Intelligence Goodput

A metric to measure the speed of intelligence

Haifeng Jin, August 2025

Processing speed has long been recognized as an important metric in human cognitive ability [1] and intelligence scale measurement [2,3,4]. It reflects the fundamental capacity to perform daily tasks as a human, including reading comprehension, communication, and driving. Both the quality of work and the time required are essential to its evaluation.

To measure the intelligence scale of artificial intelligence (AI), various benchmark datasets have been designed, analogous to tests for humans. For example, the MMLU dataset [5] consists of multiple-choice questions to test multi-language understanding, the GPQA dataset [6] consists of graduate-level questions on a variety of subjects, and the MATH dataset [7] comprises competition math problems.

However, when experiments are conducted to test AI on these datasets, unlike tests for humans, only the quality of work is emphasized, while the time aspect of the test is typically ignored. Such an approach does not satisfy the requirements of AI in real-world applications.

From a pragmatic perspective, many tasks are time-sensitive, such as autonomous driving and customer service. In these applications, the output of intelligence is required within a limited time window. Even in the early days of deep learning, there was an implicit assumption of a time limit. When AlphaGo [8] defeated the best human Go player in 2016, it adhered to the same rules, including time constraints, as the human player.

Implications of test-time scaling

With the advent of the test-time scaling law [9], the scores achieved on these AI benchmark datasets alone can no longer fully measure intelligence.

According to the test-time scaling law, a small model can solve harder problems by spending more tokens "thinking", while a larger model can solve the same problem with fewer tokens. By analogy to a human IQ test, it is as if one person spent more time and used a long sheet of scratch paper to finish the test, while another person used very little time and no scratch paper at all. If both achieve the same number of correct answers, it is more rational to conclude that the person who completed the test faster possesses higher intelligence, likely employing a more efficient method.

That smaller and larger models can answer the same question correctly does not imply they have the same intelligence level; rather, they approached the problem differently. Therefore, without any constraint on the output, whether in terms of the number of tokens or output speed, it is impossible to measure the actual intelligence level of AI solely based on correctness.

From a user's perspective, estimating the time required for AI to complete tasks has become increasingly challenging. Prior to the advent of test-time scaling, models typically used a similar number of tokens to solve each problem. With test-time scaling, models are incentivized to generate more tokens for certain problems, leading to greater variability. Users now have access to metrics such as tokens per second for the APIs they utilize, but lack information about the total number of tokens needed for a given task. As a result, they can no longer reliably estimate the time required for task completion.

Why speed was ignored

If processing speed is so important to intelligence, why was it historically ignored? The reasons are twofold. First, AI was not sufficiently advanced. Early systems could not pass the Turing test [10], solve math or coding problems, or perform in-depth reading or writing. Given that AI was in a primitive stage, the focus was on enabling AI to perform new tasks rather than on the speed of task completion. Second, AI applications were all task-specific. The AI for X-ray image processing [11] was entirely different from the AI used for recommendation systems [12]. The applicability of AI was measured case by case by application builders, without the need for a unified processing speed metric.

However, with the advent of large language models (LLMs) [13], AI has become substantially more capable and generalizable, with applications spanning from medical diagnosis [14] to recommendation systems [15]. Consequently, it is now both rational and timely to establish a unified metric for AI processing speed.

A mental shift for evaluating AI

Additionally, there has been a significant shift in how application builders interface with models as summarized in Table 1. Initially, developers focused on training application-specific deep learning models, beginning with the advent of AlexNet [16]. With the emergence of ChatGPT [13], the paradigm shifted toward pretraining large language models (LLMs). The public release of Llama [17] enabled post-training and fine-tuning of open models. As the capabilities of open models advanced, such as the Qwen series [18], it became increasingly feasible to serve an open-weight model "as is" without additional fine-tuning. More recently, the introduction of DeepSeek [19], which substantially reduced token costs, has made it more cost-effective to utilize hosted APIs from major AI service providers rather than maintaining proprietary infrastructure.

Table 1. Paradigm shifts of AI usages

Paradigms	**Defining Moments**
Deep Learning	AlexNet
Pretraining LLMs	ChatGPT
Open Model Fine-tuning	Llama
Open Model Serving	Qwen
Hosted APIs	DeepSeek

Therefore, when evaluating the intelligence of AI, a shift is required from evaluating static models, pure mathematical constructs defined by neural architectures and parameters, to evaluating hosted AI services or APIs. It is necessary to assess their processing speed for applicability to time-sensitive tasks.

