The Power and Peril of the Black Box

When a medical AI system once recommended denying a patient treatment, the doctors hesitated—but couldn't explain why.

The algorithm's reasoning was invisible, locked inside a mathematical **"black box."** Only later did an audit reveal the model had learned to equate zip codes with health outcomes—unintentionally penalizing people from poorer neighborhoods.

This moment captures why AI explainability isn't just a technical nicety**—it's a moral and practical necessity.** By making the logic behind AI decisions transparent, we can question, improve, and trust the systems that increasingly shape our lives.

Why AI Explainability Matters

Explainability is the foundation of trustworthy and accountable AI.

By making model reasoning visible, it helps users understand decisions, builds confidence, and enables ethical oversight. Transparent reasoning also reveals hidden risks—such as bias or unsafe correlations—before they cause harm.

It's a legal and ethical requirement.

Frameworks like the EU AI Act, GDPR's provisions on automated decision-making, and the U.S. AI Bill of Rights emphasize transparency and traceability as foundations for accountability.

It enhances debugging, learning, and collaboration.

By showing which inputs drive outputs, explainability helps developers diagnose errors and refine models. It also empowers humans to question, interpret, and improve AI decisions, turning opaque systems into transparent partners that boost both reliability and trust.

Why Some Models Are Inherently More Explainable Than Others

The ease with which humans understand an AI reasoning depends on two related properties: **interpretability** and explainability. Though often used interchangeably, they address different facets of transparency.

Interpretability

is the degree to which a person can directly grasp a model's inner workings—its parameters, structure, and decision logic—without extra tools.Interpretable models, like linear regression or decision trees, are transparent by design: one can trace how inputs lead to outputs through clear, inspectable rules.

Explainability, by contrast, is how well a model can communicate its reasoning in human terms, regardless of internal transparency. It centers on the human experience of understanding, rather than the simplicity of the model's mechanics. Explainability can be achieved through interpretable architectures or post hoc through tools like **SHAP (SHapley Additive Explanations)** or visualizations. More detail further in the documentation.

Not all models are equally transparent. Simpler models—such as **K-NN (k-nearest-neighbor)** or decision trees—are inherently interpretable because their logic aligns with human intuition and is traceable.

Complex AI systems like large language models are good at spotting patterns in vast quantities of unstructured data, but they don't always explain their thinking clearly. Even when we can see how they work, their answers can change depending on the situation, which makes them hard to fully understand.

The trade-off, therefore, lies not just between performance and transparency but between scope and transparency. Small, focused AI models are easier to understand and check. Bigger, more powerful models need special tools to explain how they work, so we can still trust them.

> **In essence:** All interpretable models are explainable, but not all explainable models are interpretable. Interpretability is one path to the ultimate goal—explainability.

AI Explainability Scorecard

It's important to set clear criteria to evaluate the need for explainable AI. Some models are harder to explain than others, and not every AI use case needs the same level of transparency. High-stakes or safety-critical systems, such as medical diagnosis, financial assessment, or autonomous control, require stronger explainability standards. Lower-risk or exploratory uses can tolerate less explainability if it enables broader scope or experimentation. This scoring method helps align the required level of explainability with the risk, impact, and responsibility of the AI system's deployment.

To evaluate how effectively an AI model communicates its reasoning, each criterion should be rated on a 1–5 scale. This scale aligns directly with the "Question It Answers" column in the accompanying table.

A score of 5 indicates a strong "yes"
the explanation fully answers the question to the user's satisfaction.

A score of 1 indicates a strong "no"
the explanation fails to answer the question meaningfully.

This scoring method keeps things simple and focused on results. It looks at how well the explanation serves the user. The final score is the average of all the individual scores.

	1	2	3	4	5
Faithfulness	○	○	○	○	○
Comprehensibitlity	○	○	○	○	○
Consistency	○	○	○	○	○
Accessibility	○	○	○	○	○
Optimization Clarity	○	○	○	○	○

Criterion	Definition	Question it Answers
Faithfulness	The explanation accurately represents the model's true reasoning process.	"Is this explanation actually how the model made its decision?"
Comprehensibility	The explanation is clear and meaningful to non-technical users and subject-matter experts (SMEs), enabling them to interpret and trust the model's reasoning within their domain.	"Does this explanation help a non-technical expert understand and act within their field?"
Consistency	Similar inputs lead to similar explanations.	"Would the model explain similar decisions in the same way?"
Accessibility	The explanation is easy to obtain, interpret, and apply without excessive time, expertise, or computational resources (excluding the effort to build the model itself).	"Can this explanation be generated and used efficiently without significant burden?"
Optimization Clarity	The explanation provided has value (beyond audit trails & compliance) to engineers building AI systems, helping them debug , validate, and improve the model's design or performance.	"Does this explanation provide actionable insights to improve the development of the AI system?"

Interplay Between Criteria

These five work together, but sometimes you have to give up one to get more of another. For example, making an explanation more faithful might show how complex the model really is, which can make it harder for non-technical people to understand. On the other hand, making explanations quicker and easier to use might mean they aren't as accurate or consistent.

The objective is not to maximize all criteria, but to achieve balance aligned with the model's purpose, audience, and risk profile. A medical AI system, for example, should score high in faithfulness , consistency , and comprehensibility . A research prototype might emphasize optimization clarity and accessibility to accelerate development and learning.

