such systems are deployed in high risk areas such as healthcare or recruitment.
As AI continues to evolve, it will be essential to recognize these limitations, not only to avoid overestimating the capabilities of machines, but also to ensure that those systems are aligned with human values, resistant to abuse and capable of completion, rather than simply imitating the whole spectrum of human intelligence.

3. Chapter 3: Measuring Intelligence

A similar reductionism can be observed in the evaluation of both human and artificial intelligence. Contemporary AI benchmarks largely focus on performance in narrow tasks, such as question answering or predicting the next token in text generation, prioritizing speed and accuracy within constrained settings. This focus often overlooks broader cognitive capacities, including adaptability to novel situations and flexible reasoning.
As a result, AI systems may appear strong under controlled conditions, yet remaining unreliable when applied outside predefined scenarios. The problem is not only that current metrics are too narrow, but they also shape the definition of intelligence in a way that leaves out important cognitive qualities.
In both humans and machines, intelligence is often reduced to matching patterns and producing optimized outputs, rather than being seen as a dynamic and context aware process, as required in real environments.
This narrow view strongly influences the design of AI systems itself, leading to models that are highly effective at specific tasks but unable to generalize or engage in ethical reasoning, reinforcing the same limitations seen in older and more restricted theories of human cognition.

3.1 Measuring human intelligence

Despite over a century of research, intelligence remains a complex and debated concept. In fields such as psychology and cognitive science, there is no unified agreement on a singular definition of intelligence, whether as the ability to reason, adapt, think creatively, understand emotions, solve problems, or some combination of all. This conceptual ambiguity has led to different methods for its measurement and interpretation, especially in humans.

Historically, one of the most influential approaches to assess intelligence has been the intelligence quotient (IQ) test, trying to quantify general cognitive ability using tasks that assess reasoning, memory, processing speed, and verbal comprehension, which through their standardization and statistical consistency offered the advantage for comparisons between individuals and groups. However, IQ tests have also faced significant criticism for providing a narrow view of intelligence, reducing it to performance on abstract and decontextualized tasks that may not reflect the complex realities of everyday life, often failing to capture abilities that can play a crucial role but are not reflected in traditional scores, such as emotional awareness and creativity. Additionally, the socio historical context in which these tests were developed has further contributed to generate validity concerns, since early intelligence assessments were strongly influenced by eugenic ideologies and reflected the biases of predominantly western, white, monolingual populations.

These origins have had several implications: as noted by Ortiz and Cehelyk [37], standardized intelligence tests often disadvantage individuals with different backgrounds from the original tested population, highlighting the need for more inclusive and culturally responsive assessment tools that account for diverse linguistic and cultural experiences, particularly in a globalized world where diversity is often the norm, rather than the exception.

Recent research has tried to move beyond narrow and biased foundations by developing more multidimensional and context aware frameworks, ensuring greater equity in assessment. This shift involves both a theoretical redefinition of intelligence and the design of new tools better suited to capturing its complexity.

A major development in this direction has been the move beyond Spearman's g factor theory [6], for which all cognitive abilities derive from a single underlying general intelligence factor.

Beginning in the 1990s, different psychometric models were developed that conceptualize intelligence as a collection of multiple abilities: a particularly influential starting point for these approaches is the Cattell Horn model [38], which has become one of the most widely supported and coherent frameworks for describing the structure of cognitive abilities.

This perspective considers intelligence as a constellation of diverse attributes, shaped by both genetic and environmental factors and evolving throughout the lifetime, rather than as a single and unitary construct.

The central distinction made by the model is recalling to the concepts of fluid intelligence (Gf) and crystallized intelligence (Gc) [3] already used in explaining the Flynn Effect:

- Fluid intelligence (Gf) refers to the capacity to reason, identify patterns, solve novel problems, and adapt flexibly to new situations, largely independent of prior learning and cultural context, that rely on basic processing mechanisms such as working memory and abstract reasoning.
- Crystallized intelligence (Gc) on the other hand, represents the accumulation of knowledge and skills acquired through education and cultural experience.

Beyond Gf and Gc, Horn and Cattell also expanded the model to include a broader variety of abilities, as shown below (Fig. 3.1).

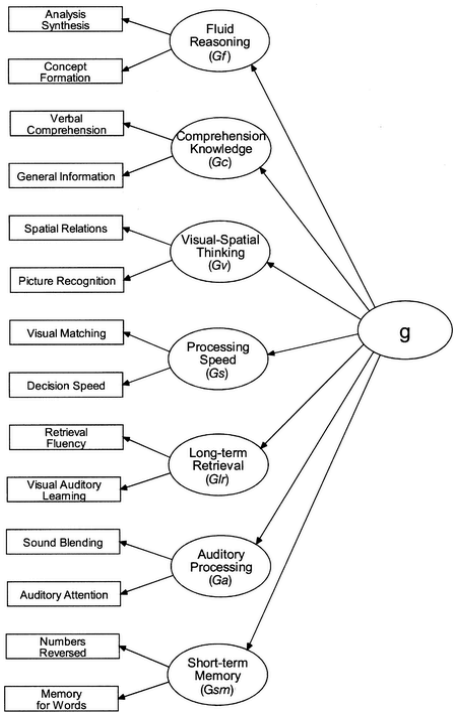


Figure 3.1: Contemporary Gf–Gc Model [39].

This evolution reinforces the concept that intelligence is not a single capacity but rather a network of interrelated yet distinct abilities, each contributing differently to human performance across contexts. moving away from the unitary view of

intelligence associated with Spearman's g factor, toward a multifactorial and dynamic understanding of these cognitive skills.
Later on, Carroll's theory [38] integrated the Gf–Gc framework into a broader hierarchical structure, known as the three stratum model of human cognitive abilities, that while showing many similarities with the Cattell Horn model, it introduced a distinctly hierarchical architecture, often represented as a pyramid.
At the apex lies stratum III, the conceptual equivalent of Spearman's g factor. Beneath it, stratum II consists of a relatively small number of broad cognitive abilities, labeled with the prefix "G," which represent fundamental constitutional characteristics that govern or influence performance across a wide range of domains, including all the abilities shown in Fig. 3.1.
Finally, at the base, lie a large number of narrower abilities corresponding to stratum I, defined also as first order factors like perceptual speed, visualization, memory span, induction, along with other intermediate skills that may serve as links between stratum II and stratum III, many of which remain only partially explored.

As seen, a persistent gap has existed between theoretical and empirical research on intelligence factors, and the practical development of instruments for assessing cognitive abilities and academic achievement. Bridging this gap is essential to align intelligence models with the tools commonly used in evaluation.
In other words, new assessment instruments should be explicitly grounded in the most current models of intelligence, enabling clinicians to select scales according to the specific abilities they aim to measure. This approach helps to avoid administering multiple subtests that capture the same ability to a single individual, or creating misleading equivalences between subtests that share similar names but in fact assess different skills..

To see how the emergence of the CHC model has influenced the development of more explicit assessment instruments, a useful example is the evolution of the Wechsler Intelligence Scale for Children [40, 41], in relation to the shift in the underlying theoretical framework.
Here, the traditional grouping of subtests into Verbal and Performance scales was abandoned, and consequently the corresponding historic terms of Verbal IQ and Performance IQ were no longer computed. The shift from a concept of intelligence as a single g factor to one as a set of multiple abilities has significantly changed the

role attributed also to the Full Scale IQ (FSIQ), increasing both the number of composite scores to be calculated, as well their specificity. Then, the importance of the FSIQ has reduced, while greater emphasis has been placed on four index scores: Verbal Comprehension, Perceptual Reasoning, Working Memory, and Processing Speed.

The examiner may also compute two additional scores: the General Ability Index (GAI), and the Cognitive Proficiency Index (CPI).

The GAI is used when there is a significant and uncommon discrepancy among the index scores and in situations where a neurological deficit is suspected.

The CPI, by contrast, is applied when it's needed to evaluate a "group of functions whose common element is the efficiency with which an individual processes certain kinds of cognitive information" [42]. This index provides a good measure of the subject's capacity to process both auditor and visual feedback.

Beyond these indices, clinicians can also use the results of individual subtests for a CHC based evaluation, allowing the measurement of broad abilities assessed by the model (Fig. 3.1). However, the WISC indices and the CHC broad abilities are not identical, the latter appear to allow for a more specific and differentiated assessment. The real value in these assessment methods is that they allow for clinical comparisons without reducing intelligence to a single reductive score, but rather highlighting meaningful patterns of strengths and weaknesses across different individuals, always accounting for their own distinctive characters.

Building on Goleman's definition of emotional intelligence, several efforts have been made to put this theory into practice by developing standardized instruments capable of assessing it. Among the most widely used are the Emotional Quotient Inventory (EQ-i) and the Mayer Salovey Caruso Emotional Intelligence Test (MSCEIT), both designed to translate the construct into measurable skills. However, the precise measurement of emotional intelligence remains debated, with ongoing debate on the validity and reliability of such tools.

An increasing body of research highlights the importance of distinguishing between cognitive processes, so how individuals solve problems, and content knowledge, intended as what they actually know.

This distinction is central to Ackerman's [43] reflections on contemporary testing practices, where evaluating both dimensions provides a more accurate understanding of intellectual performance.

Moreover, research by Vaughan and Birney [44] supports the theory by emphasizing the measurement of intra individual variability, that is how a person's cognitive performance fluctuates over time, rather than relying solely on static and single session assessments. This dynamic perspective is derived directly from the Gf–Gc framework, which highlights distinct developmental trajectories for fluid and crystallized intelligence (Fig. 3.2). Specifically, fluid intelligence, being largely biologically determined, peaks in early adulthood and then gradually declines with age, resulting in a negative trend across the lifetime. By contrast, crystallized intelligence, which depends heavily on learning and cultural background, generally continues to develop and expand throughout life, following a more positive trajectory.

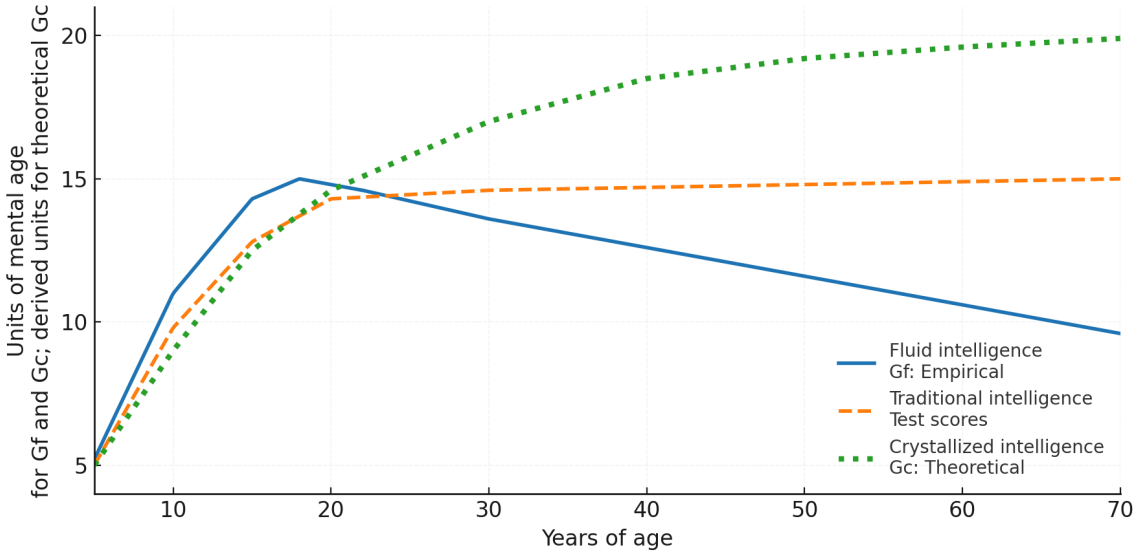


Figure 3.2: Developmental curves for fluid intelligence (Gf) vs crystallized intelligence (Gc) across lifespan [38].

The application of intelligence in real world contexts has increasingly become a focal point of analysis: Robert Sternberg [45, 46] argues that intelligence should not be evaluated only through test performance, but rather by how individuals choose to apply their cognitive abilities, whether in pursuit of selfish interests or in ways that contribute positively to society. From this perspective, a comprehensive model of intelligence must integrate not only cognitive competence but also the attitudinal and

moral contexts in which such abilities are exercised. This critique is extended to modern standardized assessments such as the SAT and ACT, which largely measure analytical and memory based skills in artificial and decontextualized scenarios. While these abilities have value, they do not necessarily translate into ethical decision making.
Detached from values and social responsibility, intelligence can even become dangerous, enabling individuals to act strategically in ways that may harm others or undermine collective well being.

This view is strongly aligned with a central issue common to the AI field, so whether an artificial model should be evaluated as a single system, judged primarily by its accuracy and performance in the tasks for which it has been explicitly trained, or as a dynamic and adaptive system, part of an interconnected framework, that for adapting to the increasing flexibility required by current real world applications should be measured for its robustness, capability to adapt, and especially to generalize to new and unseen tasks and data.
This is why evaluation cannot be limited to static measures of performance, and instead it must consider a model's capacity to generalize, as well as its ability to successfully transfer knowledge across domains, and its potential to operate responsibly in complex environments, emulating the evolution in human intelligence research.
The focus should therefore move beyond narrow measures of performance, toward assessing how abilities are applied in diverse and unpredictable contexts.

3.2 Measuring Artificial Intelligence

The evaluation of artificial intelligence systems, especially with the development of increasingly capable generative models, is currently dealing with major challenges, since traditional machine learning assessment and benchmarking methods are insufficient to stay up to date with the requirements of modern large scale generative systems.
As they become increasingly complex, conducting empirical evaluations in a systematic and reproducible way has become extremely challenging, with this difficulty that is not arising from a lack of capability by researchers, with all the significant effort and resources that have been invested to developing numerous

benchmarks and test cases, but rather from the fact that the evaluation requirements of generative models fundamentally break with the paradigm of traditional benchmarking that has succeeded in the field during last decades of progress. These frameworks are typically based on fixed tasks, clearly defined ground truths, and independently sampled test distributions, conditions that in open ended tasks are sometimes difficult to be met.

As a result, a redefinition of generalization is required, one that moves beyond measuring performance on unseen examples from a static distribution, and that instead focuses its evaluation on a model's ability to perform well on tasks that are entirely new to it, a broader notion of generalization that is more aligned with human intelligence, and that establishes a higher standard for adaptability and reasoning in artificial systems.

Chollet [4], in trying to define more precisely how this intelligent behavior should be, introduced a range of generalization levels (Fig. 3.3).

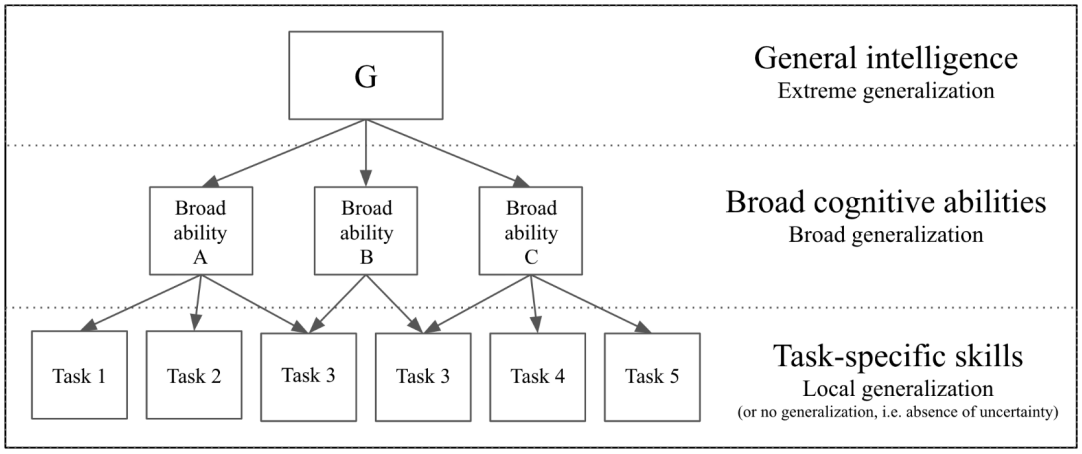


Figure 3.3: Hierarchical model of cognitive abilities and its mapping to the spectrum of generalization.

At the lowest level, with no generalization, a system performs well only in scenarios explicitly covered by its design or training data. With local generalization, the system demonstrates robustness by handling new inputs from a known distribution within a single task. Broad generalization refers to the ability to operate across a range of tasks within a given domain, adapting to novel situations without retraining. Finally, extreme generalization characterizes systems that generalize across domains and tasks, including those not specified by their creators, reflecting the adaptability observed in human intelligence.

At the current state, most artificial systems operate at the level of local generalization, with developing systems capable of extreme generalization that remains an open research

challenge, requiring evaluation methods that test cross task adaptability and general problem solving capacity. Thanks to recently developed post training techniques, artificial models can specialize and adapt to new tasks without full retraining, showing in a certain degree a sort of broad generalization.

I. Early attempts to measure AI

The earliest and most iconic attempt to define and assess artificial intelligence was introduced by Alan Turing [18]. Driven by the foundational question “Can machines think?”, he tried to find an answer through a practical experiment known as the Imitation Game, later named the Turing Test. Here, a machine is considered to exhibit intelligent behavior if it can conduct a written conversation indistinguishable from that of a human.

This method was strongly influential because it evaluated intelligence in artificial systems through observable and context dependent interaction, rather than through examination of the system’s internal mechanisms or symbolic reasoning processes. In his view, intelligence was a function of what the system could do under specific communication limits.