Existing metrics

There are established metrics to evaluate the speed of LLMs, such as time to first token (TTFT) and tokens per second (TPS) [20]. These are suitable metrics for serving language models, but there are two major limitations when considering them as general metrics for AI processing speed.

Unlike the time-limit for an IQ test, these metrics focus on the number of tokens rather than the actual useful information produced by intelligence. For example, a simple program without any intelligence can output random tokens at a very high speed.

Therefore, a new metric is needed that measures the intelligent portion of the output rather than the total volume of tokens produced per unit time.

Intelligence goodput

I propose intelligence goodput as a metric for measuring the processing speed of AI.

Intelligence goodput is a measurement indicating the maximum amount of intelligent information that an AI service can produce in a given amount of time, formally expressed as:

$$G = \frac{I}{t}$$

where G is the intelligence goodput, I is the amount of intelligent information, and t is the total time spent.

It is important to note that intelligence goodput primarily measures the output speed of AI, rather than input, for two reasons. First, the main impact of intelligence goodput is in human-AI interaction, which will be discussed further in the next section. Second, AI's processing speed is mainly bounded by output, not input. Thus, it is more meaningful to track AI progress with a metric that measures its bottleneck.

Despite the formal definition of intelligence goodput, certain ambiguities remain in practical measurement. First, "intelligent information" is not an unambiguous term. Defining the amount of intelligence contained in the output of an AI model remains a challenging task. Second, time is also subject to multiple interpretations in computer science.

Regarding time measurement, the two popular choices are CPU time and wall time. Wall time is preferred, as intelligence goodput is primarily relevant for human-AI interactions and applications. End-to-end latency is more informative than technical details such as CPU time.

To address the ambiguity in "intelligent information" in the definition, a method for calculating intelligence goodput are proposed. We divide the score achieved by AI on benchmark datasets by the time used to produce the answer. In this way, we delegate the challenging task of measuring intelligence to the existing benchmarks. It can be formally expressed as follows.

Let $S = s_1, s_2, \ldots, s_n$ be a set of scores from n individual AI benchmarks normalized to the same range, and $W = w_1, w_2, \ldots, w_n$ be a corresponding set of weights, where w_i represents the relative importance of each benchmark. The total time expended across all assessments is denoted by t. Intelligence goodput G can be defined as:

$$G = \frac{\sum_{i=1}^{n} w_i s_i}{\sum_{i=1}^{n} w_i \cdot t}$$

Experiments

We evaluated several high-quality models from top AI companies for their intelligence goodput. The source data, including intelligence benchmark scores, tokens/s, and total number of tokens used when running intelligence benchmarks, were collected from Artificial Analysis [21].

The experimental results are shown in Table 2, which displays their intelligence index (I) (normalized and averaged benchmark scores), speed (measured by tokens/s), verbosity (the total number of tokens produced during all benchmarks, including reasoning tokens), and intelligence goodput (IG). For more details on the methodology on the calculations of the intelligence index, speed, and verbosity, please visit the Artificial Analysis [21] website.

Table 2. Intelligence Goodput Results

Models	I	Speed	Verbosity	IG
Grok 4 Fast	60	257	60.5M	254.88
GPT-5 Medium	66	138	44.9M	202.85
Gemini 2.5 Flash	54	262	71M	199.27
GPT-5 High	68	125	85M	100.00
Gemini 2.5 Pro	60	156	101M	92.67
Claude 4.5 Sonnet	63	60	41.2M	91.75
Grok 4	65	35	121.6M	18.71

Since all models included have similar high intelligence scores, we focus mainly on speed and verbosity. Based on their intelligence goodput, the models can be roughly grouped into three clusters:

- The 200 club: Grok 4 Fast, GPT-5 Medium, and Gemini 2.5 Flash are in this category with intelligence goodput ranging from around 200 to 250. Grok 4 Fast performs well in both tokens/s and output tokens, ranking at the top. GPT-5 Medium has low verbosity, while Gemini 2.5 Flash excels in tokens/s.
- The 100 club: GPT-5 High, Gemini 2.5 Pro, and Claude 4.5 Sonnet belong to this category with intelligence goodput ranging from around 90 to 100. GPT-5 High and Gemini 2.5 Pro have high tokens/s, while Claude 4.5 Sonnet has much lower verbosity.
- The below 20 club: Grok 4 has both low tokens/s and high verbosity, placing it at the bottom of the leaderboard in contrast with Grok 4 Fast, which ranks at the top.

Discussions

Benchmark datasets score only the final answer, disregarding intermediate outputs such as the reasoning process. This approach effectively isolates the intelligent portion of the total output, aligning with the definition of intelligence goodput.