Ultimately, this scorecard ensures that explainability is treated as a measurable spectrum of quality, where each score reflects how effectively a model's explanations answer the questions that matter to its users.

Comparing Explainability Across Model Types

In this section, we apply the AI Explainability Scorecard to evaluate four model types: K-Nearest Neighbors (K-NN), Neural Networks, Transformers, and Large Language Models (LLMs). These four models were selected because they represent key stages in the evolution of AI systems and distinct approaches to learning and reasoning.

K-Nearest Neighbors (K-NN)
A simple, clear method that makes decisions based on similar or nearby data points—easy to understand.

Neural Networks:
Powerful tools used in deep learning. They work well for specific tasks, but it's hard to see how they make decisions.

Transformers
A newer type of neural networks that use "attention" mechanisms to focus on important parts of the input. This allows it to handle more generalized tasks.

Large Language Models (LLMs)
The latest and most advanced systems—an evolution of transformers. They can handle even more complex, and generalized scope of tasks, but how they decide things is mostly hidden.

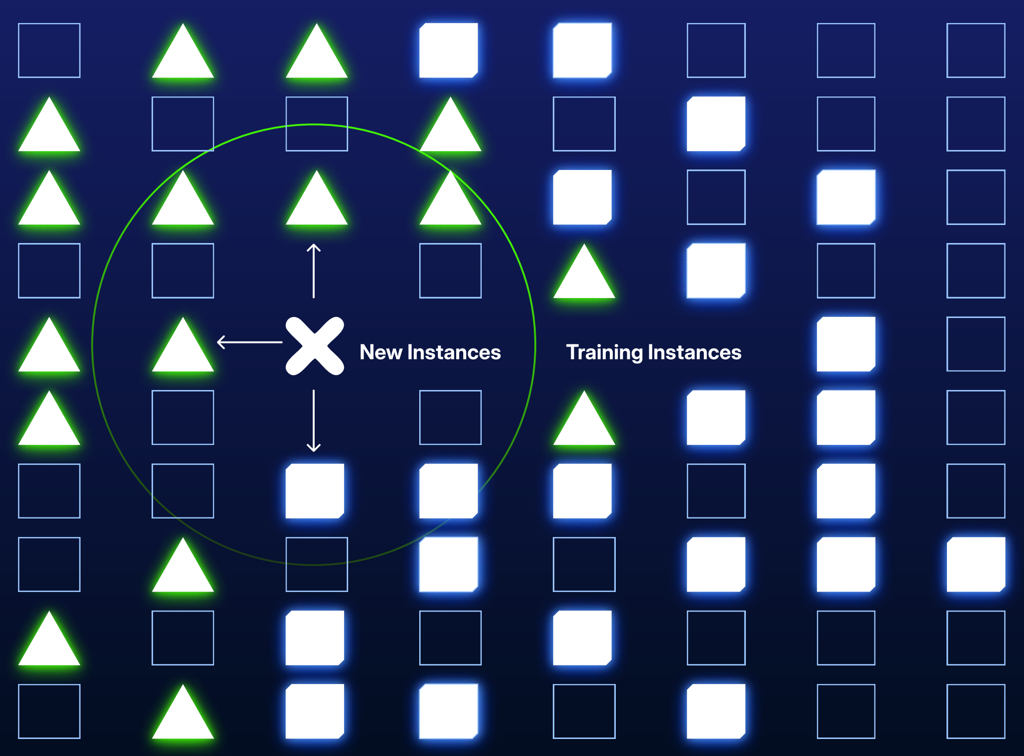
Together, these models span the full spectrum of explainability — from inherently interpretable methods to those requiring advanced, external explanation tools. The goal is to assess the level of explainability achievable with minimal effort.

> **The goal is to assess the level of explainability achievable with minimal effort.**

K-Nearest Neighbors (K-NN)

K-NN is a simple AI method that makes decisions by looking at the most similar examples from its training data. The model's behavior is directly tied to its data, making it one of the most inherently interpretable AI methods.

KNN Algorithm Example



- **Faithfulness** 5.0

  KNN is highly faithful to its decision process—each prediction can be traced to training samples—though results depend on how distance metrics and data scaling are defined.

- **Comprehensibility** 4.0

  K-NN reasoning is intuitive.Similar cases leading to similar outcomes" is easy to grasp. But there is complexity in interpreting a list of neighbors, distances, and labels to connect examples to outcomes. It's largely understandable, but less direct than a clear narrative explanation such as an LLM's Chain-of-Thought.

Consistency 5.0 ◉◉◉◉◉

KNN is deterministic when data and tie-breaking rules are fixed, though some implementations introduce randomness in resolving equal distances.

Accessibility 5.0 ◉◉◉◉◉

Explanations are **immediate and transparent**—the relevant samples, distances, and labels can be directly viewed without any tooling or additional computation. Access to the decision evidence is as simple as inspecting the nearest data points.

Optimization Clarity 5.0 ◉◉◉◉◉

K-NN provides **complete actionable insights** for engineers. Developers can clearly identify how to improve model outcomes by directly adjusting the training data: add or remove samples, refine feature scaling, or correct mislabeled data without the need to re-train the model. This direct mapping between data and behavior offers unparalleled debugging, optimization potential and iteration speed.