While elegant in its formula, the Turing Test soon revealed limitations, confusing the ability to produce human like conversation, that is an extremely easy task for nowadays language models, with general intelligence that even the most advanced models are not yet able to fully reproduce [47, 48]. In addition, the evaluation results are inherently subjective, as they depend on human judgments and provide limited insight into how or why a system succeeds or fails, being weak in terms of explainability.

Therefore, as artificial systems developed more sophisticated capabilities, it became evident that alternative evaluation methods were required.

As explained, the central principle behind the Turing Test is indistinguishibility, so if human evaluators are not able to differentiate between a human and a machine, then the machine is considered to have passed the test. Under this logic, every modern LLM to which is given a prompt to adopt a humanlike behavior is going to succeed, as proved in research by Cameron et. al. [49].

Here 4 systems (ELIZA, GPT-4o, GPT-4.5, and LLaMa-3.1-405B) were assessed in two controlled turing tests, with evaluators that had 5 minute conversations simultaneously with another human participant and one of these systems before

judging which one was really human. Not surprisingly, only when prompted to adopt a humanlike behavior (or persona), some of the models passed the test: GPT-4.5 was judged to be human 73% of the time, significantly more often than evaluators selected the real human counterpart. LLaMa-3.1, with the same prompt, was judged to be human 56% of the time, similarly to the humans they were compared to (Fig. 3.4). The other models, also earlier released in terms of timeline, achieved win rates significantly lower (23% and 21% respectively).

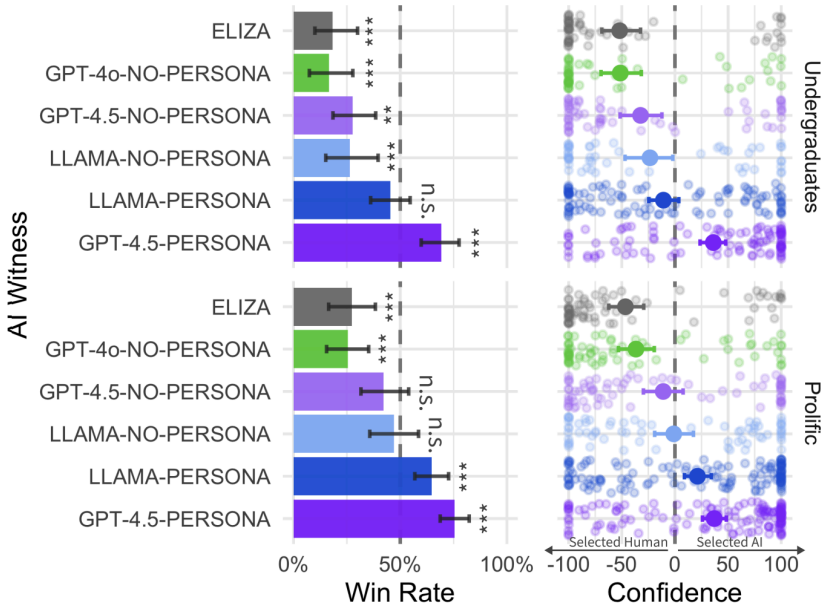


Figure 3.4: Left, win rates for each AI model, shown as the percentage of times evaluators judged the AI system to be human (asterisks indicate significance above chance). Right, confidence levels in judgments for humans vs. AI selections, with each point representing a trial (greater distance from center denotes higher confidence) [49].

Having seen that contemporary systems pass the classical version of the test, future work exploring alternative implementations could create other theoretical and practical important questions.

Turing's seminal paper is vague on how exactly a test should be implemented, including the demographic composition of participants, whether evaluators should be experts or beginners in a specific field, how familiar they should be with one another, and what incentives they should receive. While he suggested a five minute duration, he did not elaborate on whether longer sessions could provide more rigorous assessments, potentially useful to reveal more advanced capabilities.

Although much has been written about all the possible implementations, many debates have focused on whether machines could ever pass the test, and what it

might mean if one ever did. Therefore, it could be useful to understand motivations behind human evaluations in the test: only 12% of participants made their decision based on topics that Turing had originally considered, for example regarding chess or mathematics, many more focused on social and emotional aspects of intelligence, such as human like language use or personality traits.
This shows a shift from viewing traditional cognitive skills as the primary markers of humanity, possibly because computational systems now excel at logical and numerical tasks. In some cases, evaluators identified a human participant based on their lack of knowledge, reinforcing the perception that social intelligence is more challenging for machines to be replicated [49].

As seen, GPT-4.5 and LLaMa passed the test only when provided with persona based prompts, raising doubts about whether their success relied only on stylistic details, like grammar or vocabulary usage, that evaluators associate with human communication. However, this alone cannot fully explain their success, since in this three person version a machine must not only appear human, but also be judged as more convincingly human than the actual counterpart, enforcing the theory for which it's not a direct measure of intelligence, but an assessment of human likeliness.
For Turing, intelligence may have been the most significant barrier to achieve it, but since machines are increasingly matching or even exceeding human performance in cognitive tasks, other differences have become more relevant, and intelligence alone is no more sufficient to appear convincingly human.
Given also the complexity and multidimensionality nature of intelligence discussed so far, no single test can be considered as definitive, and to the extent that the Turing test indexes intelligence, it should be considered only as one part of evidence among others.
Current debates over whether LLMs can be considered intelligent increasingly focus around the validity of the benchmarks and evaluation methods employed, and the concern that these assessments remain too narrow and metric based. Here, the Turing test can be understood as providing a complementary form of evidence, since it's grounded in interactive judgments made by human evaluators rather than in a static evaluation.

II. Current evaluations methods

Evaluating modern LLMs is particularly challenging because they are not simple binary classifiers, but generative systems capable of producing extended text and open ended answers.
As new models are released with always improved capabilities, the overall evaluation becomes increasingly difficult, reflecting not only the growing complexity of their architectures but also the greater variability and sophistication of their responses.
Considering that standard metrics are optimized for lexical overlap, they often fail to capture actual quality or relevance, with a model that may obtain high scores on metrics while still generating incorrect or misleading content, demonstrating that similarity is not equal to usefulness. Instead, LLMs should be evaluated by more human centered standards, being helpful and reliable, dimensions that are difficult to quantify.
The current state of the art in 2025 involves increasingly using LLMs themselves as evaluators, with models generating responses which are then assessed by another model, acting as a judge. Although this approach may appear recursive, it has proven effective in practice: leading organizations such as OpenAI, Anthropic, and Hugging Face have implemented many variants of it, and specialized critic models have been developed to rate outputs and provide systematic feedback.

Given the limited scope of single benchmarking tasks, the comprehensive evaluation of LLM based systems requires end to end metrics that consider all the steps performed by the model, as well as the output generated. Such approaches not only enable a more holistic assessment of system performance, but also point toward a longer term goal of developing dynamically self optimizing models.
At the same time, LLMs present a unique evaluation challenge being generative systems, due to the open endness of their outputs, as well as its unpredictability in many cases, demonstrating a general degree of complexity ever shown before by artificial systems.
Therefore in tasks like insight generation or in dialogues, where variability in responses is a defining characteristic and makes it difficult to establish clear success criteria, there are issues similar to those encountered in the Turing Test.
As a result, evaluations often rely on human judgment to decide whether a response is relevant or clear, an approach that is often costly and even inconsistent. In

addition, LLM outputs can vary significantly based on small changes in the prompt or the trajectory of a conversation, showing the large number of conditions that must be tested to ensure evaluations are both reliable and repeatable, a fundamental requirement for a successful benchmarking framework.

The most common evaluation strategy combines quantitative metrics and qualitative analysis, with metrics that can be categorized into automatic statistical, model based metrics, and human assessments, with additional task specific custom metrics that can also be employed for particular use cases.
This evaluation process follows the staged approach of the development lifecycle in artificial models. At each step, LLMs require distinct evaluation sets due to the goal of its training: in fact every machine learning model is designed to be trained on massive amounts of unlabeled text data to learn general patterns of language, like grammar and semantics, acquiring broad linguistic and world knowledge so the model can work properly. However, pretraining provides general competence on data but doesn't guarantee alignment with human intentions, or task specific expertise, requiring deeper calibrations. This is why the purpose of pre training evaluations is mainly to assess the model's general language understanding before it's specialized, evaluating the quality of learned knowledge and identifying potential biases or gaps. Typically, this stage is assessed through statistical metrics, among which the most used are:

- Accuracy: measures the proportion of correct predictions made by the model out of the total. It's widely used in classification tasks but limited if not complemented with other metrics.
- Perplexity: it reflects how well language models predict text. Lower values are signs of greater confidence, but while useful for model comparison and training evaluation, it doesn't assess whether outputs are factually correct or useful, it only shows how closely the output follows common language patterns.
- Cross Entropy Loss: measures the difference between predicted and true probability distributions, offering a more sensitive indicator of model confidence than accuracy because it evaluates the entire token sequence, like a full sentence. Lower values indicate better performance, reflecting greater confidence in correct predictions

A range of other several metrics can be used to evaluate the pre training phase, but while useful, they generally operate at the same static level. These measures remain valuable for determining whether a model has successfully captured general data patterns, yet they are not suitable as effective evaluation tools once the model requires fine tuning.
One limitation is that they rely on static and publicly available datasets that may not reflect the characteristics and requirements of each domain. In practice, models are expected to operate in real applications on dynamic or proprietary data, often in contexts where they are not in optimal formats that benchmarks might not capture, and often dealing with class imbalances and noisy inputs. In addition, many of these static datasets are now saturated, with top performing models reaching or even surpassing human level scores, so further developments on these tests may not be signs of improvements in performance, highlighting the need for more advanced and diversified ways of evaluation.

For these reasons, post training evaluation relies on task specific metrics and, in many cases, human in the loop assessments to better capture the context behind performance, and to provide the level of dynamism required for applied use cases. Once a model has been post trained through fine tuning or related techniques, and adapted to a specific task, its evaluation must shift from measuring broad language competence to assessing task specific performance, based on the task's nature, the dataset's characteristics, and other requirements of the system once deployed.
The most commonly used metrics for post training evaluation include:

- Precision: measures the proportion of positive predictions that are correct, with higher values indicating fewer false positives, computed as:

  $$Precision = \frac{TP}{TP + FP}$$

- Recall: it's the proportion of actual positive cases correctly identified, with higher values indicating fewer false negatives, computed as:

  $$Recall = \frac{TP}{TP + FN}$$

- F1-Score: it's the harmonic mean of precision and recall, balancing trade-off between them.

  $$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

In generation tasks, metrics such as BLEU (precision oriented n-gram overlap) and ROUGE (recall oriented overlap) are frequently applied to tasks like machine translation and summarization, and while useful, they primarily capture similarity rather than semantic accuracy or factual correctness.
In practice, precision and recall are often considered together, since improving one typically comes at the expense of the other.

All these metrics are being widely used because they are automatic, consistent and provide a quick way to compare different models or fine tuning results, however they still imply several limitations. First, most rely on having a reference output or ground truth for comparison, which is often impractical in open ended tasks such as dialogue generation, where no single correct response exists and costly data creation is required to approximate one. Second, they primarily assess superficial similarity: a model's output may express the same meaning as the reference but in different words and receive a low score, while copying parts of the reference can result in a high score despite producing irrelevant or incoherent text. Third, these measures fail to directly capture key dimensions outlined earlier, like accuracy or fairness in response, such as a grammatically correct sentence that matches the reference but may still be factually incorrect. And finally, optimizing models to improve only these specific metrics can lead to overfitting, where the model learns to perform too well on the metric without truly improving its quality.

Single metrics provide clear and quantitative signals of performance on a specific task, but while being valuable for narrow and well defined evaluations, they fail to capture the multidimensional nature of LLMs. In fact, they are not designed to excel at one task alone, they must demonstrate competence across a broad range of domains and situations, especially since they are moved by prompts from humans with language variability. Therefore, benchmarks are more useful than single metrics because they integrate multiple tasks, datasets, and evaluation criteria into a unified structure. For instance, a model may achieve high accuracy on a classification task, but a benchmark reveals whether it can also perform well on natural language inference or question answering.
Another advantage of benchmarks is that they enable comparability across models and research groups. Standardized benchmarks establish common ground, ensuring

that results are not artifacts of task choice or dataset selection. This comparability fosters progress by allowing researchers to track improvements over time and across different architectures. Moreover, benchmarks can incorporate emerging capabilities and risks: while a single metric may remain static, benchmarks evolve to include new tasks and new evaluation dimensions. In this way, benchmarks better reflect the complexity of real world applications, where LLMs are required to perform dynamically across diverse conditions rather than optimizing toward a single number.

Like individual metrics, also benchmarks are specific for tasks, each evaluating different aspects of models.
For pretraining, SuperGLUE is the most used one, as it's an advanced version of the GLUE (General Language Understanding Evaluation), designed to test generalization across complex NLP systems.
Another widely adopted one for general evaluation is MMLU (Massive Multitask Language Understanding), which is designed to evaluate models on their ability to handle a broad range of subjects and tasks. Its strength consists in assessing knowledge acquisition and general language understanding across diverse domains, but it remains limited since it primarily measures performance on predefined tasks with fixed answers, rather than testing the ability to generate novel insights from multiple datasets or to reason about complex, real world relationships.
BBH (Big-Bench Hard) focuses on evaluating models through particularly challenging tasks that require multi step reasoning, and unlike knowledge oriented benchmarks, it emphasizes the capacity for logical inference and reasoning under complexity, making it a valuable complement to knowledge based benchmarks, as it pushes models to demonstrate reasoning skills beyond the simple recall of information.
In the post training phase, for the high specificity of the tasks, are used evaluation benchmarks that focus the performance of the models on specific domains. Most of them are not standardized public benchmarks, but custom evaluations to address each own use case.
Within model development, the HumanEval benchmark is used to assess coding abilities by requiring models to complete Python programming tasks, typically through the generation of functions that satisfy given specifications. Its primary

focus is on evaluating code generation and programming understanding, making it highly relevant for applications in software development. However, its scope is limited since it emphasizes correctness in isolated programming tasks but does not extend to broader aspects like pattern recognition, data handling, or statistical analysis, which are essential for generating novel insights in real world problem solving. More broadly, strong performance on a benchmark doesn't guarantee a model is good in real applications since they are still static measures, while real environments are dynamic and continuously evolving.
Moreover, many benchmarks are built on the assumption of a single correct answer, while in practice most real world questions are multifaceted and open ended. In such contexts, a single numerical score is insufficient to capture the nuances of subjective and context dependent tasks.
Complex problems cannot be adequately explained or evaluated by reducing them to a binary value.

Despite being still commonly used in research, particularly for tasks with available reference text like translation and summarization, every quantitative measurement is most effective when combined with other complementary evaluation methods that can capture deeper aspects of model performance, including human evaluation.
Human judgment is especially valuable for subjective dimensions like coherence, tone, and alignment to human values, ensuring that evaluation captures both quantitative performance indicators and qualitative attributes compliant with end user needs.
Because this approach can be time consuming and expensive, both in terms of cost and resources invested, it's typically applied to a smaller sample of outputs rather than to an entire dataset.
Since human evaluation doesn't follow direct numerical metrics, it's essential to define clear evaluation criteria: in comparative tasks, for instance, annotators must be instructed on whether to judge overall quality, factual accuracy, or other specific attributes.
Human evaluation can indeed address multiple dimensions simultaneously, which reflects the flexibility of this approach but also increases the time and effort required to judge.

Blinding and randomization are also important aspects to consider when reducing biases: by comparing outputs from two different models, the order in which they are shown should be randomized so that annotators do not develop preferences based on model identity. Similarly, if the evaluation includes human written references, raters should not be told whether an answer was written by a model or a person, a successful trick that has been taken from Turing's imitation game.

This approach still presents several challenges that limit its practicality and reliability, especially at scale: conducting human evaluation for every model version or parameter update is not feasible due to the time and resources required, as well as the costs. As a result, teams often reserve human evaluation for key checkpoints, like comparing final model variants, while relying on automated metrics to monitor day to day progress.
Time constraints also pose a challenge, since setting up human evaluations, collecting responses, and cleaning the data can take days or weeks, time in which the model may continue to evolve, creating a delay between generation and evaluation, and ultimately slowing the feedback loop and affecting how actionable the evaluation results are going to be.
Moreover, judges' bias and subjectivity are other critical issues, especially in tasks without a known ground truth. People implicitly bring their own preferences and cultural backgrounds to the task, so if the model is intended for a global audience, it's important to include a diverse group of annotators to avoid skewed results and better reflect the range of user expectations. For example, an annotator's judgment of whether a response is polite may vary depending on his linguistic or cultural norms. These best practices are all equally relevant for the development of more robust evaluation methods, including adaptations of the Turing Test.

Practically, the best strategy is to combine human and automated evaluation methods in controlled environments: automated metrics can provide continuous feedback during development, while human evaluations are used on a regular basis to ensure that improvements reflect real gains in output quality.
Human feedback can also play an active role during model training, as with Reinforcement Learning from Human Feedback (RLHF), where human preferences guide and optimize the model's behavior in real time, an approach that has become central in the current wave of Agentic AI development.

An additional consideration is on the ethical dimension, as annotators must be guided on how to respond when models generate harmful or offensive content.
Safety and fairness have never been as central as they are today, requiring specific evaluations: safety can be assessed by exposing models to sensitive or potentially harmful prompts, and measuring if inappropriate responses are generated, with automated tools like toxicity detectors can identify unsafe outputs, and the proportion of flagged responses can serve as a metric. Fairness, instead, can be evaluated by testing whether the model treats different demographic groups, for example by posing equivalent questions about various groups and comparing the tone or sentiment of the generated responses. Developing models that ensure fairness is a pressing requirement for all actors in the AI field, involving efforts to eliminate the inherited biases that systems have absorbed from historical data and early developments.