For AI application developers working with time-bounded tasks, the tokens-per-second metric provides limited insight, as the number of tokens required to complete a task remains unknown beforehand. In contrast, intelligence goodput offers a more informative measure of how much useful work can be accomplished per unit time.

Furthermore, incorporating intelligence goodput as an optimization target during model training may help mitigate the verbosity problem commonly observed in LLMs. This problem manifests as unnecessarily long chains of thought that repeatedly revisit the same logical steps. Since longer reasoning processes result in slower time-to-answer and consequently lower intelligence goodput, optimizing for this metric naturally incentivizes more concise and efficient reasoning.

Limitations

The proposed intelligence goodput metric has two primary limitations.

First, it is currently limited to text-based outputs. While AI models are increasingly multimodal, benchmark datasets for evaluating the intelligence of outputs in other modalities, such as images, remain underdeveloped. Such outputs are typically assessed for real-world fidelity and artistic merit rather than intelligence.

Second, the metric is computationally expensive to measure. Calculating intelligence goodput requires significant engineering effort to build evaluation infrastructure capable of running APIs through comprehensive benchmark datasets. Additionally, the cost of API token consumption for executing these benchmarks can be substantial.

Third, tokens wasted on incorrect answers are ignored. This characteristic of intelligence goodput as a metric may bias results in favor of more intelligent models, since less intelligent models waste many tokens while producing incorrect answers that contribute nothing to the final score.

Conclusions

This article introduces the concept of intelligence goodput a metric that measures the processing speed of AI services by quantifying the rate at which they produce intelligent information. The article advocates for a new evaluation paradigm that focuses on dynamic, served AI systems, emphasizing the crucial interplay between hardware, software, and models.

Ultimately, the most important takeaway is that when evaluating AI for performing tasks traditionally performed by humans, the evaluation criteria should mirror those used for humans. The distinction between assessing human and AI performance for a given task will become increasingly blurred. Innovative approaches are required to integrate processing speed into comprehensive intelligence assessments for AI.

References

[1] Woodcock, R. W., & others. (1989). *Woodcock-Johnson tests of cognitive ability*. DLM Teaching Resources.

[2] Fry, A. F., & others. (1996). Processing speed, working memory, and fluid intelligence: Evidence for a developmental cascade. *Psychological Science*.

[3] Lichtenberger, E. O., & others. (2012). *Essentials of WAIS-IV assessment*. John Wiley & Sons.

[4] Flanagan, D. P., & others. (2017). *Essentials of WISC-V assessment*. John Wiley & Sons.

[5] Hendrycks, D., & others. (2021). Measuring Massive Multitask Language Understanding. *ICLR*.

[6] Rein, D., & others. (2024). GPQA: A Graduate-Level Google-Proof Q&A Benchmark. *COLM*.

[7] Hendrycks, D., & others. (2021). Measuring Mathematical Problem Solving With the MATH Dataset. *NeurIPS*.

[8] Silver, D., & others. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*.

[9] Snell, C., & others. (2025). Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *ICLR*.

[10] Turing, A. M. (2009). *Computing machinery and intelligence*. Springer.

[11] Çallı, E., & others. (2021). Deep learning for chest X-ray analysis: A survey. *Medical Image Analysis*.

[12] He, X., & others. (2017). Neural collaborative filtering. *The Web Conference*.

[13] Brown, T., & others. (2020). Language models are few-shot learners. *NeurIPS*.

[14] Ullah, E., & others. (2024). Challenges and barriers of using large language models (LLM) such as ChatGPT for diagnostic medicine with a focus on digital pathology–a recent scoping review. *Diagnostic Pathology*.

[15] Wu, X., & others. (2024). Could small language models serve as recommenders? Towards data-centric cold-start recommendation. *The Web Conference*.

[16] Krizhevsky, A., & others. (2012). Imagenet classification with deep convolutional neural networks. *NeurIPS*.

[17] Touvron, H., & others. (2023). Llama: Open and efficient foundation language models. *arXiv Preprint arXiv:2302.13971*.

[18] Bai, J., & others. (2023). Qwen technical report. *arXiv Preprint arXiv:2309.16609*.

[19] Guo, D., & others. (2025). DeepSeek-R1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv Preprint arXiv:2501.12948*.

[20] Zhong, Y., & others. (2024). DistServe: disaggregating prefill and decoding for goodput-optimized large language model serving. *OSDI*.

[21] Artificial Analysis, Inc. (2024). *AI Model & API Providers Analysis*. (https://artificialanalysis.ai/)