AI Explainability Score Summary

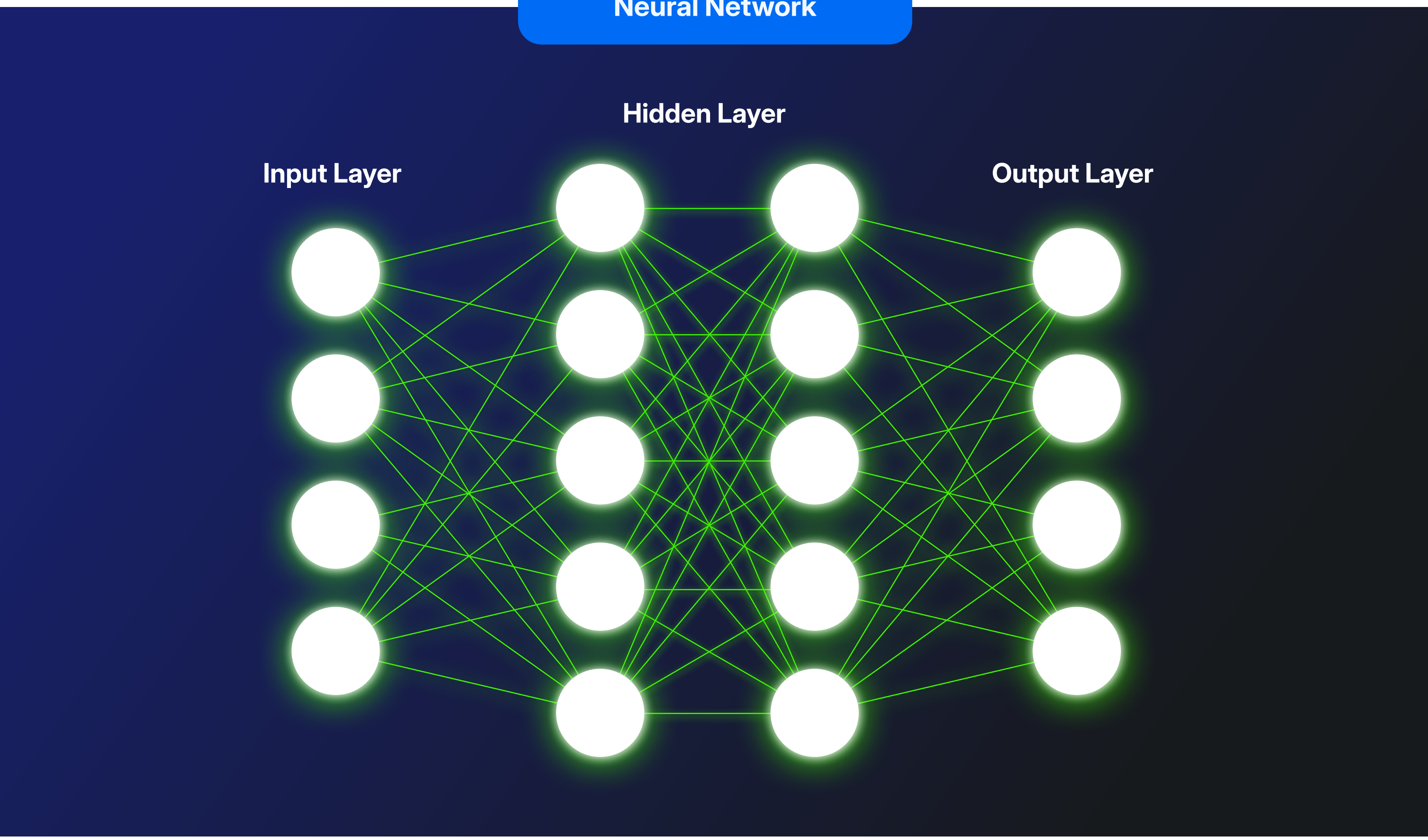
4.8 out of 5

K-NN achieves near-perfect explainability. Its logic is explicit, faithful, consistent, and easy to inspect. Even if it takes a little effort to understand how K-NN measures similarity, it's always clear how the model makes decisions and how to make it better. In short, K-NN is built to be clear and easy to explain—making it a top choice for simple, understandable AI.

Neural **Networks**

Neural networks are made of layers of connected points that change input data using math and special rules. They're really good at finding patterns in complex data and usually focus on one clear task with clear inputs and expected outputs. But how they make decisions is hard to see or understand. Without special tools, it's almost impossible to explain their thinking.

Neural Network



- **Faithfulness** 3.0

  SHAP estimates the contribution of input features to predictions—it doesn't reflect internal causal reasoning, but provides an interpretable approximation of feature influence. Its biggest drawback is that it is computational heavy, slow and expensive to perform while not being able to grasp the entire model at once.

- **Comprehensibility** 3.0

  When we use tools like SHAP or visualizations, neural networks become easier to understand. Because they focus on one task, it's possible for experts to see which parts of the data affect the results. But people without technical training might still need help to understand the explanations.

Consistency 4.0

Neural networks produce consistent outputs and explanations when inputs and parameters are fixed. Small changes in input or using different explanation tools might slightly change the results, but the overall reasoning stays consistent for the task.

Accessibility 3.0

Tools like SHAP and LIME make explanations possible but require technical skill and computing resources. Generating meaningful visualizations is practical for engineers, but not effortless or broadly accessible.

Optimization Clarity 4.0

Developers can inspect weights, gradients, and activations to debug and optimize performance. These tools clarify how the system works, but not why specific outputs occur.