3.3 Dynamic Approaches for Evaluation

Given the limitations of narrow quantitative metrics and the heavy requirements of human evaluation, current research is increasingly oriented to develop evaluation systems that are more dynamic and autonomous, evaluating systems end to end in a holistic way.
Traditional metrics provide useful insights into a model's ability to learn data patterns and achieve task specific accuracy, but when they are deployed into real world applications is a totally different story. The growing complexity of LLMs, both in their architectures and their expanding capabilities, reinforces the need for such holistic approaches. Modern models are no longer isolated entities, but the core of broader interconnected ecosystems that may include retrieval augmentation, external memory, multi agent collaboration, or even the deployment of multiple tools simultaneously, demonstrating increasing capabilities in solving tasks. A key way of capturing this progress is by measuring AI performance in terms of the length of tasks AI systems are able to complete: this trend has been consistently growing at an exponential rate over the past six years, with a doubling time of roughly seven months. Extrapolating this trend suggests that within the next decade, AI systems and agents could independently complete a substantial fraction of software tasks that currently require humans days or even weeks to accomplish (Fig. 3.5).

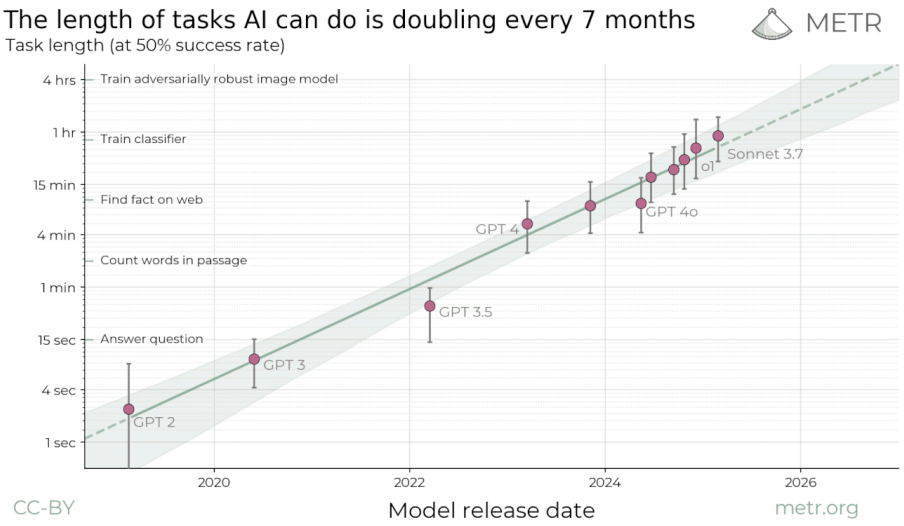


Figure 3.5: The task length that frontier model agents can complete autonomously with 50% reliability has doubled about every seven months over the past six years [50].

In this context, static benchmarks built on fixed methods are no longer sufficient to evaluate their overall performance, highlighting the need for comprehensive benchmarking frameworks, rather than reliance on isolated standardized metrics. Therefore, the development of evaluation should increasingly emphasize dynamic approaches and adaptive benchmarking frameworks, that provide flexible evaluation setups that better reflect the generalization, robustness, and reliability of LLM based systems across diverse scenarios, assessing them in more fluid, interactive, and context aware environments, providing a more realistic measure of both utility and intelligence.
Further on, are explored several emerging strategies and how they try to fill the gap left by traditional evaluation paradigms.

In the field of benchmarking, recent evaluation initiatives have moved beyond task accuracy to address broader concerns such as robustness and reliability, as well as ethics compliance.
The HELM (Holistic Evaluation of Language Models) framework, developed at Stanford, represents a shift by adopting a multidimensional perspective on evaluation: in fact rather than focusing only on correctness, it assesses models across several aspects, like fairness, bias, toxicity, and efficiency (Fig. 3.6). It also measures the model behavior under varying conditions, such as noisy input or high uncertainty,

combining both quantitative and qualitative indicators. Unlike static benchmarks, HELM functions as a dynamic evaluation platform, offering a more comprehensive view of model performance and behavior in real world scenarios, something that is more aligned with the needs of current generative models.

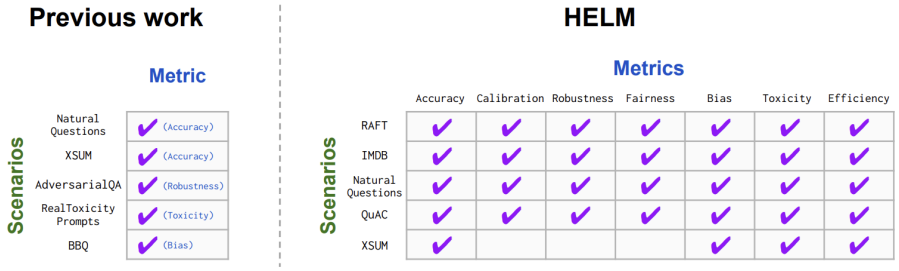


Figure 3.6: Unlike most prior benchmarks for language technologies, which emphasize accuracy and treat other criteria in separate or limited datasets, HELM adopts a multi metric approach, highlighting dimensions beyond accuracy and making it possible to systematically analyze the trade offs among different metrics [51].

Recent language model based evaluators go beyond similarity scoring to assess responses holistically, using large models, such as GPT-4, not just to compute semantic proximity but to reason explicitly about the quality of an output. Instead of matching tokens to a reference, LLM judges can be prompted to evaluate dimensions like accuracy, coherence, safety, or persuasiveness, providing explanatory feedback alongside scores.
In this way, evaluation has evolved from token overlap (with BLEU, ROUGE and other statistical metrics), to embedding based semantic similarity, finally becoming reasoning based evaluators. Each step reflects a move toward capturing deeper, more context aware qualities of generated text and aligning more closely with human expectations of quality, usefulness, as well as truthfulness.

As LLMs continue to improve, they show potential in replacing human annotators in many tasks, and a recent and extremely interesting approach involves using themselves as reasoning evaluators. This method, often referred to as LLM-as-a-judge, gained popularity especially with the development of advanced reasoning models like GPT-5, and consists of prompting an LLM to assess the outputs of other models, or even its own one. This allows to reduce the reliance on human annotators by automating the evaluation process, resulting in a more scalable and automated approach.

The evaluation model is provided with the original prompt or input, and two candidate outputs, one from Model A and another from Model B, in this case. It's then asked to judge which output is better, based on specific criteria such as accuracy, clarity, politeness, or alignment with task instructions.

The format of the evaluation can vary, with some cases where the model simply selects the best between two answers, and in others where it assigns a numerical score, providing a detailed explanation for its choice.
Prompting strategies are important for guiding this evaluation effectively: for instance, including structured guidelines or using chain of thought prompting, where the model is encouraged to explain its reasoning step by step, has been shown to improve judgment quality and transparency [52].

This approach provides two main advantages: first, it enables scalability by reducing the need for human involvement, supporting scalable benchmarks and rapid iterations. Second, LLM judges generate not only scores but also explanations, making their reasoning interpretable.
On the other hand, it also presents several limitations and biases. A major concern is the phenomenon known as self enhancement bias, where a model tends to favor responses it has generated itself. Other issues include a predisposition to favor outputs in certain positions over others (Fig. 3.7), a tendency to prefer longer and more verbose responses even when shorter ones may be clearer or more accurate, and limited judgment capabilities in certain domains.

Judge	Prompt	Consistency	Biased toward first	Biased toward second	Error
Claude-v1	default	23.8%	**75.0%**	0.0%	1.2%
	rename	56.2%	11.2%	**28.7%**	**3.8%**
GPT-3.5	default	46.2%	**50.0%**	1.2%	2.5%
	rename	51.2%	38.8%	6.2%	**3.8%**
GPT-4	default	**65.0%**	30.0%	5.0%	0.0%
	rename	**66.2%**	28.7%	5.0%	0.0%

Figure 3.7: Position bias across different LLM judges. Consistency measures the percentage of cases where a judge gives the same result after swapping the order of two assistants. Biased toward first reports the proportion of cases where the first answer is favored. Error reflects outputs in the wrong format. The two highest values in each column are highlighted in bold [52].

Given these limitations, LLM-as-a-judge is a very helpful technique but not yet a replacement for human eval in all cases.
Despite this, model based evaluation remains central for progress, serving both as a reward model during training and as a partial substitute for human evaluation.
Typically, the standard approach to train such evaluators involves collecting large volumes of human judgments on model outputs, a process that is costly and could become rapidly obsolete as models continue to improve.

One proposed solution introduced by Wang & Tianlu et al. [53] is to improve evaluators without relying on human annotations, by leveraging synthetic training data. Starting with unlabeled instructions, this iterative self improvement approach generates contrasting model outputs and trains an LLM-as-a-judge to provide both reasoning explanation and final judgments.
At each new iteration, the model is retrained using the enhanced predictions from the previous cycle, gradually improving its evaluation capabilities.

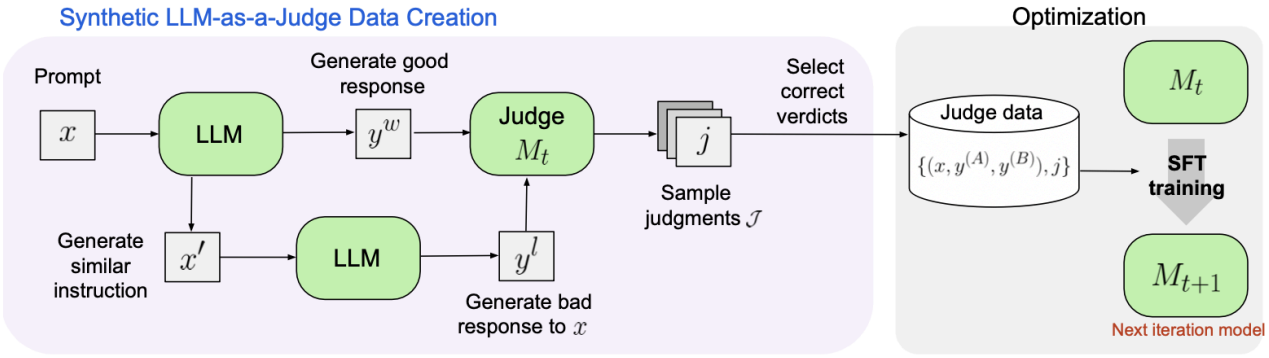


Figure 3.8: The self-taught evaluator scheme trains an LLM to judge its own outputs by iteratively generating responses, producing synthetic preference data (good vs. bad answers), and using a judge model to evaluate them; these judgments are then fed back into supervised fine tuning to improve the next model version [53].

Synthetic data has emerged as a promising solution for efficiently acquiring training examples, and can be particularly valuable in situations where real world data can be hard to access, for example in the case of weather data, and where correct annotations can be challenging to acquire. Additionally, they have the benefit of being easily customizable to specific requirements, like different evaluation criteria or safety constraints, avoiding further post training steps.

Their use has already demonstrated benefits in many applications: in model alignment, they have been used to improve the original model's capabilities and teach the model new skills, while in evaluation to measure tasks such as safety and general instruction following, showing strong correlation with real human judgements.

A fundamental truth is that most benchmarks and current metrics measure performance rather than intelligence. They are designed to test whether a system can produce the correct answer to a given question, but they don't assess whether the model truly understands the concepts, if it can generalize across domains, or if it can apply flexible reasoning to new situations.
As models increase in scale and capabilities, they also tend to become more opaque and less clear in their functioning. This is defined as the "black box problem", where models generate outputs without transparent or interpretable reasoning, making it difficult to determine the depth or validity of their cognitive processes.
This point is similar to the long debates around human intelligence testing, where correct answers on an IQ test does not necessarily correspond to real intelligence, particularly when the evaluation context is biased or narrow.
The problem is derived by the growing complexity of AI systems, which has surpassed current methods of evaluation. Modern LLMs, such as the GPTs, Claude, or LLaMA, have evolved into generalist systems capable of performing a wide range of cross functional tasks, from essay writing and code generation to dialogue and domain specific insight discovery. However, the benchmarks commonly employed to evaluate them remain fragmented and bounded, failing to capture the emergent cross domain capabilities that define these models.
As a result, there are discrepancies that emerge, and these inconsistencies raise a deeper question: what does it mean to call a model intelligent, and is it truly reasoning?
If benchmarks only measure narrow task performance, they risk overlooking the broader, adaptive qualities that intelligence, both human or artificial, should demonstrate.

Chapter 4: A dynamic benchmarking framework for LLMs and the impact of complexity

4.1 Theoretical foundation behind complexity: The illusion of thinking

The rapid development of large language models has increased attention to their capacity for reasoning, in part fueled by the adoption of chain-of-thought prompting that allow models to perform and show intermediate reasoning steps [54], and by the rise of a new class of systems described as large reasoning models. These developments have changed the expectations of what generative models can achieve, shifting the focus from fluency and coherence in natural language generation to a deeper aspect, whether such systems are capable of structured reasoning and genuine insight discovery.

Current evaluations primarily focus on benchmarks that highlight final answer accuracy, but this strategy has become limited due to data contamination and doesn't provide insights into the reasoning structure.

Yet, the already asked fundamental question remains: do these models actually reason, or do they only simulate reasoning by reproducing linguistic pattern recognition and matching?

Apple's paper "The illusion of thinking" [55] provides a strong systematic analysis toward finding a solution: it reveals that much of what current benchmarks interpreted as evidence of reasoning in LLMs may in fact be an illusion itself, created by evaluation frameworks that fail to consider the role of complexity in problem solving, prioritizing output's accuracy.

While models often appear proficient in tasks of low or medium complexity, they collapse under conditions of higher difficulty, revealing not only the fragility of current evaluation methods, but also the structural limits of LLM reasoning itself. In fact what looks like intelligence at the surface may be little more than the model's ability to memorize and statistically recombine solutions that only resemble reasoning, without actually demonstrating it.

This critique is significant for two reasons: first, it clarifies the limitations of current reasoning models, showing that their problem solving abilities do not scale directly with task complexity. Second, it exposes the limitations of existing evaluation and

benchmarking methods, risking to attribute models some reasoning abilities without truly testing the depth or generalizability of their skills.
By focusing too strictly on final answer accuracy, traditional benchmarks fail in considering how models reach their conclusions, or whether they can keep coherence across longer reasoning processes, or with more variables to consider.
Even more critically, they don't systematically consider problem complexity, which is a key factor in understanding their limits, and it must play a central role in benchmarking new generation models.
The challenge, therefore, is not building models with stronger reasoning abilities, that is becoming easier thanks to increasingly available computational resources and technical know how, but in designing evaluation frameworks and benchmarks that consider complexity as a central driver that needs to be measured.

This recalls Chollet [4] proposed definition of intelligence as skill acquisition efficiency, posing emphasis not on narrow task performance but rather on a system's ability to generalize and adapt across a wide range of unfamiliar scenarios.
However, before building such systems, it's essential to clearly define and evaluate what constitutes intelligence: a common definition of AGI is "a system that can automate the majority of economically valuable work", that while useful in defining broad objectives, fails to really capture the essence of intelligence.
From this perspective, many current benchmarks may be seen as similar tools to narrow IQ tests: useful in constrained contexts, but inefficient if taken as comprehensive measures of reasoning ability. Automating tasks at scale doesn't necessarily require reasoning or adaptability, it may simply reflect efficiency in executing routine operations.
It's precisely this reductionism that Apple's study underscores, highlighting the urgent need for new forms of benchmarking that move beyond outcome correctness and instead capture the robustness and generalization of problem solving.
For many years, LLMs worked like black box systems, producing answers without revealing the reasoning that led to them. The introduction of newer models like OpenAI's o1 and o3, DeepSeek-R1 and Claude 3.7 Sonnet, marked a shift in their approach, starting to generate intermediate reasoning steps, or hidden chains of thought, before presenting a final answer, with some of them even providing self reflection mechanisms to refine their outputs and continuously improve.

Unlike earlier studies that focused on statistical metrics, or on the degree of human likeliness to evaluate the reasoning capabilities of language models, the deployment of controllable environments allows for precise manipulation of problem complexity while keeping consistent logical processes, enabling a more rigorous analysis of reasoning patterns and consequently of their limitations.

At the current stage it's not clear whether the performance enhancements of recent post trained thinking models are caused by an increased exposure to a greater amount of data, increased computational capabilities, or to a reasoning ability developed by such models.

Nevertheless, looking at how reasoning models perform on benchmarks focused on outcomes can still give useful insights: for example, on the MATH-500 dataset [55], which measures mathematical problem solving ability, thinking models and their non thinking counterparts achieve broadly similar results given the same inference token budget, suggesting that apparent reasoning advantages may diminish when computational resources are the same.

In contrast, the picture changes on other benchmarks like AIME24 [55] and AIME 25 [55], where reasoning oriented models display a marked performance gap over their non thinking counterparts, giving a first suggestion that relative advantage of reasoning may depend not only on token budget but also on complexity and compositional structure of the tasks themselves.