AI Explainability Score Summary

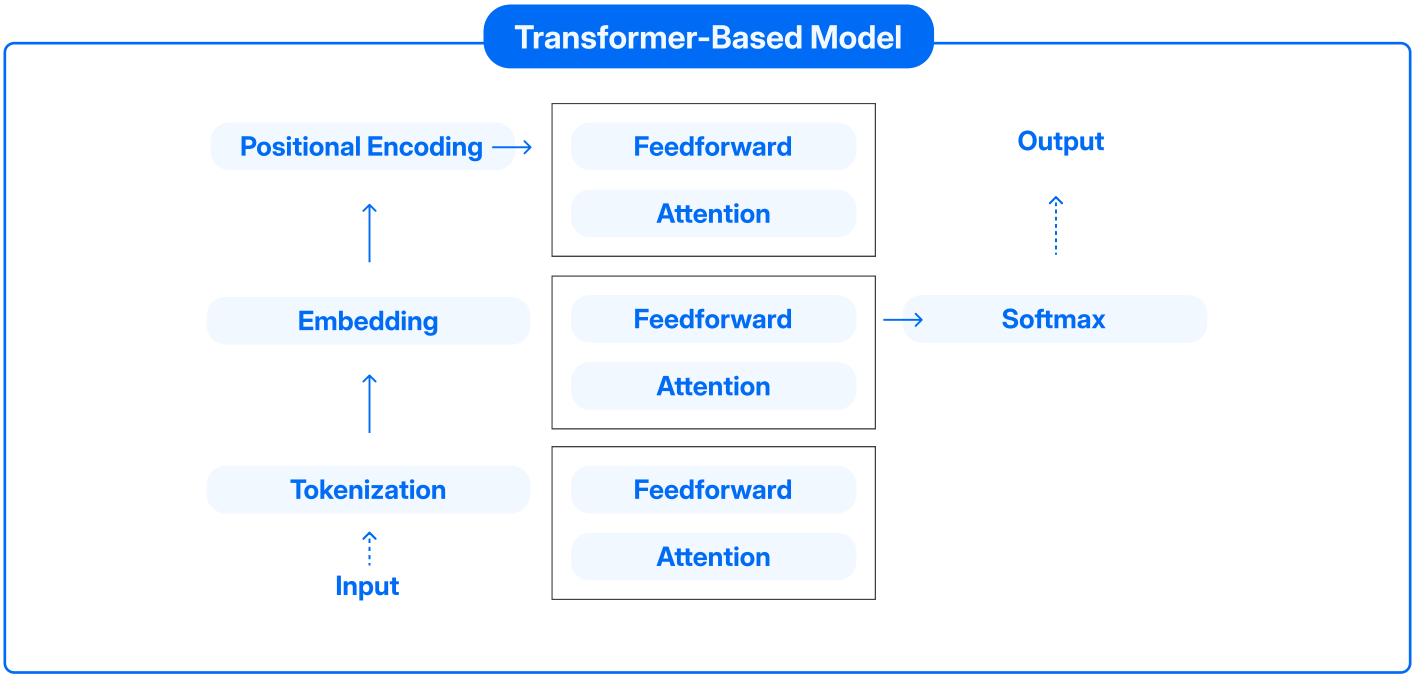
3.4 out of 5

Neural networks offer moderate explainability through external tools or methods, rather than inherent transparency. They work well for specific tasks, are consistent, and can be studied by technical experts. With tools like SHAP and visualizations, their thinking can be partly understood—enough for experts to improve them—but they're still hard for everyday users to fully grasp.

Transformer-Based Models

Transformers are a type of neural network that uses self-attention to understand how different parts of the input relate to each other—especially in natural language. But because they're so complex, it's hard to see how they make their decisions.

To interpret their decisions, without auxiliary tools like attention visualization or surrogate modeling, transformers offer limited—though not zero—explainability. This surrogate model provides a simplified, interpretable representation of how inputs influence outputs—allowing partial insight into the transformer's logic. Without such a model, transformers would score near zero on explainability.



Faithfulness 3.0

Surrogate models try to infer how transformers behave, but they can't fully show how transformers actually think. They show patterns between inputs and outputs, but not the real reasons behind decisions. So, they're only partially faithful.

Comprehensibility 2.0

Surrogate models produce results—like numbers and charts—that can still be hard for most non-technical people to understand. You need to know some data science to make sense of them. These explanations help analysts, but aren't easy for everyday users to follow.

Consistency 3.0

When the data and rules stay the same, surrogate models usually give steady explanations for similar inputs. But small changes—like slightly different data or retraining—can change which input features the transformer model deems important (or where the attention is focused). Explanations are somewhat stable, but not always reliable.

Accessibility 1.0

Creating a surrogate model requires separate development, tuning, and validation—adding significant overhead. Unlike inherently interpretable models, explanations are not available "out of the box.· This extra effort makes accessibility very low.

Optimization Clarity 4.0

Surrogate models give engineers structured insights into model behavior, feature effects, and potential biases. These explanations are highly useful for debugging and improving system design. However, they are still just indirect inferences and don't give a clear look inside how the transformer actually works.

AI Explainability Score Summary

2.6 out of 5

Transformers achieve moderate explainability only through surrogate models. While these auxiliary models provide partial faithfulness and strong technical insight, they offer limited clarity for non-technical users and require substantial extra effort to build. Explanations are moderately consistent but remain indirect approximations of the transformer's true reasoning. In short, transformers are powerful yet opaque systems, dependent on surrogate models for interpretability, and far from transparent by design.