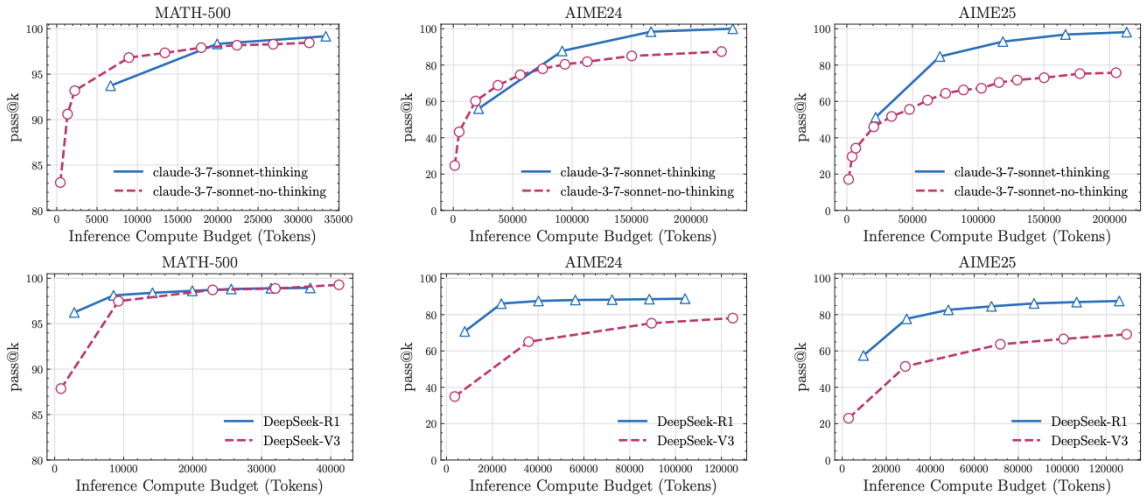


Figure 4.1: Comparative analysis of models with and without "thinking" modes across mathematical benchmarks shows an increase in performance when thinking mode is enabled [55].

By measuring models' performance across static benchmarks, in this case math related ones, it's clear that performance generally improves when a thinking mode is

enabled (Fig. 4.1), as it allows models to allocate more computational effort to the reasoning process, almost like humans taking additional time to solve complex tasks. However, this also raises a critical concern about the reliance on standardized benchmarks: they capture measurable improvements but fail to evaluate the model's underlying reasoning or generalization capabilities, reflecting optimization toward specific test sets.

For this reason, setting a controlled environment allows for a more precise manipulation of problem complexity, deliberately designed to be transparent, scalable in complexity, and free from potential contamination in pretraining data.
In the considered theoretical framework developed by Apple, such environments are constructed through formal puzzle domains, enabling the evaluation of models under conditions where performance could not be attributed to memorization or data contamination, but instead to genuine reasoning abilities.
Among the puzzles used, the most convincing case for analyzing evaluation failures through complexity is the Tower of Hanoi, since it allows to track how complexity scales exponentially in a mathematically and transparent way, while the other tests employed in the study don't offer the same precisely defined and exponential scaling of complexity, relying instead on state tracking, which makes them valuable for testing coherence but less rigorous for exposing reasoning collapse in a measurable way.
The Tower of Hanoi is a classical puzzle used in computer science and cognitive psychology to study structured reasoning and planning strategies.
The problem consists of three pegs and a set of discs of decreasing size stacked on one, with the objective to transfer the entire stack from one to another, following strict rules: only one disc can be moved at a time, and no larger discs may be placed on top of a smaller one. What makes it a valuable tool is its well defined complexity growth: with each additional disc, the minimum number of required moves doubles plus one, increasing exponentially and making it an ideal environment for controlled benchmarking, since every increment in complexity corresponds to a precisely calculable increase in difficulty.
Tasks have been grouped into three complexity levels: low (in yellow), medium (in blue) and high (in red), representing the same type of task but with an increasing number of variables.

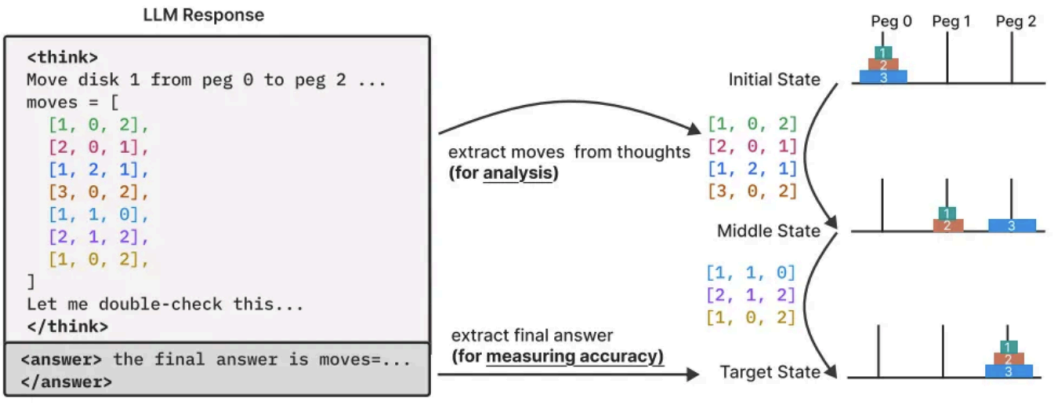
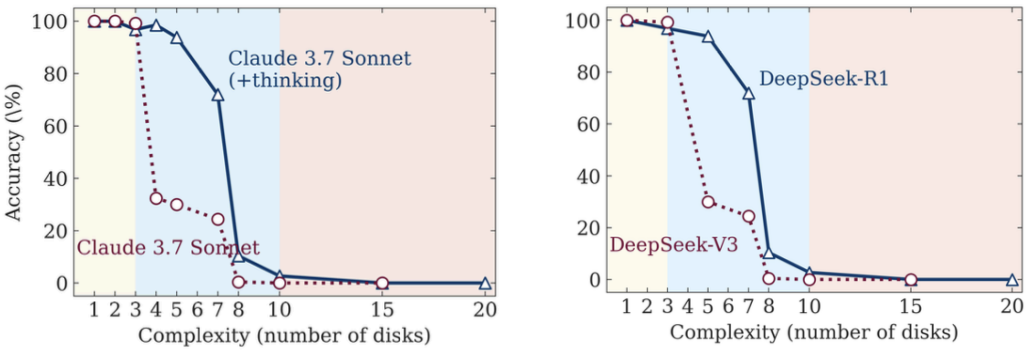


Figure 4.2: LLM solution to the Tower of Hanoi, showing intermediate reasoning steps (<think>) and the final answer (<answer>). Moves are analyzed for reasoning quality, while accuracy is measured against the correct target state [55].

While classical algorithms can solve the Tower of Hanoi problem efficiently, large reasoning models start to fail when reaching a certain complexity level.
They begin to show significant performance drops as the number of discs increases: by the time the puzzle reaches eight discs, both thinking (in blue) and non thinking (in red) models fail to produce valid solutions. Initially, as task complexity grows, the number of tokens generated during reasoning also increases, which aligns with expectations. However, beyond a certain complexity threshold, the token count decreases showing a breakdown in reasoning ability.
Although reasoning models perform better than non thinking ones at medium complexity, both approaches collapse beyond this threshold, often generating shorter and incomplete reasoning traces.



a) Accuracy vs Complexity for Claude 3.7 Sonnet (thinking vs non thinking).

b) Accuracy vs Complexity for DeepSeek-R1 (thinking) and DeepSeek-V3 (non thinking).

Figure 4.3: Accuracy of thinking models (a) Claude 3.7 Sonnet with thinking and (b) DeepSeek-R1 versus their non thinking counterparts in the Tower of Hanoi puzzle environment [55].

For tasks that are solved correctly, thinking models generally produce valid solutions faster at low complexity levels, but require increasingly more intermediate reasoning steps as complexity rises, indicating that as the number of variables increases it's needed an additional computational effort to maintain coherence and reach the correct outcome.

More interestingly, analyzing the reasoning steps reveals that the number of tokens generated tends to increase with task complexity, but then declines once a certain threshold is reached, showing a breakdown in the reasoning process once complexity exceeds a critical level.

At moderate levels of complexity, reasoning models outperform their non reasoning counterparts, but they require more computational steps to do so. Beyond this threshold, however, they collapse, producing shorter and less coherent reasoning traces.

In correctly solved tasks, tested models are able to produce valid solutions quickly in low complexity scenarios, but require progressively longer reasoning sequences as complexity rises, while in unsuccessful cases, they often remain stuck to incorrect intermediate steps, eventually finishing their token budget without really converging on a correct solution.

Both successful and unsuccessful cases therefore reveal inefficiencies in the reasoning process: in the first efficiency declines as solutions demand disproportionately longer reasoning sequences, and in the second inefficiency manifests in the inability to correct early errors, leading to wasted computation and failure.

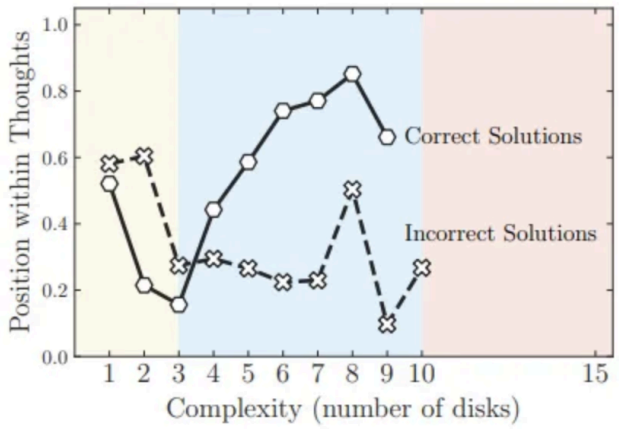


Figure 4.4: relationship between task complexity (measured by the number of discs in the Tower of Hanoi) and the position of correct and incorrect solutions within the reasoning trace [55].

As said, the decision of focusing on the Tower of Hanoi to test the capabilities of reasoning models is due to three reasons: it's controllable, since task complexity can be systematically scaled by adding discs and each increase gives a mathematically defined exponential rise in difficulty; it's verifiable, as the correct sequence of moves is fully known in advance, which enables researchers to check the model's output step by step against the optimal solution; and finally, it's clean, given that these formal puzzles are not appearing in pretraining data, reducing the risk of data contamination and ensuring that success cannot be given to memorization.

These findings explain the theoretical critique advanced in the illusion of thinking [55], since reasoning traces, despite their superficial connection to human logical processes, often mask inefficiencies and structural limitations. The collapse observed at higher complexity levels shows that current LLMs architectures, whether labeled as thinking or non thinking, cannot keep performance once the task exceeds the patterns learned during training. It's also possible to say that existing benchmarks hide this illusion by reducing evaluation to final answer accuracy, not accounting for both intermediate reasoning traces and the decisive role of increasing complexity. This is precisely the core of current benchmarks limitations, and the motivation behind the development of the benchmark developed with SparkBeyond, that will be presented in detail in the following sections.
Instead of relying on static datasets that test correctness in narrow domains, it will use synthetic data generation to build controlled environments where complexity can be manipulated with precision to understand where agents struggle, much like the Tower of Hanoi that provides a mathematically transparent scale of difficulty,
In this sense, the framework shifts the focus from outcome correctness toward the dynamics of reasoning itself, capturing more core dimensions that are critical for a full comprehensive evaluation.

4.2 SparkBeyond case study: Development of a strong dynamic benchmarking framework and agents testing

I. SparkBeyond and Project overview

SparkBeyond is an innovative AI driven analytics company headquartered in Israel and with operational hubs worldwide, founded in 2013 by Sagie Davidovich (past CEO & current President) and Ron Karidi (CTO), with the goal of building

technology that augments human intelligence by automating hypothesis generation, uncovering insights from complex data, all leading to accelerating problem discovery.
The company's mission is centered around transforming traditional analytics into proactive and explainable decision intelligence that moves beyond pattern recognition, toward the optimization of decision making across all aspects of business operations.
SparkBeyond is trying to solve one of the most pressing challenges that organizations are facing today: while modern AI systems such as large language models can act as powerful discovery tools, the most valuable enterprise knowledge often remains locked within structured operational systems. Data collected across customer relation management and enterprise systems, and production records accounts for more than 80% of real enterprise value, yet it remains largely underutilized since LLMs, even when enhanced with retrieval based techniques, lack direct access to such structured datasets and therefore fail to base their results on the quantifiable realities of business performance.
To address this gap, SparkBeyond developed its proprietary Hypothesis Engine, an AI driven mechanism created to discover structured data through a scientific and systematic approach: it generates and evaluates millions of hypotheses against interconnected KPIs, using automated feature engineering and advanced ranking methods. Unlike traditional black box models that often hide interpretability, it provides consistent and transparent insights, highlighting not only correlations but also causal patterns and anomalies that shape business outcomes.

Without structured data access, generative outputs remain disconnected from enterprise performance, and therefore untrustworthy. To fill this gap, the company has developed a solution called "structurally grounded generative reasoning", a method where insights discovered via structured data analysis serve as inputs for LLMs, which then narrate, recommend and adapt decisions accordingly.
Above all, SparkBeyond sees this approach as a foundation for the future of agentic AI: GenAI will not just summarize or answer questions based on simple prompts, instead it will observe KPI shifts, detect anomalies, hypothesize causes, simulate impacts and suggest next best actions autonomously.

SparkBeyond's contribution to the GenAI space is then not to train new models, but to redefine how they are used. This is a deeply grounded, domain aware, decision centric vision, where the eloquence of LLMs is matched with the precision of statistical insight, and where creativity is no longer separated from operational truth. In doing so, it's re imagining what intelligent systems can do when they speak from data, not just about it.
Insight discovery is a task that strongly differs from prediction: despite predictive modeling is concerned with producing accurate outputs given specific inputs, evaluating accuracy against labeled outputs, insight discovery involves generating interpretable statements in natural language that explain why outcomes occur and what patterns drive them.
Therefore insights are intended not as predictions, but as explanations, correlations, or causal patterns expressed in human readable terms, and their evaluation requires not only checking correctness but also measuring whether the agent discovers all relevant insights, if they are truly grounded in data, and whether it communicates them coherently, aspects of evaluation that surpass output accuracy.

To bridge this gap, SparkBeyond developed the first ever benchmark for evaluating insight discovery capabilities in AI agents, introducing a dynamic and synthetic approach to benchmarking consisting of both a problem generation system and an accompanying evaluation framework with performance metrics.
This gives important contributions in three ways: firstly, it establishes insight discovery as a benchmarkable capability, that being a highly open ended task is not easy to be assessed in a systematic way. In second place, it provides a synthetically generated and scalable benchmarking framework with ground truth insights embedded in data. And finally, it actually develops a benchmark not as a static scoreboard, but as part of a cycle of evaluation and optimization, with the optimal goal to enable agents to self improve.
A further motivation for this work lies in the disconnection between generative outputs and structured data: LLMs, even when augmented with retrieval methods, remain often ungrounded in enterprise data systems, making their outputs fluent but often unreliable.

In the majority of cases, a common evaluation approach is creating a reference benchmark and evaluating the agent against it, but how to create a benchmark in a setup where every response of the agent changes the customer response?
A static benchmark, even if custom built for the use case, would not do the job.
This approach is somehow a step toward a new conceptualization of AI, in a way that it naturally adapts to changes and understands new scenarios, more similar to how humans think.

Before going to the details of the project structure, it's important to explain the parallelism to the theoretical background that motivates this research.
Previously has been explained how by testing large reasoning models in controlled puzzle environments, the evaluation frameworks that only measure output accuracy give a distorted impression of reasoning ability: models appear competent at low or medium difficulty but fail under higher complexity, showing a thinking collapse by producing shorter and less coherent reasoning chains, precisely when tasks demand more effort.
This phenomenon reveals a fundamental evaluation failure, since exactly how it's done with standardized IQ tests for human intelligence, benchmarks that focus narrowly on correctness of responses create illusions of reasoning in models, masking fragility and preventing the detection of collapse under certain levels of empirical complexity.
SparkBeyond's framework is very useful also to respond exactly to this type of failure: by embedding explainability into the benchmark, the framework ensures that evaluation doesn't stop at correctness but also probes the coverage, coherence and adaptability of agent outputs. Just as in the pointed research, where controlled environments were used to show reasoning collapse, here it's possible to use synthetic benchmarks as a dynamically controlled environment to ensure that the illusion of thinking is not happening, where fluent sounding outputs may hide a lack of true discovery.

II. Methodology

In the context of AI and machine learning research, a benchmark represents more than a static dataset with which to compare the outcomes or performances of a model, but it works as a standardized test case or reference scenario created to

evaluate and compare the capabilities of systems under consistent conditions, with the goal to optimize them by providing a transparent and repeatable framework through which strengths and areas for improvement can be systematically identified.

The methodology adopted in this work is grounded in the principle that a benchmark shouldn't only provide standardized tasks for evaluation, but also capture the richness, variability and unpredictability of real world problems.
To achieve this, the starting point has been the definition of requirements, formalized as a collection of benchmark problems and datasets, accompanied by an evaluation framework capable of systematically assessing performance across diverse dimensions. Each problem functions as a unit of evaluation and is carefully designed to represent both the structure and the complexity that an agent would encounter in real world data exploration tasks.

An item in the benchmark corresponds to a problem specification, which integrates all the essential components needed to formulate the problem and assess its solution. The specification is defined through several key components: a unique problem name and its domain, the problem description to outline the business or analytical objective of the task, the explicit definition of the required tables alongside an indication of the primary one to which the core task is connected and the secondary tables, a target name representing the outcome variable linked to the underlying KPI, and finally, an explicit enumeration of the insights to be discovered, which form the ground truth reference for evaluation.
To ensure full transparency, have been appended three forms of commentary to each problem specification, whether on data, on the target, and on the structure, ensuring that the overall schema is well understood.
Once a problem specification is fully defined, the benchmark generation transforms it into concrete datasets and evaluation frameworks.