Large Language Models (LLMs, Chain-of-Thought Only)

LLMs are big AI models built to guess the next word in a sentence by learning from huge amounts of text. They can understand and generate language in smart ways, but how they think and reason is a mystery.For this comparison, explainability is evaluated using Chain-of-Thought (CoT) reasoning only—where the model expresses its reasoning steps in natural language—without applying additional interpretive methods.

- **Faithfulness** 2.0

  CoT outputs often sound plausible but do not necessarily reflect the model's actual internal decision-making.

- **Comprehensibility** 5.0

  The CoT narrative is highly understandable to both SMEs and general users.

- **Consistency** 1.0

  Explanations vary with the same input, even when settings like temperature and random seed are left consistent.

- **Accessibility** 5.0

  CoT reasoning can be easily generated through simple prompting—requiring no technical expertise.

- **Optimization Clarity** 2.0

  The narrative format provides limited value for debugging or improving models.

AI Explainability Score Summary

3 out of 5

LLMs, when using Chain-of-Thought alone, excel in accessibility and surface-level comprehensibility, but perform poorly in faithfulness, consistency and optimization clarity. Their explanations are behaviorally convincing yet structurally opaque, underscoring the need for additional methods to achieve deeper transparency.

Understanding the Tradeoffs in Model Explainability

There's no single "right" model—just varying degrees of explainability based on your organization's risk tolerance. This table isn't about judgment—it's about clarity.

Use it to choose the right level of explainability for your AI use case.

Criterion	K-Nearest Neighbors (K-NN)	Neural Networks	Transformer-Based Models	Large Language Models (LLMs, Chain-of-Thought Only)
Faithfulness	5	3	3	2
Comprehensibility	4	3	1	5
Consistency	5	4	5	1
Accessibility	5	3	3	5
Optimization Clarity	5	4	4	2
Overall Score	4.8	3.4	2.6	3
Summary	Fully transparent and deterministic; highly explainable within narrow, well-defined scopes.	Focused,task-specific models with clear inputs and outputs; moderately explainable when aided by visualization or SHAP.	Technically analyzable but not human-interpret able; consistent yet semantically opaque.	Highly accessible and human-readable but weakly faithful and inconsistent in reasoning.

Beyond CoT, Improving the Explainability of Large Language Models

Now that we've seen basic LLM explanations—like Chain-of-Thought—are easy to get but not very accurate or detailed, let's look at other methods that can make these explanations better in all areas.

To structure this analysis, we use the DARPA Explainable AI (XAI) framework , which defines three major categories of explainability methods.

Deep Explanation
Techniques that expose or visualize a model's internal reasoning processes.

Interpretable Models
Architectures that are transparent by design, inherently understandable without external tools.

Model Induction
Post-hoc methods that build simplified approximations or external analyses of black-box model behavior.

Because LLMs are built on high-dimensional transformer architectures with billions of parameters, they are not interpretable. Therefore, we focus on Deep Explanation and Model Induction approaches using practical strategies to illuminate how these opaque systems reason, respond, and generalize.

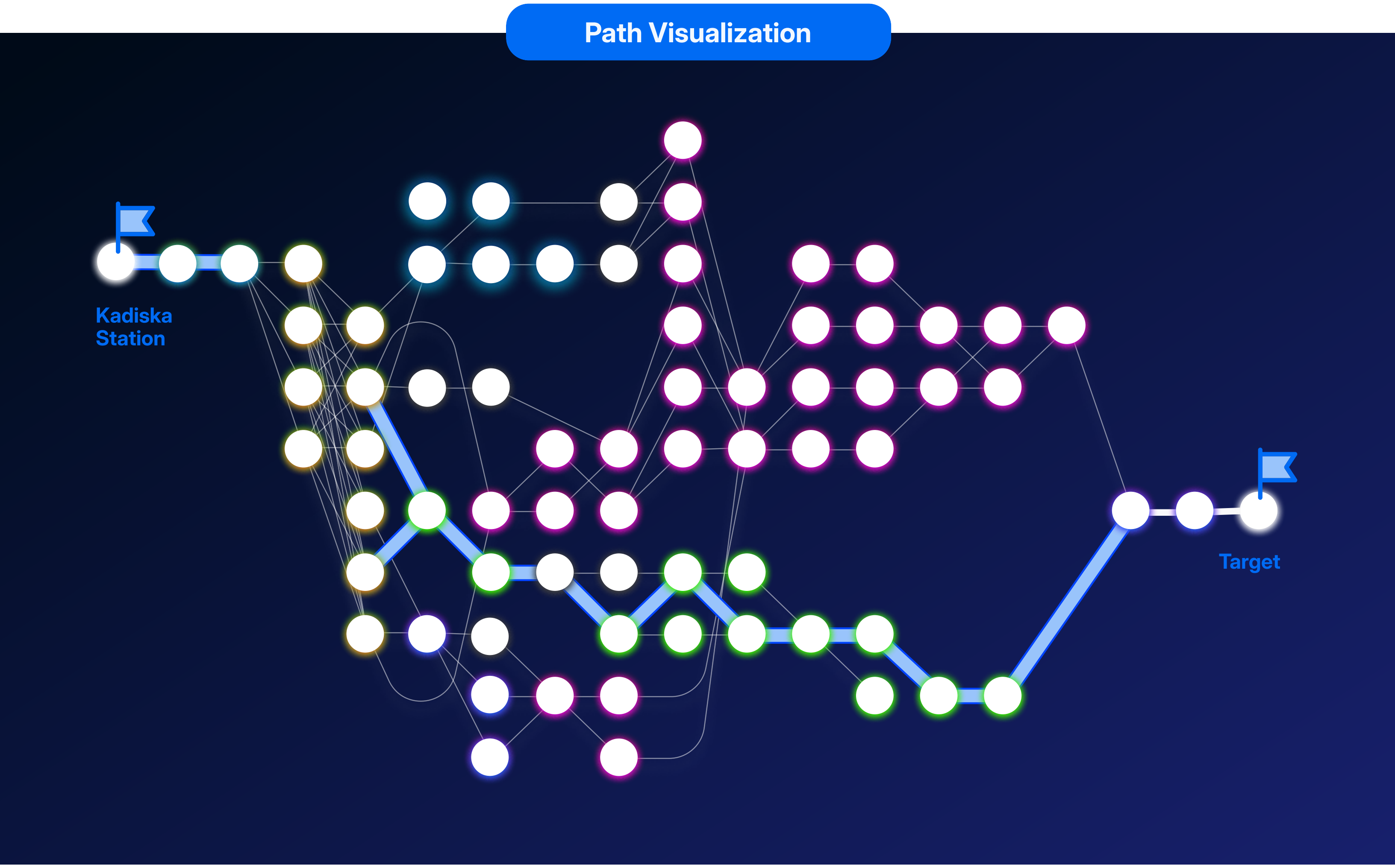
Each of the following techniques represents a different way to reveal, approximate, or contextualize the internal reasoning of LLMs. Then, we will assess how each influences the AI Explainability Scorecard.