The strength of a benchmark doesn't lie in producing datasets, but in respecting three fundamental properties: richness and diversity, robustness and realism.
To be considered rich and diverse, a benchmark must cover a broad curriculum of problem domains and types, analyzing instances from multiple industries and contexts, and diverse target definition approaches.

Beyond domain and target heterogeneity, diversity is also captured in the difficulty of the problems themselves. Problems differ in the number of patterns to be discovered, the number of tables composing the schema, and the depth of the logical or statistical expressions that define the patterns, as well as from the computed empirical complexity. This metric is problem based, so if the two baseline Single-step and Two-iterations agents struggled or failed in finding a solution to it, it's assigned a score between 0 and 1 to measure its degree of complexity.
This ensures that agents are evaluated not only on simple tasks that to be solved require pattern matching, but also on problems that require multi step reasoning and the integration of several complex relationships.

The second principle of a strong benchmark is robustness, since it should however provide a clear ground truth against which the generations can be measured. This is the real difficulty when developing a benchmark that involves open ended answers like insight discovery ones, especially in how to build reliable ground truth solutions with outputs that can vary based on multiple variables. This is a critical difference with real world settings, where insights are often subjective, incomplete, or open to interpretation: by establishing ground truth, the benchmark ensures a degree of comparability of results across agents.
The framework must be capable of dynamically generating new benchmark problems that are similar in structure and complexity to an existing reference problem, and it's crucial since needs and capabilities evolve quickly and with them the types of problems that are relevant for evaluation.
This dynamicity should lead to an optimal auto improvement of agents; by continuously creating new problems that challenge specific weaknesses, the benchmark itself becomes a tool for agent training and refinement.

The third pillar is realism to ensure that the benchmark remains grounded to the practical challenges that motivate it. Realistic benchmarks are those that mirror actual relationships that occur naturally across data tables, to ensure that performance on synthetic tasks correlates with that in actual deployments.

This methodology is designed not only as a performance test but as a behavioral evaluation: it ensures that insight discovery is not achieved through simple pattern matching or statistical overfitting but through processes that mimic valid analytical

reasoning. By grounding the benchmark in synthetic but realistic datasets with embedded ground truth insights, it provides a controlled yet scalable environment where agent capabilities can be tested.
Moreover, by ensuring explainability, through explicit ground truths and clear evaluation metrics of each process step, the framework helps to diagnose where evaluation fails, demonstrating a sort of illusion of discovery that occurs when outputs are fluent but ungrounded, and leaving room for improvements.

a. Agents

For the purposes of this research, were developed and tested three proprietary agentic LLMs, each differing in their structure and in the degree of refinement applied to their generations: the Single-Step Agent, the Two-Iterations Agent, and the DP Agent. These models represent progressively greater levels of depth and autonomy. All three agents are built upon the Hypothesis Engine, which synthesizes vast amounts of data into candidate features and relationships, providing their foundational substrate. What distinguishes them, however, is not the underlying generative capacity of the Hypothesis Engine, but the agentic layer, implemented through internal frameworks such as Agentune, that governs how hypotheses are selected, validated, and iteratively improved. The key difference between the Single-Step, Two-Iterations, and DP Agents lies in the interaction with the Hypothesis Engine: a single pass, two successive refinements, or an open ended continuous cycle, with each stage marking a further step in adaptiveness and performance.

The single-step agent is the simplest one, providing a direct application of SparkBeyond's hypothesis generation capabilities. When an input problem is provided, the single-step agent calls on the Hypothesis Engine once, generates a set of plausible features or insights, and outputs them directly, with no further refinement of the generation. Its strengths lies in fact in its simplicity: it's fast, requires minimal computational resources, and is particularly effective when the problem is well defined or when actionable insights can be captured in only one pass. However, it has obvious limitations since real world problems are often noisy and with hidden dynamics, not possibly detectable by a one time generation of hypotheses, that may capture only clear patterns, risking to miss deeper relationships.

This is where the two-iterations agent emerges as a more sophisticated alternative: instead of stopping after the first pass, it introduces a second cycle of reasoning.
In the first iteration, it generates candidate hypotheses just as the single-step agent does, but instead of presenting these raw hypotheses as final, it performs a second pass to refine, filter, or rerank hypotheses, incorporate additional context, and resolve inconsistencies. This represents a fundamental evolution in agentic design, since the second iteration allows the agent to act almost as a peer reviewer of its own work, interrogating the initial outputs and correcting them before they reach the decision maker, and introducing a degree of self improvement.
This two-step architecture makes the agent both more cautious and more accurate, leading to insights that are better aligned with the business context.

Also the two-iterations model demonstrates clear limitations, since by restricting itself to just two cycles, it still operates within a bounded reasoning loop, and while it may correct some errors of the first iteration, it cannot sustain a dynamic open ended process of self improvement. The demand for these types of tasks is agents that continuously analyze their own performance, identify weaknesses, and adapt strategies over time, and here is where the DP agent emerges as the most strong one, being far more valuable than one that delivers a static set of recommendations.
The DP agent is not a formally branded product on SparkBeyond's public website, but it's closely associated with their Agentune framework and their broader articulation of Always-Optimized™ AI. It can be interpreted as shorthand for a dynamic decision planning agent, but in practice it signifies an agent that embodies a full iterative cycle of reasoning and improvement. Unlike the single-step and two-iterations agents, the DP agent does not stop after one or two passes, instead it's designed to enter into a self improving loop that includes four stages: analyze, improve, simulate, and deploy.

The first stage, analysis, requires the agent to assess its current performance against predefined KPIs, evaluating how well its current strategy meets these goals.
In the improvement stage, it generates modifications to its internal hypotheses or even strategies, again leveraging the Hypothesis Engine but guided by the feedback from the previous analysis.

The simulation stage allows the agent to test these modifications in a controlled environment, without risking real world failure. Only after validating improvements does the agent deploy the updated strategy into a live setting.
Importantly, once deployed, the agent continues to collect data on its real world performance, which feeds back into the next cycle of analysis, creating a loop of continuous optimization, and over time the DP agent becomes increasingly aligned with its KPIs.
This iterative process gives the DP agent an adaptability that the two baseline counterparts cannot match.
Its behavior mirrors the scientific method more closely than the other two: hypotheses are not only generated but continuously tested, refined, and tested again.
In a sense, the DP agent represents SparkBeyond's attempt to make in practice the idea of self improving AI: it's not enough to discover an insight once; the true value lies in building a system that never stops improving its insights in pursuit of defined objectives.

b. Dynamic benchmarking framework

The benchmarking framework is built as a dynamic curriculum of problems, designed to assess the ability of artificial systems to move from predictive modeling to the generation of human interpretable hypotheses grounded in data, or insight discovery problems.
Unlike static benchmarks, which rely on fixed datasets and quickly become vulnerable to memorization, this framework emphasizes continuous novelty through the automated generation of a dynamic framework, that rather than producing a series of isolated datasets, it creates a representative ecosystem of problems, each linked to a meaningful KPI and constructed at varying levels of empirical complexity, showing both the diversity and the challenges in real world analytic environments.

To provide the controllable data needed for such evaluation, SparkBeyond has developed a generative system named ProblemMaker, which synthetically builds hundreds of relational datasets, each designed to give an insight discovery challenge.
This synthetic design is crucial for building a controlled environment that is embedded with hidden rules and patterns, constituting the ground truth insights that

enable a consistent evaluation of results: in fact, being insight discovery an open ended task, to successfully evaluate and compare dynamic responses are needed ground truth data that, despite the exact generated words, allow evaluators to map each response back to the underlying statistical or logical relationships encoded in the dataset.

After the preparation of these complete problem specifications, data tables are actually built up, that will become the dataset for each problem. This includes creating primary and secondary tables that reflect the structure of real world operational data, complete with connection keys, that link tables together in meaningful ways simulating the structure of real world data environments where multiple sources must be joined and interpreted together.
In addition to standard columns, quantitative models representing the embedded information are also generated at this stage, and added to the data as enriching columns. These could include simple linear relationships, logic based on more complex threshold or combinatorial conditions covering multiple variables.
However, to make the final benchmark really challenging, these columns representing the information will be removed in a following stage, leaving only the raw data and the target column. This ensures that any insight the LLM discovers comes from the analysis of the remaining data, not from direct observation of those models in an explicit column, with agents that have to rediscover them later.
Data is then splitted into training and testing partitions, following the typical rule of a 70/30 ratio, and builds a rich metadata package that describes the schema, table relationships, hidden insight definitions, and indexing for evaluation purposes.

The third step is target creation, where the target column for the primary table is generated so that it strongly correlates with a linear or logical combination of the newly created insight columns. The idea is to ensure that this information is not just hypothetical, but actually drives the target, exactly as real world KPIs are affected by factors hidden in business data. For example, the target could be calculated as a weighted sum of several insight variables or as a logical threshold condition that depends on multiple variables.
This approach ensures that the information is not only identifiable, but also really important for the target prediction.

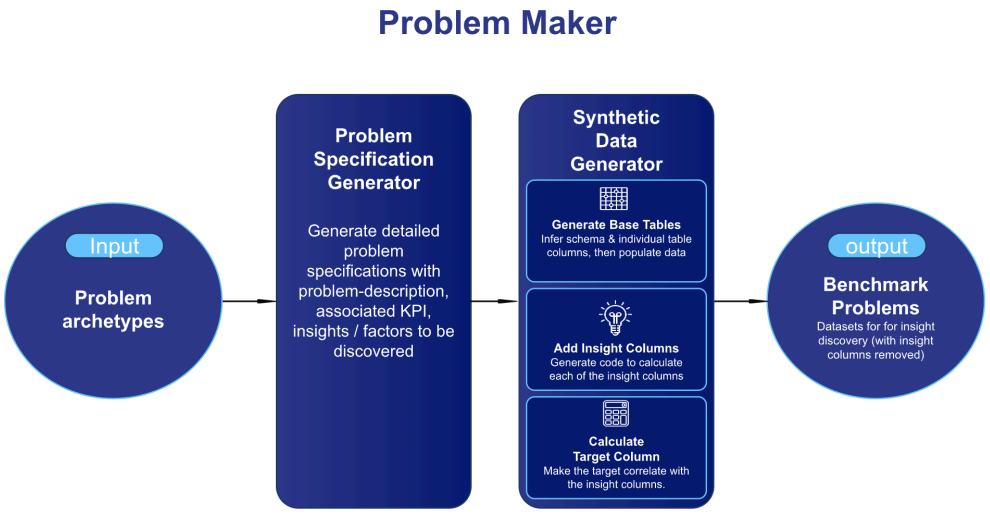
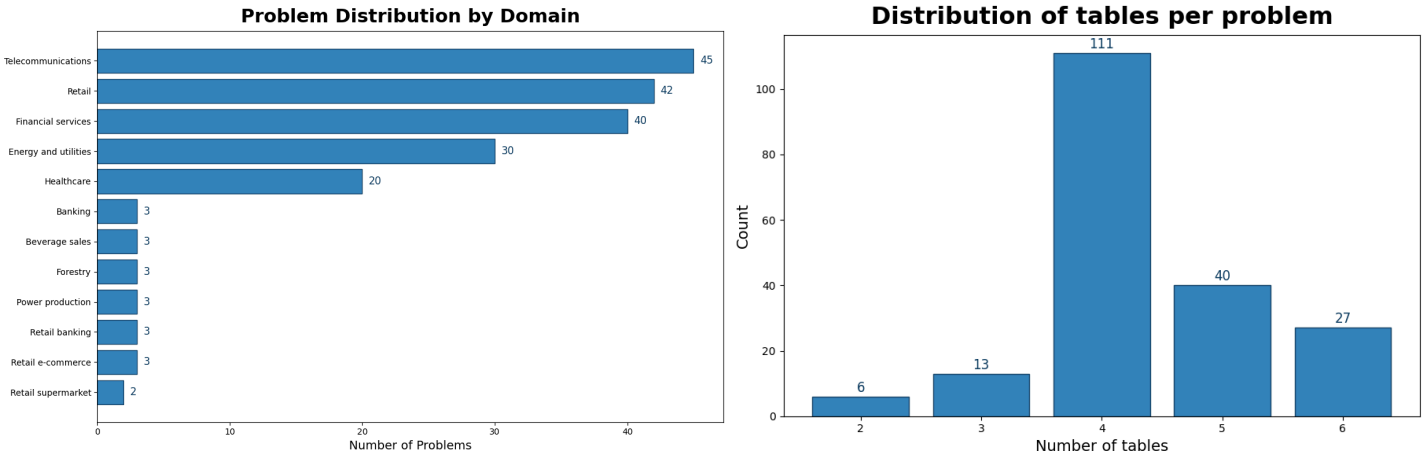


Figure 4.5: Complete Problem Maker Pipeline, showing how synthetic generated problems are created before deployment in the evaluation framework. Source: SparkBeyond.

It's important to note that not all intuitions planned during the specification phase can always be perfectly realised in synthetic data. In some cases, it may not be possible to generate data that accurately reflects the specified insight pattern or the correlation between the insights and the target, which may be weak.
This is a natural result of working with complex synthetic generation and reflects real data modeling challenges, where some signals may not be as strong or clear as initially expected.

With the release of the 1.1 version of the Benchmark, are included 197 curated problems, each with high quality ground truth data and enriched metadata, varying across multiple domains, number of tables, insights, and empirical complexity, making the benchmark significantly more comprehensive and realistic.
Each problem could appear in multiple variations depending on how the target has been generated: targets were built through either linear or logistic models.
Within each approach the correlation between predictors and the outcome was tuned according to three regimes: weakly biased, balanced, and strongly biased, allowing not only to enrich the dataset, but also to adjust the empirical complexity and bias conditions of the task in a controlled way, while keeping the same underlying data.
Each variation of the problems was named following a conventional schema to ensure transparency in how targets were derived:
$P[\#] - [problem\ name] - variation[\#] - type[\#][a|b]$

The complete benchmark curriculum touches multiple dimensions of empirical complexity, domains, and structures, as shown by the distributional analysis of problems (Fig. 4.6): these were not randomly distributed, but deliberately spanned across a broad range of categories (a), while structural complexity varied with the number of tables, columns and implicit relationships (b).



a) Problem distribution by domain.

b) Distribution of the number of tables per Problem.

Figure 4.6: The generated problems go from domains (b) rich in data such as telecommunications, retail, financial services, energy, and healthcare, but also including niche sectors like forestry and retails. The number of tables per problem (b) reveals that the majority use four or more tables, replicating the relational complexity of real world enterprise data. Source: SparkBeyond.

To ensure that the benchmark does not only test performance on a typical average case, but instead provides coverage across tasks of several levels of challenge, the distribution of problems intentionally captures a long spectrum of problem types and associated KPIs, from simpler to highly complex ones.

This avoids an excessive focus on the same classes, and instead produces not only individual datasets, but a representative ecosystem of tasks that reflect the diversity of operational key performance indicators that real world organizations seek to optimize.

As already explained, the benchmark is dynamic and scalable, and since the data generation process is automated, systems can continuously produce new problems with varying structures and complexity, ensuring that the benchmark will not saturate or become obsolete, while the inclusion of explicit ground truth insights guarantees that each task remains verifiable and reproducible. In this way, it represents a shift

away from static evaluation datasets toward a living framework that mirrors the adaptability and variability of real world scenarios, as well as its complexity.

c. Evaluation Framework

To evaluate an agent's performance on a benchmarking problem, it's essential to employ an evaluation framework that captures performance in a comprehensive way by defining the dimensions through which evaluation takes place, and assessing whether the insights discovered align with the ground truth solution, measure their predictive power, and verify the proper use of data.

Therefore, given a set of ground truth and agent discovered insights, the evaluation solutions will be based on three critical dimensions:

- Coverage: how well the set of insights discovered by a candidate agent cover the set of ground truth insights.
- Statistical power: how good is a predictive model trained (with a standard classifier), using the proposed insights as features in predicting the target on the test set.
- Target leak: happens when a model gains access to info during training that it wouldn't realistically have in real world deployment (artificially high performance but poor real world generalization); for example when the temporal data is provided, is the agent respecting the temporal constraints to avoid use of data that wouldn't be available at time of inference?

Primarily, in the first beta version of the benchmark, it has been observed that Correlation Coverage doesn't always capture the full extent to which ground truth insights are reflected in the agent's solution. Specifically, when a discovered insight reproduces the essence of a ground truth feature but does so via an inverse relation, no correlational metric would adequately reflect this relationship. This makes it possible to evaluate whether an agent's output engages with the same variables and relationships encoded in the benchmark specification, even if they are expressed approximately, with the metric that doesn't simply test if an agent can identify one or two isolated patterns, but whether it can systematically reconstruct the broader insight space embedded in the problem. Therefore, several additional metrics were

explored (Fig. 4.7).

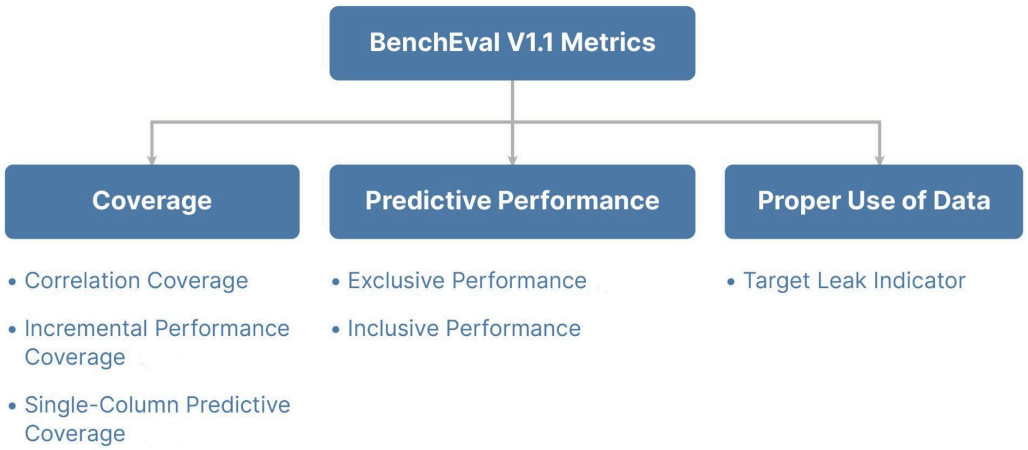


Figure 4.7: Metrics employed in the benchmark to assess how well agents recover ground truth insights, the strength of their predictive power, and safeguard against target leakage.
Source: SparkBeyond.

i. Coverage Metrics

They assess the agent's ability to discover insights regarding factors that affect the target KPI, directly measuring the relationship between automated discoveries and expert knowledge, rather than assuming that high predictive performance indicates good insight discovery.

This evaluation assesses semantic alignment with domain expert knowledge through four distinct approaches:

- Incremental Performance Coverage: it quantifies the extent to which each ground truth insight is "covered" by the insight identified by the solving agent. It measures the marginal predictive performance that could have been gained by explicitly adding ground truth features to the agent's solution.
  With c as ground truth insight column, S as the set of insight columns provided by the agent, the incremental Performance Coverage of c given S is:
  $1 - max(\rho(Performance(Model(S \cup \{c\})) - Performance(Model(S))), 0)$
  with $\rho(x) = 2 \times max(x - 0.5, 0)$ as a threshold function that discards low quality performance values below 0.5.
  If adding ground truth column c to the agent's insight set S results in measurable improvement in predictive performance (after thresholding), then solution S is missing that potential improvement, resulting in lower

Incremental Performance Coverage for c. Contrarily, if adding c doesn't improve performance, the column is considered as covered, gaining a coverage score of 1.

- Single-Column Predictive Coverage: the Predictive Coverage metric quantifies how well each ground truth factor can be recovered from the insight columns provided by the agent, capturing information not reflected by incremental performance coverage, as it focuses on predicting ground truth columns rather than the final target. The Single Column Predictive Coverage extends this concept by forcing the reconstruction to use only individual discovered features rather than all features combined. Due to its stricter constraint, this metric is generally more challenging than its all column counterpart. The motivation behind this adoption is in terms of: interpretability, since if each highly influential ground truth column c can be predicted with high accuracy from a single solution insight column $s \in S$, then the semantic relationships uncovered by the agent are more readily interpretable; and actionable insights, since in real world scenarios users can control or influence input features, understanding that feature s applies its influence on the target through a mediating factor c (present in ground truth) enables actionable intervention.
  Given a set of solution columns S and ground truth insight column c, the Single Column Predictive Coverage of c is:
  $max\ \rho(Performance(Model(\{s\})\ predicting\ c))$, for $s \in S$
- Correlation Coverage: it serves as a computationally efficient analogue to Single Column Predictive Coverage, but rather than training predictive models, it evaluates Spearman correlation between ground truth and solution columns. While high correlation indicates that solution columns are informative with respect to ground truth columns, the converse doesn't necessarily hold: low correlation does not imply lack of predictive information, particularly when relationships are inverse and therefore not captured by Spearman correlation, showing limitations in capturing complex relationships.
  Given a set of solution insight columns S and a ground truth insight column c, the Correlation Coverage of c is:
  $max\ SpearmanCorr(c, s)$, for $s \in S$

Overall, to provide a unified per problem metric that balances different aspects of semantic alignment, has been computed a Combined Coverage Score, that through a calibrated weighting of 30% to Incremental Performance Coverage (IPC) and 70% to Single Column Predictive Coverage (SCPC), reflects the framework's emphasis on explainability while keeping predictive validity:

$$Combined\ Coverage = 0.3 \times IPC + 0.7 \times SCPC$$

ii. Performance Metrics

Performance metrics assess the predictive capability of agent generated insights using multiple baselines to understand their added value.

- Inclusive Performance: it measures the predictive power using both the original problem features and the agent generated insight columns, representing real world deployment scenarios. Training data consists of the union of base columns and insight columns, model evaluation is made on test data target values, and RandomForest model is used with consistent hyperparameters across all evaluations, overall reflecting the maximum predictive capability when combining human engineered features with automated discoveries:
  $Performance(Model(base\ features\ +\ agent\ features) \rightarrow target)$
- Exclusive Performance: it measures predictive capability using only the agent generated insight columns, excluding original problem features. A capable agent should identify and extract the base columns that are informative for predicting the target, effectively prompting them to insight columns. This is particularly important when predictive signals are explicit and don't require complex reasoning. Overall it tests whether the agent can independently discover all relevant patterns without relying on pre engineered features.
  $Performance(Model(agent\ features) \rightarrow target)$
- Naive Performance: it provides a baseline using only the original problem features, enabling assessment of the added value from automated insight discovery. Overall it serves as a comparison baseline for measuring the real improvement achieved from the model.
  $Performance(Model(base\ features) \rightarrow target)$

iii. Proper Data Usage Validation

To ensure that insights are valid for real world deployment, the framework implements comprehensive target leakage detection, that occurs when feature engineering functions inadvertently use information that wouldn't be available at prediction time, particularly in temporal scenarios.

This is checked through three strategies: statistic analysis, that examines function source code for direct target variable references to identify and flag suspicious data access patterns; dynamic testing, executing functions under controlled conditions with target information masked, comparing outputs before and after target masking to detect functions whose behavior changes when target access is removed; and temporal validation, ensuring feature functions respect temporal constraints, preventing use of future information relative to prediction time.

It's considered as a binary value, 1 if it's detected and 0 if not.

To provide an overall evaluation that integrates all the dimensions discussed so far, is computed a Combined Score is computed to give a comprehensive assessment of an LLM agent's performance, capturing not only whether the final outputs are accurate or correct, but also how those outputs were produced and to what extent they reflect systematic reasoning rather than superficial pattern matching.

Unlike traditional benchmarks that rely on a single metric, such as final answer accuracy, this combined approach prevents the evaluation from being skewed by one isolated dimension. Accuracy alone, for example, cannot reveal whether the agent used data responsibly, whether its reasoning traces were coherent, or whether its insights aligned systematically with the ground truth space, and by integrating multiple complementary metrics into a unified score, the benchmark reduces the risk of overfitting evaluation to narrow criteria and instead balances correctness with coherence and generalization.

Importantly, this is not another single metric: the Combined Score isn't a substitute for the underlying dimensions but rather a synthesis that ensures no aspect of the evaluation is ignored, preserving the nuance of the multi dimensional evaluation framework.

$$Combined\ Score = 0.5 \times Inclusive\ Performance + 0.5 \times Coverage - 1.0 \times Target\ Leak\ Penalty$$

III. Results

The evaluation of agentic LLM architectures through the proposed SparkBeyond benchmark enables a systematic analysis of how different reasoning strategies perform under controlled and comparable conditions.
This approach makes it possible to draw conclusions not only about relative performance, but also about the underlying behaviors of the agents, their strengths, and their limitations, provided that the results are carefully interpreted through the lens of the multidimensional metrics that shape the entire research. In doing so, the benchmark reflects the spirit of modern LLM testing, moving beyond single dimension evaluations focused narrowly on accuracy and instead capturing the overall quality of discovery.
This allows to interpret not just whether an agent succeeds, but how it succeeds, or struggles, across a diverse set of problems, offering a deeper perspective on what reasoning strategies are genuinely effective and where their limits lie.

Moreover, the real value of this benchmarking framework is proved when comparing different agentic architectures: by evaluating the performance of the three proposed DP, Single-step and Two-iteration agents, it becomes possible to identify how iterative reasoning strategies translate into measurable differences across complexity levels and evaluation metrics.
Within this context, despite producing the least number of valid solutions (156 out of 197), the DP agent emerges as the strongest performer, demonstrating both its capacity for insight discovery and the robustness of its reasoning strategy under the multidimensional evaluation framework

The combined score previously computed immediately reflects how well an agent balances all the dimensions relevant for a strong agent trying to successfully discover and generate insights.
The Single-Step Agent achieved a combined score of 0.58, the Two-Iteration Agent scored 0.65, while the DP Agent resulted as the strongest with a score of 0.72, clearly demonstrating that the last one could be seen as the strongest.
These results show clear implications, with agents that employ no iteration performing the weakest, while adding a second iteration introduces a noticeable improvement.

This is the central meaning of the benchmark, confirming that evaluation requires a comprehensive and integrative metric, with no single dimension that is sufficient to capture the multifaceted nature of each performance. The combined score reveals that the DP Agent is not just slightly better in some areas, but is the most balanced architecture tested, excelling across multiple dimensions simultaneously.

But what really contributes to these differences? Looking more closely at coverage metrics allows for a deeper understanding of why these combined scores diverge so much: coverage measures the extent to which an agent recovers the ground truth insight space embedded in the synthetic problems.
The Single-Step Agent achieved a coverage score of only 0.42, the lowest among all, further demonstrating the limits of one shot generation. Without the opportunity to refine or validate hypotheses, the agent tends to be stuck on superficial correlations, that while appearing predictive can fail to reflect the hidden drivers of the problems. As a result, large parts of the ground truth insight space remain unrecovered, leaving gaps that are determinant in the agent's performance.
The two-iteration agent improved to 0.52, showing that an additional cycle of reasoning allows for a small expansion of coverage, but despite this partial improvement, without structured validation steps the second iteration frequently leads to redundant or noisy hypotheses, rather than systematically filling the missing gaps.
The DP Agent, by contrast, achieved a coverage score of 0.61, substantially higher than both: this shows that its discovery process is not only broader but also more coherent with the underlying structure of the problems. For DP, the minimum IPC was 0.82, meaning that adding any ground truth feature to its solution yielded almost no additional predictive advantage, a strong sign that the agent had already almost effectively captured those relationships through its discovered features.
The Single-step, by comparison, left much more room for improvement: adding ground truth features often gave substantial performance gains, reflecting its failure to reconstruct important drivers. The Two-iteration narrowed this gap in some way, but still shows weaknesses, with many ground truth features not fully accounted for.

Another point of view comes from the single-column predictive coverage, which measures whether features discovered individually can predict ground truth columns independently.

The DP agent's score of 0.55 suggests that more than half of its discovered features are directly related and in an interpretable way to underlying drivers, a crucial aspect for their usability. In fact in real world scenarios like the ones simulated through synthetic data, decision makers benefit most from insights that can be explained in terms of single and clearly interpretable variables.
The Single-Step agent scored significantly lower (0.39), indicating that roughly only one third of the instances directly align with individual ground truth columns.
The Two-iteration agent again performed somewhat better than the Single-step (0.48), still falling short of the DP agent, probably since its iterative reasoning produced more features that yet failed to ensure their clarity.
These results highlight a key strength of the DP agent: its iterative discovery process not only expands coverage but also filters features in ways that preserve interpretability.
Predictive coverage metrics further reinforce this picture: DP agent achieved a mean predictive coverage of 0.66, while Single-step and Two-iteration agents gained significantly lower values, indicating that their discovered features carry less genuine predictive signals.
The quality of generated features is further highlighted by the minimum incremental performance coverage: for the DP Agent, even the weakest features contributed meaningful predictive value, as reflected in its high IPC. The Single-Step Agent, however, often produced empty features that added no predictive power, while the Two-Iteration Agent reduced but did not eliminate this issue. This ability of DP Agent to minimize noise proves that its discovery process has effective mechanisms for filtering out weak hypotheses.

Performance metrics provide yet another perspective: a huge disparity suggests that the features are not independently strong but need some incremental value when combined with base features. By achieving almost identical inclusive and exclusive scores, an agent proves that its insights are not auxiliary but stand on their own. This robustness is a strong characteristic of effective insight discovery, since in many practical contexts agents must identify predictive patterns without relying on pre existing engineered features.
In fact, Single-step agent achieved almost the same value at inclusive performance (0.73) and exclusive performance (0.72), showing that while it could contribute to

predictive accuracy when combined with base features, its insights alone were insufficient to independently reconstruct the predictive drivers.
The Two-iteration agent performed better, with both inclusive and exclusive performance around 0.78. Still, adding a second iteration improved its ability to act independently, but the gap between its exclusive performance and the ground truth benchmark (0.90) is wide.
The DP agent, however, achieved both inclusive and exclusive performance scores of around 0.82, showing that the agent's discovered features effectively reconstruct the true drivers embedded within the problems, setting it apart from the other agents. Moreover, the DP Agent's exclusive performance almost approaches the ground truth level (0.89), indicating that while it's not perfect, it comes closer to full reconstruction than any of the alternatives.

On the dimension of proper data usage, all three agents performed equally in one sense: none exhibited target leakage. This is comforting, as it confirms that the benchmark's checks are effective and that none of the results are artifacts of unrealistic shortcuts. However, leakage prevention, while necessary, is not sufficient for distinguishing agent quality.

When these results are considered together, a clear explanation emerges for why the DP Agent outperforms.
The Single-Step Agent illustrates the limitations of one shot reasoning: it can identify some predictive patterns but fails to reconstruct the insight space systematically, leaving many ground truth drivers uncovered and producing a substantial proportion of weak or irrelevant features.
The Two-Iteration Agent shows that iteration improves performance, as expected, but iteration without structured validation is insufficient. It generates more features, some of which are predictive, but it also produces redundancy and lacks mechanisms to filter, resulting in less interpretable and balanced solutions.
The DP Agent, by contrast, integrates iteration with validation, reconstructing more of the ground truth insight space and achieving higher coverage.
Its inclusive and exclusive performance values confirm that it can function robustly both with and without base features, demonstrating independence as well as complementarity, and its balanced combined score further confirm this superiority,

showing that it doesn't excel in one area at the expense of others but achieves consistently high results across the majority.

The implications are significant: in real world analytic scenarios, decision makers require insights that are not only predictive but also interpretable and trustworthy. High predictive accuracy in the absence of coverage can mask critical gaps in understanding, whereas coverage without predictive strength produces insights that lack practical applicability. The DP Agent demonstrates that it's possible to achieve both, making it the most suitable candidate for deployment in contexts where analytic reliability and interpretability are critical.

The combined score, which balances inclusive performance and coverage while penalizing leakage, shows a clear and direct deterioration as empirical complexity rises when we look at the aggregate across agents, demonstrating the structural pressure it exerts on all agents.
On very easy tasks, labeled with empirical complexity 0, the Two-iteration agent actually shows the highest combined score (0.83) among all the others, with the single-step and DP agents both scoring 0.78, but as empirical complexity increases to easy and medium, the DP agent make a jump and surpass the other two, and when approaching from hard to very hard levels, every agent drops substantially, with the DP keeping a more graceful trajectory than the others (Fig. 4.8).

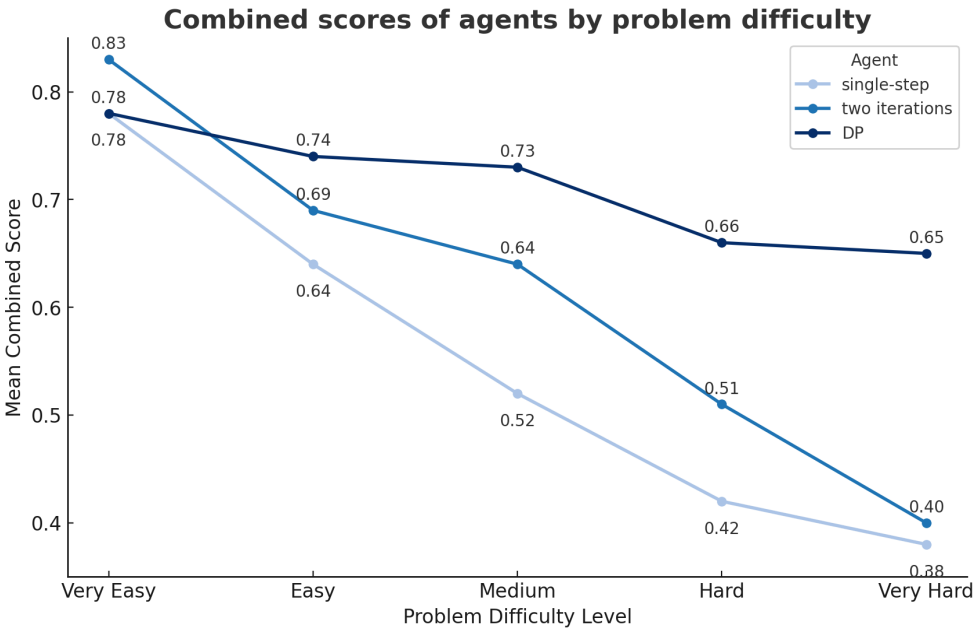


Figure 4.8: The trajectory of combined scores by agent shows how agent performance declines as problem empirical complexity increases, with all agents losing performance, but the DP agent remains far more resilient than the counterparts.

At a very hard level of problem empirical complexity, DP still scores 0.65, while the single-step and two-iteration fall to 0.38 and 0.40 respectively.

Those results reinforce what, yet theoretically, proved by the three complexity regimes highlighted in the illusion of thinking [55]: standard LLMs can match or beat large reasoning models (in this case represented by the DP agent) on low complexity items, but beyond a specific complexity threshold all models suffer a form of collapse.

The DP agent's main advantage is not its very high performance with easy problems, but instead its robustness in decline. From very easy to very hard, its combined score drops by roughly 15%, while single-step and two-iteration experience also a decrease by more than 40%.