Deep Explanation Methods

These approaches analyze or visualize the internal structures of LLMs to make their internal decision pathways more interpretable.

Feature Visualization

Aggregates attention weights or activations across layers to approximate how information flows through the model. However, due to the massive parameter scale of modern LLMs, only coarse, high-level summaries are practical to visualize.



- **Faithfulness** 2.0

  Captures broad input dependencies but fails to represent the true causal reasoning or interactions among billions of parameters.

- **Comprehensibility** 2.0

  Visualizations become abstract and hard to interpret at LLM scale; even simplified flow maps convey limited meaning
  for most users.

- **Consistency** 4.0

  Small differences can arise from floating-point arithmetic and parallel execution order, not inherent randomness in the model.

- **Accessibility** 3.0

  It's easy to create these technically, but making them clear and useful—especially for large models—takes a lot of computer power and often needs extra steps to simplify the data.

- **Optimization Clarity** 2.0

  Provides limited insight into internal reasoning; engineers can observe attention focus areas but not the underlying transformations or decision pathways.

AI Explainability Score Summary

2.6 out of 5

Feature visualization provides minimal interpretability for LLMs due to their vast scale. While it can depict coarse attention flow, these visualizations lack both user-level and optimization clarity, offering little practical value to either audience. The method remains deterministic and easy to implement but conveys limited meaning or trust in how the model reasons. In essence, feature visualization produces attractive visuals, not genuine understanding, its explainability impact is low where it matters most.

AI Models Are Too Big To See

While methods such as integrated gradients, concept activation vectors (CAVs), and feature visualization offer valuable perspectives on model reasoning, they also underscore the limits of explainability at scale with large language models (LLMs). We discussed feature visualization to illustrate this point.

To visually depict even relatively small LLMs that can run on standard hardware are far too complex to interpret at the level of individual neurons or weights, considering they can have 1 to 10 billion parameters. This makes this category of AI Explainability possible, but not effective for LLMs.

Model Induction Methods

These methods treat the LLM as a black box—deriving insights from its input–output behavior rather than its internal mechanics. These methods are model-agnostic, allowing easy reuse as the model evolves or is retrained with minimal additional effort.

We'll focus on two approaches:

- **SHAP (SHapley Additive Explanations)**
  This one of the most popular AI Explainability techniques. We want to illustrate how many AI explainability techniques struggle to scale with large, general-purpose LLMs.

- **Surrogate Models (Monitoring Approach)**
  This is a technique that works in the real world. A more practical method that effectively explains LLM behavior by observing and comparing similar examples.

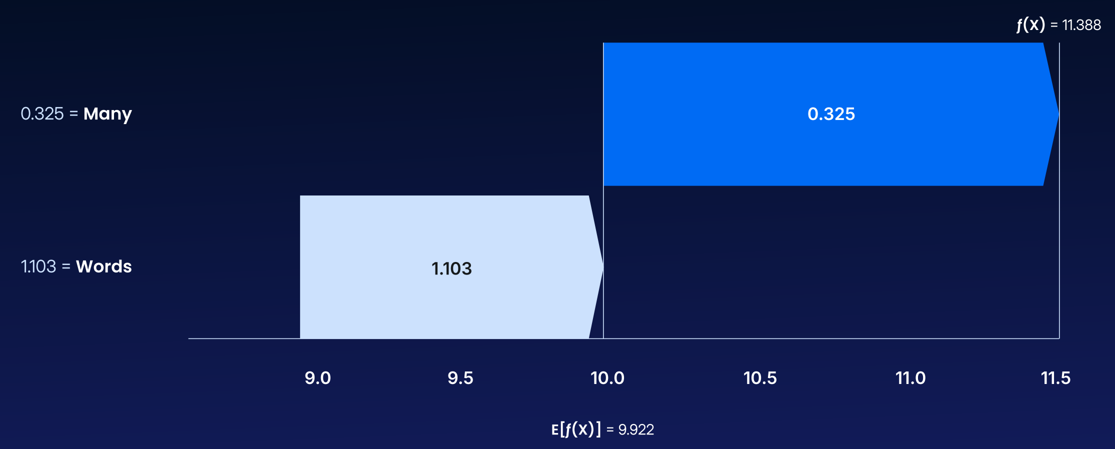
SHAP (SHapley Additive Explanations)