Coverage metrics make this difference even more concrete, since at very easy empirical complexity, all three tested agents show a healthy mean correlation coverage score, with DP and single-step at 0.7, while two-iteration even at 0.75, still showing greater performance than the counterparts on easy tasks.

As empirical complexity rises to very hard, the single-step and two-iteration agents' coverage collapses to less than 0.1, with over 70% loss for each, and DP scoring 0.54 (Fig. 4.9). This pattern is even more marked in single-column predictive coverage:

DP drops from 0.7 to 0.35 (about 50% down), yet single-step agent drops from 0.76 to 0.06 and two-iteration from 0.78 to 0.04 (more than 90% down respectively). In other words, as problems become more complex, the simpler agent can no longer produce individually meaningful features that map to ground truth columns, surviving only through weak combinations. DP, by contrast, still keeps a base of single-feature interpretability even at the highest levels of empirical complexity, showing once again its robustness. Beyond a certain complexity threshold, models generally abandon principled search in favor of simple approaches such as early stopping, yet the distinction here is that a discovery process agent resists this collapse far more effectively than an agent relying only on generation.

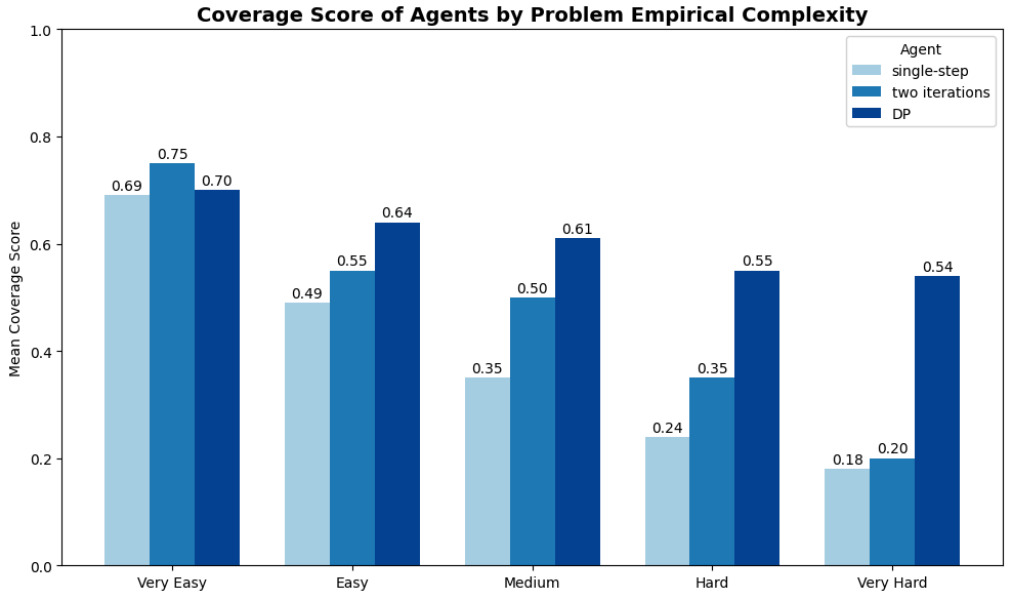


Figure 4.9: Coverage score of agents by problem empirical complexity, showing that while the baseline agents (Single-step and Two-iterations) experience a strong decline as complexity increases, the DP agent maintains comparatively high coverage even at the hardest levels.

The minimum incremental performance coverage, or in other terms the check for whether the intended coverage space has in fact been addressed, tells a similar story. At a very easy level, all three agents look strong (DP at 0.94, single-step at 0.93, and two-iteration at 0.96), but at the very hard level, DP has a relatively smaller decline (only about 20% compared to a 60% for the counterparts). When this metric stays high, it means that adding any missing ground truth feature gives almost no marginal predictive gain, so the agent already captured the essence. The DP agent guarantees it also into the hard regime, but the other two don't: by very hard level, their

performance improves only when ground truth features are reintroduced, revealing how incomplete their reconstructions have become.
Also predictive coverage reinforces these conclusions: mean predictive coverage falls with empirical complexity for every agent, but the trajectories are again radically different (DP with 22% compared to an average of 70%), indicating that as problem structure gets harder, the quality of explanatory signal in discovered features decreases.

Looking at inclusive and exclusive performance further clarifies how structural weakness in coverage translates into predictive fragility. At the very easy level, the Two-iteration agent again holds a slight advantage, achieving scores of 0.90 on both inclusive and exclusive performance, while the Single-step and DP agents scores only marginally lower, but as empirical complexity rises, DP's inclusive/exclusive curves nearly overlay each other at each level, showing that its discovered features stand on their own rather than relying on the original base features.
The single-step and two-iteration agents, however, not only lose more ground overall but also show a strong dependence on base features as empirical complexity grows, with exclusive performance eroding faster than inclusive performance, and by very hard, their exclusives are both slightly above 0.5.

Going back to the combined score, it clarifies why DP wins by a margin that grows with empirical complexity: this score is not simply a reward for raw accuracy, but it harmonizes the core dimensions reliably and penalizes leakage.
Because DP sustains interpretable single column alignment (SCPC), preserves marginal coverage (IPC) also in the hard regime, and holds exclusive performance close to inclusive performance, it accumulates advantages on every variable that the combined score calibrates.
The two baseline counterparts, in contrast, suffer SCPC and IPC as soon as relationships are no longer linear, which decreases their combined score faster, even when inclusive performance remains sufficient for a while thanks to the base features.

IV. Beyond Statistics: semantic coverage and qualitative assessment

While the introduction of the evaluation framework in this benchmark represents a significant advancement for evaluations of LLM agents, it remains incomplete in one crucial dimension: the semantic quality of the insights themselves.
The existing metrics, like coverage, predictive performance, and proper use of data, offer a strong statistical framework for quantifying whether an agent identifies patterns aligned with the ground truth features, if those patterns exhibit predictive power, and if the data has been used appropriately. Yet, these measures remain insufficient to capture the semantic relevance and qualitative interpretability of the generated insights, which are essential to assess whether outputs are meaningful and usable in real world analytic contexts.
The notion of semantic coverage directly addresses this gap: unlike statistical coverage, which is grounded in overlap with ground truth variables, it evaluates how closely the agent's natural language insights align with the meaning of the ground truth solutions.
This dimension has been assessed through an LLM-as-a-judge approach, by prompting GPT-5, GPT-4o and Claude-4 to evaluate the quality and relevance of the generated insights.
Unlike the previously discussed metrics, this approach uses the advanced linguistic capabilities of large language models to recognize semantic proximity, understanding the nuances behind sentences even when the words are different, an aspect where statistically based metrics would fail.

Thanks to this qualitative evaluation, it was possible to perform a dual analysis of performance. First, LLM-as-a-judge assessed whether the generated features were derived in strict alignment with the problem description provided to each model before finding a solution, and second it evaluated how closely these features semantically approximated the hidden ground truth features, which had been embedded during dataset generation but hidden from the models to ensure an independent discovery.
This stage is particularly important since statistical metrics alone cannot determine whether the generated insights are clear, relevant for who may use them, and coherent with one another. In other words, qualitative assessment is necessary to assess whether they are suitable for being used by decision makers for the optimal goal of insights, that is generating value.

To understand this, a qualitative analysis has been conducted on 10 problems selected for their relevance to the evaluation, particularly based on their empirical complexity and bias conditions. The chosen problems vary among multiple domains, including Retail, Financial Services, Energy and Utilities, Telecommunications, and Healthcare, and two instances have been chosen for each level of empirical complexity, labeled as: very easy (0), easy (1), medium (2), hard (3) and very hard (4). Each instance represents a variation of a main problem, carefully selected to preserve the high degree of variability that the benchmarking framework is able to guarantee.
It's important to recall that a problem is classified as empirically complex when the baseline agents (single-step and two-iterations) either struggled or failed to identify a valid solution. In such cases, complexity is quantified by assigning a score between 0 and 1 to reflect the degree of difficulty encountered.

Returning to the analysis, by prompting the mentioned large language models has been possible to classify the generated and humanized features in three categories, all interpretable through a humanized explanation of the decisions. The categories were defined as follows based on their relevancy with the problem description: relevant features capture behavioral or contextual signals that logically contribute to the target prediction; redundant features show weak or indirect correlation, often repeating low impactful data; and contrasting features include opaque thresholds, extreme numerical values, or over engineered logic that lack interpretability.
Among the models, GPT-5 provided the most convincing and critical labels, demonstrating a deeper sensitivity to nuances since, in many instances, it assigned a lower relevance score than its counterparts. This stronger performance can be attributed to GPT-5's more advanced contextual reasoning, which allows it to evaluate not only lexical overlap but also the latent meaning and logical coherence of the features in relation to the problem. By contrast, the key difference focused on how much the topic expressed by the generated feature was directly related to solving the problem description: in many cases, GPT-4o was able to correctly label features when their topic was strictly aligned with solving the problem descriptions, not in terms of sharing the same words, since domain and meaning were sufficient. Claude-4's judgments were less consistent, appearing more dependent on the immediate prompt and the surface level relation between a feature and the problem

description, often oscillating between GPT-4o's lack of strictness and GPT-5's rigorous evaluations.
What distinguished GPT-5 was its ability to critically assess context, successfully uncovering weak or hidden relationships between the generated features and the problem descriptions. For this reason, GPT-5's classifications were selected as the primary source of qualitative evaluation in this analysis, given its superior contextual understanding and its status as the most recent and advanced model among those tested.

The analysis of the generated features did not reveal a direct relationship between a problem's empirical complexity and the degree of relevance of its features: for instance, the problem P31-1, "Product Return Likelihood", categorized as very easy, produced a large majority of redundant features, those that were unclear, repetitive, or added little beyond insights already established (Fig. 4.10).
Conversely, the problem P45-1, "Chronic Disease Progression", with the goal to predict the progression of chronic illnesses from patient data and labeled as an easy problem for the agent, had the highest number of relevant features (17), alongside only two redundant and one contrasting.
These findings suggest that, while empirical complexity may influence correctness and statistical coverage, semantic quality depends more strongly on the agent's generative coherence and contextual reasoning than on the structural difficulty of the dataset itself.

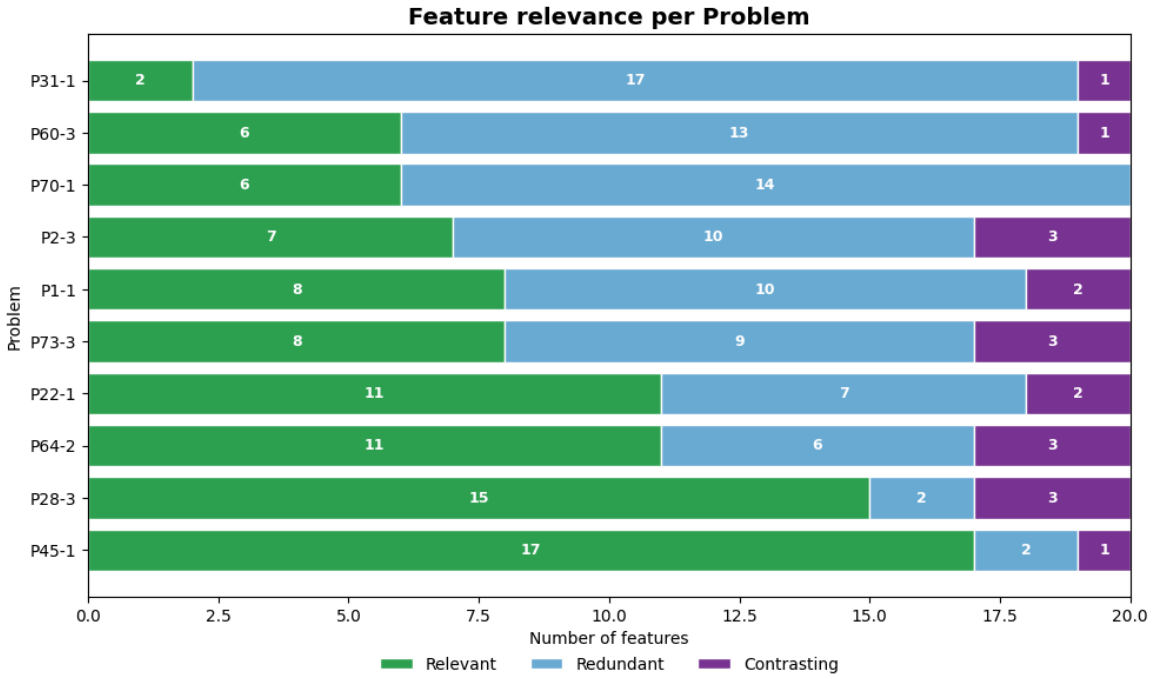


Figure 4.10: The distribution of feature types per problem shows that there is not a direct relationship between problem difficulty and generation of relevant features, with very easy and easily labeled problems that are producing a majority of redundant generated features.

Apart from the already statistically considered drivers, such as the number of tables per problem, the number of embedded insights, or the ratio of categorical to numerical variables, the question arises of how, from a qualitative perspective, one can determine whether the process has been truly successful. In other words, has the agent been able to generate feature solutions that meaningfully represent the ground truth?

It is worth recalling that, unlike the problem description, which is provided to the LLM and is used as the basis for generating the synthetic datasets, the ground truth features are initially embedded in the datasets as enriched columns but subsequently removed, ensuring that the agent must discover the relevant features independently and without prior knowledge. To evaluate this, the GPT-5 model was given both the ground truth features and the generated solutions for all problem instances, and was prompted to classify the generated features according to whether they matched one of the hidden ground truth features, making it possible to assess whether the agent pursued a similar reasoning path to solve the problem.

It is important to specify that, as the statistical metrics, particularly coverage, have already shown, every tested problem reached a solution to some extent. The aim of this qualitative analysis, however, is to move beyond this certainty and examine the

practical utility of the generated solutions. The focus here lies on whether the generated insights are interpretable and clear, qualities that are indispensable for insights to be easily interpretable and valuable in real decision making contexts.

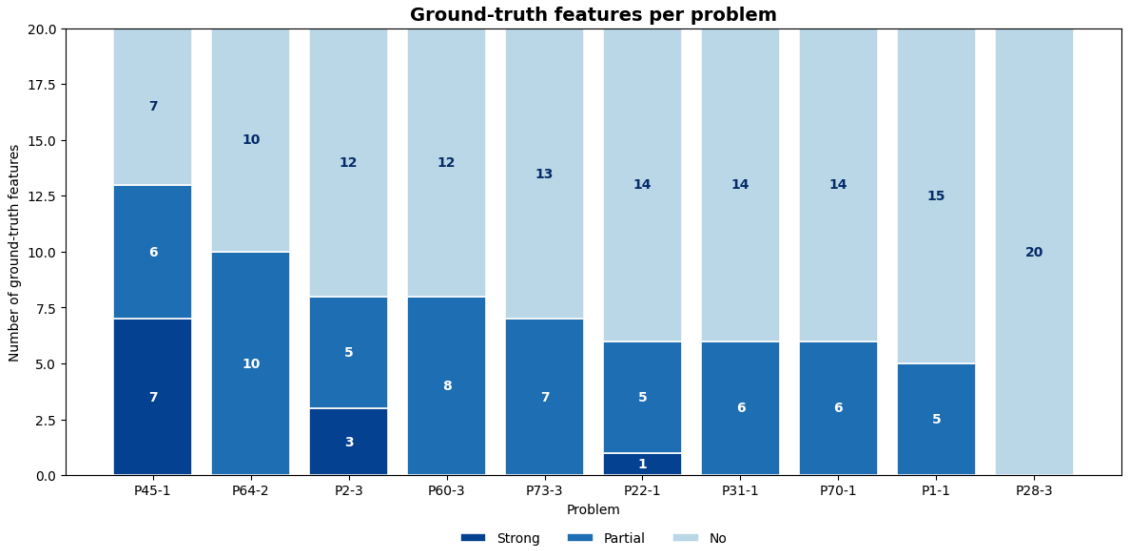


Figure 4.11: Ground truth features per problem, illustrating how generated solutions aligned with ground truth features across instances. The variability observed suggests that alignment is influenced less by empirical complexity and more by the agent's contextual reasoning and coherence.

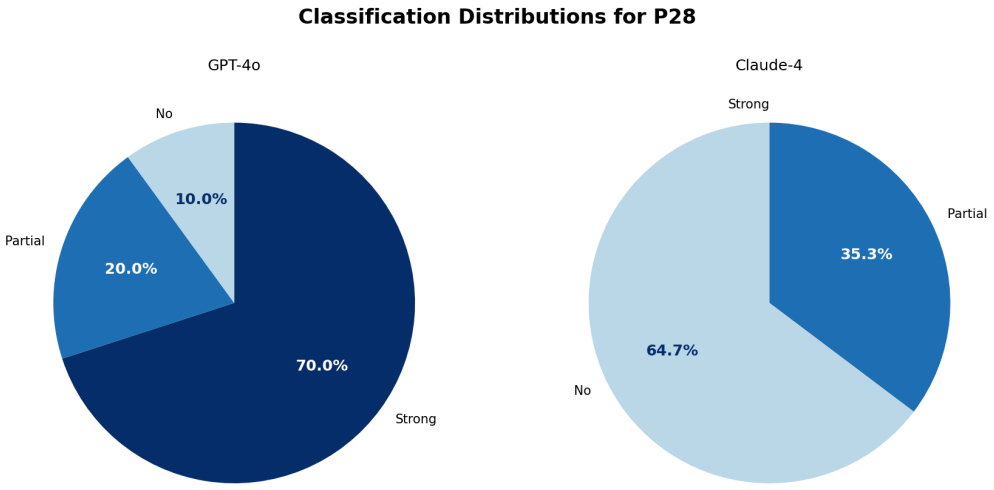
By semantically comparing the ground truth features per problem variation with the generated feature solutions using GPT-5, several interesting patterns emerge: problems with a higher proportion of relevant features (Fig. 4.10) also tend to show stronger alignment with ground truth features (Fig. 4.11). For example, P45-1 shows both the highest number of relevant features (17) and the highest number of strong matches (7), and similarly P64-2, with 11 relevant features, achieved 10 partial matches, still a considerably higher alignment than most other problems, suggesting that features classified as relevant often translate into stronger or partial semantic matches with the ground truth.