SHAP estimates how much each input token contributes to a model's output. In LLMs, a token is a fragment of text (Depending on the model can be part of a word, such as "un-" or "-ing", or a full word). SHAP assigns each token a "value" representing its marginal contribution to the prediction.

Many words map to one token, but some don't: indivisible.

Traditional SHAP struggles at LLM scale, though optimized variants (e.g., DeepSHAP) can provide partial, layer-level insights at high computational cost.

SHAP Graph



Faithfulness 3.0

SHAP tries to show how each token helped make the prediction by dividing credit based on how they worked together. But since LLMs have billions of parameters, it can only guess at those interactions—so the accuracy of the explanation drops in real use.

Comprehensibility 2.0

Token-level attributions are difficult for non-technical users to interpret, especially since tokens often represent subword pieces rather than intuitive linguistic units.

Consistency 4.0

SHAP gives steady results when the input and model stay the same. Small changes can affect which tokens seem most important, but the explanations are mostly consistent.

Accessibility 1.0

Using SHAP with an LLM takes a lot of computing power. To figure out each token's impact, the model has to run many times—which isn't practical for models this large. There are faster versions of SHAP, but they still use a lot of computing resources. Doing this for every AI interaction would be cost prohibitive.

Optimization Clarity 2.0

SHAP can provide engineers with insight into which tokens most influence predictions, but not how those contributions propagate through deep attention layers. It is technically informative at the surface level yet lacks depth into model internals.

AI Explainability Score Summary

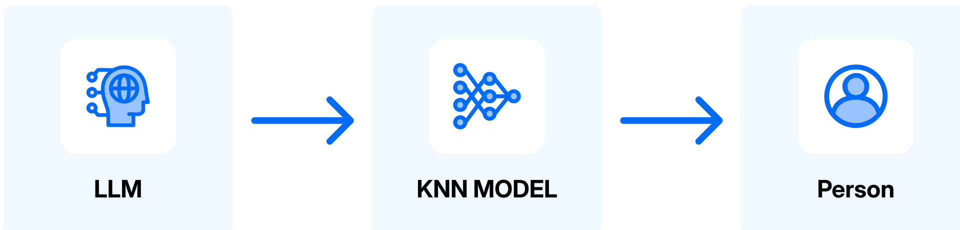
2.4 out of 5

SHAP is based on solid math and gives clear explanations, but it doesn't work well with today's huge LLMs. It's useful for small models or simple cases, but not practical for explaining real-world, large-scale AI systems.

Surrogate Models (Monitoring Approach)

In this setup, K-NN works as a real-time monitoring surrogate, not a replacement for the LLM. It looks at how the model represents full sentences and finds the ones most like it from past examples. This helps explain the model's behavior in a simple way—like saying, "The model answered this way because it saw something similar before and gave a similar response."

A KNN surrogate can monitor LLM behavior post hoc by mapping outputs to similar historical examples—useful for interpretability but not part of the live inference pathway.



- **Faithfulness 3.0**

  The surrogate shows how model inputs and outputs are related, but it can't reveal how the LLM really thinks or makes decisions inside. It reflects what we can see, not the deeper cause behind the choices.

- **Comprehensibility 4.0**

  Users can directly relate the model's output to similar examples, creating clear, context-driven reasoning for both technical and non-technical users.

- **Consistency 5.0**

  If the embeddings, distance rules, and reference data stay the same, K-NN always gives the same results. The same input will always find the same matches and explanations, making it very stable and easy to repeat.

- **Accessibility 2.0**

  Creating a good, varied set of labeled sentence examples takes a lot of time, effort, and money. However, once established, the system is cheap and efficient to operate—generating explanations quickly and with minimal computational cost.

- **Optimization Clarity 5.0**

  Engineers gain precise, actionable insight into model behavior by examining which prior sentences influence current outputs. This clear mapping between examples and responses supports robust debugging, bias analysis, and behavioral transparency.

AI Explainability Score Summary

3.8 out of 5

A sentence-level K-NN monitoring surrogate provides strong, practical explainability for LLMs. It delivers deterministic, intuitive, and technically transparent explanations that bridge expert and user understanding. Although costly to build initially, once operational it offers a scalable, low-cost method for ongoing interpretability grounded in real examples.

Scaling AI Safe & Secure With Surrogates

While methods such as SHAP (SHapley Additive Explanations) and Surrogate Models represent two core approaches to model induction, they reveal distinct trade-offs in explainability for large language models (LLMs).

SHAP, one of the most widely used explainability techniques, illustrates how many traditional methods struggle to scale effectively with large, general-purpose LLMs, where feature interactions are too complex for stable attribution.

In contrast, Surrogate Models—which approximate model behavior through observation and comparison of similar examples—offer a more practical, real-world approach to interpreting LLM decisions.

Additional techniques such as LIME, counterfactual explanations, and influence functions remain valuable areas for exploration but may face similar challenges in adapting to the scale and complexity of modern AI systems.

Other techniques to explore

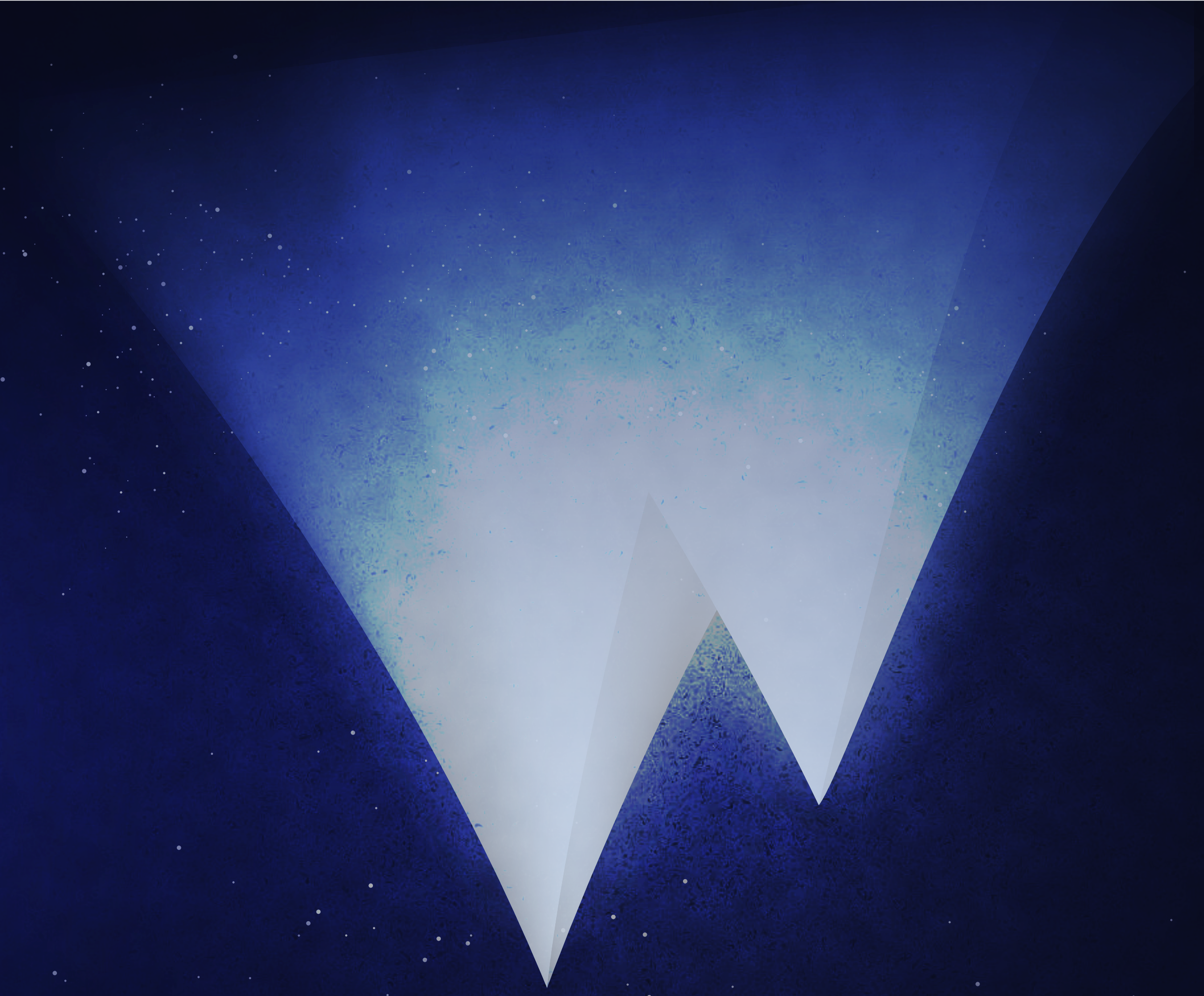
- LIME (Local Interpretable Model-Agnostic Explanations)
- Counterfactual Explanations
- Influence Functions

Trust What You Can Trace

Explainability is the cornerstone of trustworthy and responsible AI—bridging human understanding and machine reasoning. As models grow more complex and autonomous, transparency becomes both harder and more essential. Agentic AI systems, which act and reason independently, cannot be considered trustworthy without robust explainability.

Explainability spans a continuum—from inherently interpretable models like K-NN to opaque large language models that rely on post-hoc interpretability methods. Traditional tools such as SHAP falter at LLM scale, whereas a surrogate model using sentence-level K-NN monitoring offers a promising lens into model behavior.

Ultimately, explainability must integrate interpretability, continuous monitoring, and transparent design to ensure that increasingly agentic AI systems remain understandable, auditable, and aligned with human values.



Aiceberg

AI Security Platform

Get real-time visibility into agentic AI security, so you can detect and respond to threats, and scale with cool confidence. We monitor hundreds of agentic AI risk signals, from prompt injection to data leakage and role impersonation, so you always stay in control. With customizable protection and full visibility, your teams can freeze out threats before they surface.

Book My Demo