It is important to clarify that the categories redundant and partial are not intended to report errors: redundant features capture insights already expressed elsewhere, while partial features reflect incomplete but directionally correct alignment with ground truth solutions. Only contrasting features are treated as misclassifications, given their opaque thresholds, extreme values, or lack of interpretability.

The relationship between redundancy/contrast and alignment is also visible: problems with many redundant or contrasting features in Fig. 4.10 correspondingly displays large "No match" segments in Fig. 4.11. For instance, P31-1 and P2-3 both

contain significant blocks of redundant and contrasting features, yet in Fig. 4.11 they register only a few strong/partial matches (6 and 8, respectively) and many unmatched features (14 each). In the same way, P73-3 contains three contrasting features, corresponding to a high number of unmatched ground truth features (13), indicating that redundant or contrasting solutions rarely align semantically with ground truth features, therefore increasing the "No match" count.

An especially significant case is P28-3: despite showing a large proportion of relevant features relative to its problem description, with very few redundant or contrasting ones, GPT-5 consistently identified no matches with the ground truth features. After rerunning the evaluation, the results were replicated, suggesting that while the generated features were relevant in context, they diverged semantically from the hidden ground truth. To investigate further, the same solutions were assessed with GPT-4o and Claude-4, which produced markedly different classifications (Fig. 4.12), highlighting the variability among LLM-as-a-judge models.



a) GPT-4o: Predominance of Strong Matches b) Claude-4: Majority No Matches

Figure 4.12: Classification distributions for P28 across models. (a) GPT-4o shows a predominance of strong matches (70%), while (b) Claude-4 reports mainly no matches (64.7%), highlighting the contrasting evaluation styles of the two models.

The GPT-4o based model surprisingly classified the vast majority of generated features as strongly aligned with the ground truth, with only one out of ten instances receiving a "no match." By contrast, Claude-4 classified approximately one third of features as partial matches, consistent with its behavior across other evaluations.

What is impressive, however, is the wide discrepancy between GPT-4o and GPT-5 in their classifications: at a first look this might appear to be an error, yet after both human evaluation and repeated testing, it became clear that the divergence reflects two equally valid interpretive strategies. As previously discussed (Section 2.2), LLMs and large reasoning models possess a strong capacity to generate intermediate reasoning steps and explanations for their classifications, producing outputs that are more easily explainable.

Examining the explanations, GPT-4o consistently justified its strong matches with generalized reasoning: for multiple instances it stated: “Captures issue resolution history, which directly reflects customer complaints”.

Given that the problem’s objective was to predict customers likely to switch energy providers, such an explanation is logically correct.

GPT-5, on the other hand, provided more personalized and specific justifications, and rather than relying on broad semantic equivalence, it critically assessed the contextual nuances of each feature. For example, it highlighted how a feature captured not only the presence of customer complaints but also their temporal frequency, escalation patterns, or interaction with other variables, distinguishing between superficially similar but semantically distinct insights.

This more critical approach often resulted in GPT-5 assigning lower relevance labels than GPT-4o, reflecting its ability to detect weaknesses or redundancies within the generated features.

Taken together, the divergence between GPT-4o and GPT-5 demonstrates two complementary evaluation modes: GPT-4o’s tendency toward generalized domain alignment, contrasted with GPT-5’s stricter and context sensitive evaluation.

Both can be considered valid, yet GPT-5’s evaluations appear more reliable for benchmarking purposes, as they place stronger emphasis on coherence and the practical usefulness of insights.

The integration of semantic coverage and qualitative assessment complements the existing evaluation framework in two fundamental ways: first, it ensures that generated insights are relevant, non contradictory, and interpretable, aligning more closely with the practical requirements of decision making. Second, it demonstrates that although empirical complexity doesn’t directly determine semantic relevance, it can influence the internal coherence of the outputs.

These perspectives, combining quantitative measures of semantic similarity with qualitative evaluations of coherence, represent an essential step toward a more holistic assessment of LLM agents, ensuring to judge them not only by their statistical alignment with ground truth patterns, but also by the interpretability and real world usability of their insights.

5. Learnings, what to expect next, & conclusion

The starting point of this research is the observation that traditional approaches for measuring human intelligence are too narrow, failing to capture the full range of cognitive abilities. Psychometric traditions, most famously represented by the intelligence quotient (IQ), reduce the richness of human thought into a single number, overlooking crucial dimensions such as creativity, adaptability, social reasoning, and emotional understanding.

This reductionist perspective has not been confined to human psychology, but has also shaped how artificial intelligence has been evaluated, since from the earliest stages of AI research, benchmarks were designed to replicate human brain functions and were structured around the evaluation of static tasks, accuracy based measures, and single dimensional outcomes. While these approaches have been useful in establishing comparability and standardization, they overlook the essence of intelligence itself: the capacity to adapt, to generate meaning, and to reveal what is not immediately apparent.

As with human intelligence, so too with artificial intelligence: what is required are more complete frameworks that capture the multifaceted nature of reasoning and discovery.

In human sciences, alternative models such as Gardner's theory of multiple intelligence or the Cattell Horn Carroll (CHC) model, are trying to broaden the understanding of cognitive diversity, and similarly in artificial intelligence, particularly with the rise of large language models and agentic systems, new benchmarks are required.

These benchmarks must be dynamic rather than purely static, capable of evolving with the systems they assess, and multidimensional rather than narrow, measuring not only final output accuracy but also the processes that generated them. They should avoid obsolescence by relying on more than fixed datasets and being able to assess the interpretability and coherence of insights, focusing on their practical utility.

This research has proposed such a direction through the collaboration with SparkBeyond.

The new generation benchmarking framework developed in this context was designed to meet the requirements of modern AI models: dynamic problem generation, adaptability to multiple domains, and the ability to test insight discovery.

The findings showed that while traditional statistical metrics, like coverage and predictive accuracy, remain indispensable, they still are insufficient to assess the real usefulness of generated insights, being a tool very common in practical life.
Semantic coverage, or the qualitative evaluation of whether insights are interpretable, non contradictory and meaningful, emerged as a missing but yet necessary dimension. This parallels the broader trajectory of human intelligence research, where the field has moved beyond IQ to consider deeper drivers.

From these findings, several broader lessons emerged: first, and most important, intelligence being either human or artificial performance, is irreducibly multidimensional, and any attempt to reduce it into a single metric inevitably distorts its nature, and for this all the assessment methods grounded in this reductive view have gradually been superseded.
Second, early AI evaluation methods inherited this reductionism, focusing on static tasks and accuracy, rather than the richness of reasoning and adaptability, and just as human intelligence requires multidimensional evaluation, modern AI systems, especially LLMs, demand dynamic and evolving benchmarks that go beyond static datasets.
Third, evaluation shouldn't only measure final output accuracy but also assess how solutions are generated, including their reasoning steps.
And finally, evaluation is itself a dynamic process; the frameworks used must evolve at the same rate of the systems they measure, following societal complexity to always reflect the real world needs.

Looking forward, the next steps for SparkBeyond's benchmarking framework point to extend its scope by evaluating customer facing agents against real KPIs, aligning evaluation more closely with practical business impact. Expanding insight discovery to conversational data will further improve the assessment in interactive domains, where adaptability and coherence matter most.
In this way, the platform seeks to become totally self improving, embodying the principle of agents that continuously refine themselves through feedback.
Additional introducing metrics for insight coverage in dialogue to ensure that conversations consistently surface meaningful and actionable knowledge, and further develop the semantic coverage of ground truth insights.

These steps represent a paradigm shift: from static to dynamic, from narrow to multidimensional, from product focused to process oriented, and from abstract benchmarks to frameworks aligned with societal needs.

In a broader sense, the trajectory of evaluation frameworks should reflect the growing complexity of society itself, since humans are becoming always interconnected, generating more data, and reliant on adaptive forms of intelligence, whether biological or artificial. Evaluation methods must therefore ensure that the trusted systems for decision making are not only accurate but coherent, transparent, and aligned with human values.

Just as human intelligence is assessed not only by outcomes but also by processes of reasoning and adaptation, so too must artificial intelligence be judged by its capacity to reason, to adapt, and to generate meaning.

The contribution of this research lies in bringing these threads together: demonstrating the parallels between human and machine evaluation gaps, highlighting the limitations of narrow metrics, and proposing a framework that moves closer to capturing the true essence of intelligence.

This work doesn't claim to solve the challenge of evaluation, just as psychometrics has never fully captured human intelligence, AI benchmarking will remain an evolving field, but what it offers a vision: by adopting dynamic, multidimensional, and semantically grounded frameworks, it becomes possible to design evaluations that do justice to the complexity of intelligence in all its forms.

In the end, the question of how intelligence is evaluated is inseparable from the question of what is really valued in intelligence: if adaptability, creativity, coherence, and the capacity to generate meaning are the most valued, then complete evaluation frameworks must be designed to recognize and reward them.

Bibliography

[1] NIST. (2023). AI Risk Management Framework 1.0. U.S. National Institute of Standards and Technology.

[2] Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. IBM Journal of Research and Development, 3(3), 210–229.

[3] Cattell, R. B. (1963). Theory of fluid and crystallized intelligence: A critical experiment. Journal of Educational Psychology, 54(1), 1–22.

[4] Chollet, F. (2019). On the measure of intelligence.

[5] Binet, A., & Simon, T. (1905). Méthodes nouvelles pour le diagnostic du niveau intellectuel des anormaux. L'Année Psychologique, 11, 191–336.

[6] Spearman, C. (1904). "General intelligence," objectively determined and measured. The American Journal of Psychology, 15(2), 201–292.

[7] Zhao, X., Zhang, W., Wang, J., He, X., Chua, T.-S., & Gai, K. (2019). Deep reinforcement learning for list-wise recommendations. In Proceedings of the 2018 World Wide Web Conference (pp. 95–104).

[8] Zenderland, L. (1998). Measuring minds: Henry Herbert Goddard and the origins of American intelligence testing. Cambridge University Press.

[9] Terman, L. M. (1916). The measurement of intelligence: An explanation of and a complete guide for the use of the Stanford revision and extension of the Binet-Simon intelligence scale. Houghton Mifflin.

[10] Kovács, K. (2023). William Stern: The relevance of his program.

[11] American Psychiatric Association. (2013). Diagnostic and statistical manual of mental disorders (5th ed.). American Psychiatric Publishing.

[12] Flynn, J. R. (2007). What is intelligence? Beyond the Flynn effect. Cambridge University Press.

[13] Flynn, J. R. (1987). Massive IQ gains in 14 nations: What IQ tests really measure. Psychological Bulletin, 101(2), 171–191.

[14] Gardner, H. (1983). Frames of mind: The theory of multiple intelligences. Basic Books.

[15] Goleman, D. (1995). Emotional intelligence: Why it can matter more than IQ. Bantam Books.

[16] Taleb, N. N. (2019). IQ is largely a pseudoscientific swindle. Medium.

[17] McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. The Bulletin of Mathematical Biophysics, 5(4), 115–133.

[18] Turing, A. M. (1950). Computing machinery and intelligence. Mind, 59(236), 433–460.

[19] Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review, 65(6), 386–408.

[20] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. Nature, 323(6088), 533–536.

[21] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

[22] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., … Amodei, D. (2020). Language models are few-shot learners. In Advances in Neural Information Processing Systems (Vol. 33, pp. 1877–1901).

[23] Kumar, K., Ashraf, T., Thawakar, O., Anwer, R. M., Cholakkal, H., Shah, M., Yang, M.‑H., Torr, P. H. S., Shahbaz Khan, F., & Salman Khan. (2025). LLM post‑training: A deep dive into reasoning large language models.

[24] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., … Polosukhin, I. (2017). Attention is all you need. In Advances in Neural Information Processing Systems (Vol. 30).

[25] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., … Bengio, Y. (2014). Generative adversarial nets. In Advances in Neural Information Processing Systems (Vol. 27).

[26] Floridi, L., & Chiriatti, M. (2020). GPT-3: Its nature, scope, limits, and consequences. Minds and Machines, 30(4), 681–694.

[27] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of NAACL-HLT 2019 (pp. 4171–4186).

[28] Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2020). BERTScore: Evaluating text generation with BERT. International Conference on Learning Representations (ICLR).

[29] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. OpenAI.

[31] Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction (2nd ed.). MIT Press.

[32] Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., & Amodei, D. (2017). Deep reinforcement learning from human preferences.

[33] Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., & Lowe, R. (2022). Training language models to follow instructions with human feedback.

[34] de Winter, J. C. F., Dodou, D., & Eisma, Y. B. (2024). System 2 thinking in OpenAI's o1-preview model: Near-perfect performance on a mathematics exam.

[35] Wilson, M. (2002). Six views of embodied cognition. Psychonomic Bulletin & Review, 9(4), 625–636.

[36] Barsalou, L. W. (2008). Grounded cognition. Annual Review of Psychology, 59, 617–645.

[37] Samuel O. Ortiz & Sarah K. Cehelyk, "The Bilingual Is Not Two Monolinguals of Same Age: Normative Testing Implications for Multilinguals", Journal of Intelligence, 2024.

[38] Alfonso, V. C., Flanagan, D. P., & Radwan, S. (2005). The impact of the Cattell–Horn–Carroll theory on test development and interpretation of cognitive and academic abilities. In Contemporary intellectual assessment: Theories, tests, and issues (2nd ed., pp. 185–202).

[39] Flanagan, Dawn & McGrew, Kevin. (1998). Interpreting Intelligence Tests from Contemporary Gf-Gc Theory. Journal of School Psychology - J SCH PSYCHOL. 36. 151-182. 10.1016/S0022-4405(98)00003-X.

[40] Erdodi, L. A., Abeare, C. A., Lichtenstein, J. D., Tyson, B. T., Kucharski, B., Zuccato, B. G., & Roth, R. M. (2017). Wechsler Adult Intelligence Scale–Fourth Edition (WAIS-IV) processing speed scores as measures of noncredible responding: The third generation of embedded performance validity indicators. Psychological Assessment, 29(2), 148–157.

[41] Weschler, D. (2008). WAIS-IV: Wechsler Adult Intelligence Scale–Fourth Edition. Pearson Assessment.

[42] Weiss, L. G., & Gabel, A. D. (2008). WISC-IV technical report #6: Using the Cognitive Proficiency Index in psychoeducational assessment. Pearson Assessments.

[43] Ackerman, P. L. (2022). Intelligence process vs. content and academic performance: A trip through a house of mirrors. Journal of Intelligence, 10(4), 128.

[44] Vaughan, A. C., & Birney, D. P. (2023). Within-individual variation in cognitive performance is not noise: Why and how cognitive assessments should examine within-person performance. Journal of Intelligence, 11(6), 110.

[45] Sternberg, R. J. (2017). Speculations on the role of successful intelligence in solving contemporary world problems. Journal of Intelligence, 5(4), 28.

[46] Sternberg, R. J. (2021). Adaptive intelligence: Surviving and thriving in times of uncertainty. Cambridge University Press.

[47] Shieber, S. M. (1994). Lessons from a restricted Turing test. Communications of the ACM, 37(6), 70–78.

[48] Bender, E. M., & Koller, A. (2020). Climbing towards NLU: On meaning, form, and understanding in the age of data. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 5185–5198.

[49] Jones, C. R., & Bergen, B. K. (2025, March 31). Large language models pass the Turing test.
Breiman, L. (2001). Statistical modeling: The two cultures. Statistical Science, 16(3), 199–231.

[50] Kwa, T., West, B., Becker, J., Deng, A., Garcia, K., Hasin, M., Jawhar, S., Kinniment, M., Rush, N., Von Arx, S., Bloom, R., Broadley, T., Du, H., Goodrich, B., Jurkovic, N., Miles, L. H., Nix, S., Lin, T., Painter, C., Parikh, N., Rein, D., Sato,

L. J. K., Wijk, H., Ziegler, D. M., Barnes, E., & Chan, L. (2024). Measuring AI ability to complete long tasks. Model Evaluation & Threat Research (METR).

[51] Liang, P., Bommasani, R., Lee, T., Tsipras, D., Soylu, D., Yasunaga, M., Zhang, Y., Narayanan, D., Wu, Y., Kumar, A., Newman, B., Yuan, B., Yan, B., Zhang, C., Cosgrove, C., Manning, C. D., Ré, C., Acosta-Navas, D., Hudson, D. A., … Koreeda, Y. (2022). Holistic Evaluation of Language Models. Stanford University, Center for Research on Foundation Models (CRFM), Institute for Human-Centered Artificial Intelligence (HAI).

[52] Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E. P., Zhang, H., Gonzalez, J. E., & Stoica, I. (2023). Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena.

[53] Wang, T., Kulikov, I., Golovneva, O., Yu, P., Yuan, W., Dwivedi-Yu, J., Pang, R. Y., Fazel-Zarandi, M., Weston, J., & Li, X. (2024). Self-taught evaluators.

[54] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models.

[55] Shojaee, P., Mirzadeh, I., Alizadeh, K., Horton, M., Bengio, S., & Farajtabar, M. (2025). The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity.