big data and cognitive computing

MDPI

Article

Regression-Based Small Language Models for DER Trust Metric Extraction from Structured and Semi-Structured Data

Nathan Hamill and Razi Iqbal *

Department of Computer Science, Central Michigan University, Mount Pleasant, MI 48859, USA; hamil2ns@cmich.edu

* Correspondence: razi.iqbal@ieee.org

Abstract

Renewable energy sources like wind turbines and solar panels are integrated into modern power grids as Distributed Energy Resources (DERs). These DERs can operate independently or as part of microgrids. Interconnecting multiple microgrids creates Networked Microgrids (NMGs) that increase reliability, resilience, and independent power generation. However, the trustworthiness of individual DERs remains a critical challenge in NMGs, particularly when integrating previously deployed or geographically distributed units managed by entities with varying expertise. Assessing DER trustworthiness ensuring reliability and security is essential to prevent system-wide instability. Thisresearch addresses this challenge by proposing a lightweight trust metric generation system capable of processing structured and semi-structured DER data to produce key trust indicators. The system employs a Small Language Model (SLM) with approximately 16 million parameters for textual data understanding and metric extraction, followed by a regression head to output bounded trust scores. Designed for deployment in computationally constrained environments, the SLM requires only 64.6 MB of disk space and 200–250 MB of memory that is significantly lesser than larger models such as DeepSeek R1, Gemma-2, and Phi-3, which demand 3–12 GB. Experimental results demonstrate that the SLM achieves high correlation and low mean error across all trust metrics while outperforming larger models in efficiency. When integrated into a full neural network-based trust framework, the generated metrics enable accurate prediction of DER trustworthiness. These findings highlight the potential of lightweight SLMs for reliable and resource-efficient trust assessment in NMGs, supporting resilient and sustainable energy systems in smart cities.

Keywords: distributed energy resources; networked microgrid; small language models; transformer; regression; neural networks

Academic Editor: Giuseppe Maria Luigi Sarnè

Received: 18 December 2025
Revised: 16 January 2026
Accepted: 22 January 2026
Published: 24 January 2026

1. Introduction

The shift toward sustainable energy infrastructure has driven the transition from centralized to decentralized energy generation. Renewable technologies such as wind and solar now play a central role in electricity production, deployed both in large-scale farms and in decentralized forms as Distributed Energy Resources (DERs). Approximately 4.7 million residential photovoltaic (PV) systems have been installed [1], with 5.4 GW of new distributed solar capacity added in 2024 [2], and about 90,000 distributed wind turbines installed as of 2023 [3]. DERs can operate independently or as components of microgrids, including small and localized electric grids that can function autonomously or in coordination with larger systems. Interconnecting multiple microgrids forms Networked

Microgrids (NMGs), which are designed to enhance reliability, resilience, and utilization of renewable resources while reducing dependency on centralized power generation [4]. Figure 1 illustrates a traditional NMG setup.

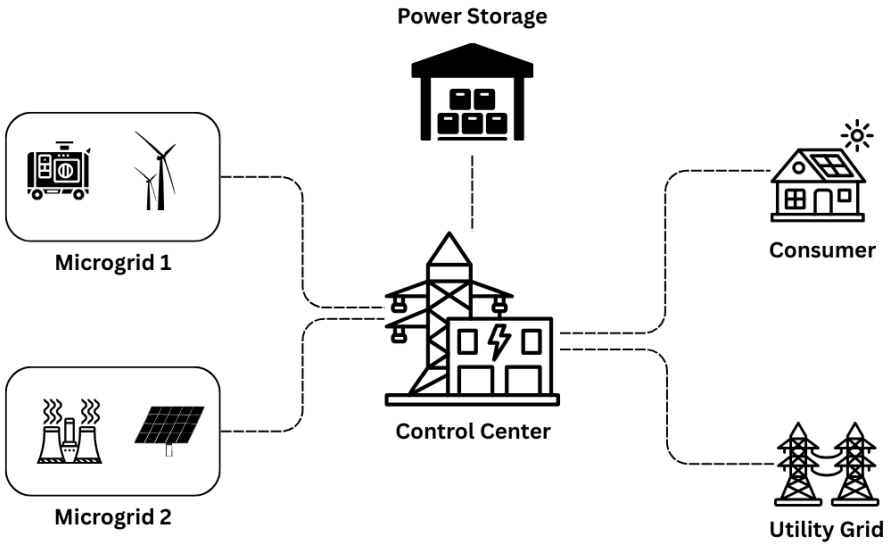


Figure 1. Traditional networked microgrid setup [5].

While the decentralized nature of NMGs provides significant benefits, it also introduces operational challenges that can compromise system reliability and security. As noted in the NREL FAST-DERMS report [6], "numerous optimization and control strategies have been developed for different types of DERs over the years; however, DER management and control are still siloed and at times conflicting between various DER technologies and the distribution management systems." DERs are often geographically dispersed, managed by entities with varying expertise and maintenance standards, and may include previously deployed units with irregular or semi-structured data formats.

These factors create substantial differences in credibility assessment for individual DERs when integrating them into an NMG. Previously deployed DERs require the evaluation of factors like performance degradation, inconsistent logs, and outdated firmware. Geographically dispersed DERs introduce variability including environmental effects like dust accumulation on solar panels or snow on wind turbines in winters. This variability can cause communication latencies that can directly impact availability and productivity of DERs. Finally, management by entities with varying professional capabilities leads to inconsistencies in data quality. For example, expert operators typically provide comprehensive, structured, and secure data, whereas less experienced entities may produce irregular, incomplete, or error-prone datasets that require additional processing.

Insufficient credibility assessment of even a single DER poses severe risks in NMGs due to their interdependence. An untrustworthy DER, whether malfunctioning or compromised, can directly contribute to system-wide instability. For example, sudden surges or drops of power from a degraded inverter can disrupt voltage and frequency stability, interfering with coordinated load balancing across interconnected microgrids. A compromised DER may also serve as an entry point for cyberattacks, enabling malicious actions that can cascade through the network. In extreme cases, these issues can augment local failures into widespread outages in NMGs compared to traditional grids.

Detecting such issues requires robust processing of structured, semi-structured, and unstructured DER data, where poor metrics can reveal malicious tampering, poor maintenance, or hidden faults. This capability is essential for trustworthy integration and preventing propagation of risks across the network.

Hu et al. [7] demonstrated a trust system for microgrids in their RADM framework, protecting against dynamic attacks using structured data. Other works have explored language models for extracting information from heterogeneous IoT sensor data in structured, semi-structured, and unstructured formats. These advances suggest a promising direction: a lightweight trust framework capable of ingesting irregular DER data to generate reliable trust metrics at integration time.

This paper pursues this direction by developing a lightweight trust assessment mechanism that can process heterogeneous (structured and semi-structured) DER data to generate reliable trust metrics prior to NMG integration. This assessment will prevent the inclusion of untrustworthy units that could trigger system-wide instability. The primary motivations of this work are as follows: (1) to handle real-world DER data irregularity arising from diverse deployment histories, geographic disparities, and varying management expertise which are typically not covered by structured-data-only methods; (2) to enable deployment in computationally limited settings by prioritizing efficiency and minimal resource usage; and (3) to provide actionable trust indicators enhancing overall NMG resilience and security. The key contributions are as follows:

1. The creation of a DER synthetic data generation system that uses real statistics as the basis for its data generation.
2. The design of a Small Language Model (SLM) for parsing semi-structured DER data into key trust attributes like availability, reliability, productivity, stability, and reputation.
3. The empirical testing of this regression based SLM against other language models and a fine-tuned model in their direct performance.

2. Literature Review

IoT devices have become increasingly popular, which has increased overall data generation. This includes structured, semi-structured, and unstructured data. With this data comes the possibility of extracting actionable information. LMs have attracted interest in this space, as they have the ability to process and interpret diverse data formats. LMs used in this way could potentially support decision-making and data analysis. This section primarily reviews the use of LMs in IoT system data processing in the existing literature.

Zong et al. [8] examined the use of Large Language Models (LLMs) in IoT environments through three case studies: DDoS (Distributed Denial-of-Service) attack detection, macroprogramming across IoT systems, and large-scale sensor data processing. As a result of this work, they found that GPT models have the potential to efficiently process large amounts of sensor data. They also found that these systems could deliver faster high-quality responses than other conventional machine learning methods. The study also found that few-shot learning performed satisfactorily and that fine-tuning did improve the performance of these systems.

Shirali et al. [9] investigated the use of LLMs for processing raw sensor data streams. The goal of their research was to see if LLMs could be used to identify and categorize sensor data. As part of this work, they developed techniques for how LLMs can abstract events from the data stream. The system they developed focused on extracting useful information from raw, long, and hard-to-read logs. This allowed the system to create a unified event log for subsequent analysis.

Kök et al. [10] explored the usage of LLM Tree of Thought (ToT) reasoning systems for edge, fog, and cloud computing frameworks for IoT applications. These different computing settings have different resource requirements, but they all have the potential to be part of a scalable solution for LLM IoT utilization. For this research, predetermined objectives were set at each tier of computing. The system that the authors presented proved to be useful for predictive maintenance and industrial monitoring tasks.

Ashiwal et al. [11] developed an LLM-based system for extracting vulnerability data from unstructured sources for a software system. The system also converts that information into a machine-readable format for later security assessment. The system developed takes supplier emails as unstructured input, and the LLM processes them to extract relevant information and converts it into a structured CSAF VEX format. This system can, in theory, support software security by extracting useful security information and providing it to a security team in a format they can efficiently analyze.

Wang et al. [12] proposed a framework using LLMs to analyze unstructured data that uses natural language queries. The framework used logical operators using LLMs in order to support semantic reasoning, concurrent semantic cost modeling, and cardinality estimation. The results of this work show that this system could reduce query execution time while maintaining high accuracy in practical data analysis contexts. This system demonstrates that LLMs could be a scalable and efficient solution for these tasks.

Joy et al. [13] investigated the potential of an IoT framework that leveraged microgrid design and operational strategies to develop a resilient energy system. The research primarily examined the context of extreme weather events and grid disruptions and how such a framework could improve resilience in these situations. The study found three key performance indicators (reliability, stability, and flexibility) that reflect performance in critical situations. This study also found that advanced control strategies are critical for these types of situations. The goal of this work was to provide practical guidance for practitioners, policymakers, and researchers in planning tasks.

Aslam et al. [14] developed a trust framework specifically for Social Internet of Things (SIoT) contexts. The primary idea of this work was to use social networking concepts for networks of IoT devices. To do this, the research focused on evaluating the trustworthiness of the services provided, rather than that of the service providers, through aggregating multiple Quality of Service (QoS) metrics. This system recognizes that trustworthy service systems can potentially provide malicious service. This work improves the efficiency of information and resource sharing for these settings.

Madani et al. [15] provided a comprehensive survey of using LLMs in smart grids, including their applications across energy management, anomaly detection, and cybersecurity. The authors emphasized on how LLMs can improve grid intelligence while identifying challenges like high computational requirements, data privacy concerns, and deployability in resource constrained environments. The survey highlights the transformative capabilities of LLMs in grid operations and emphasized on the need of a lightweight language model that can reduce the overall cost of the model, yet preserve the accuracy.

Ferraris et al. [16] provided an extension of TrUStAPIS by introducing IoT trust semantics that are compatible with Web of Things (WoT). They generated recommendations with the assistance of LLMs for integrating and refining existing methodologies. The work demonstrated how LLMs can support early stage IoT system design by producing structured trust, security, privacy and interoperability requirements in a JSON-LD.

Li et al. [17] investigated the security risks associated with deploying LLMs in smart grids by performing systematic threat modeling and experimental validation. They focused on vulnerabilities including data poisoning, prompt injection, and knowledge extraction attacks to illustrate how LLMs can be compromised and put at risk the security and reliability of the whole grid. The study highlighted the importance of trustworthiness evaluation in AI-driven grid components, especially when processing operational data.

A review of the existing literature shows that there has been work along the lines of using language models for interpreting textual data of varying structures and on trust frameworks for DERs and other IoT systems. The systems for processing textual data to generate useful information about a system do not focus on extracting metrics from

DER-type systems. The work conducted on trust metrics for DERs and IoT systems uses structured data. None of these systems incorporate all of these concepts to achieve a unified system that can read in textual data and generate trust metrics relevant to DERs. Filling this gap is the focus of this study.

3. Trust Metrics and Data Generation

The proposed system predicts five core trust metrics: availability, reliability, productivity, stability, and reputation from heterogeneous DER logs. Each metric is bounded in [0, 1], with higher values indicating greater trustworthiness (1 = optimal). Ground truth labels for training/evaluation are computed precisely through their corresponding defining equations, as discussed in Section 3.1. This ensures accurate, interpretable supervision without reliance on noisy real-world annotations. Table 1 provides a concise overview of the metrics, including definitions and ranges.

Table 1. Summary of predicted trust metrics.

Metric	Brief Definition	Range & Interpretation
Availability	Readiness to generate when resource available	[0, 1] (1 = fully available)
Reliability	Operation without forced interruptions	[0, 1] (1 = no forced outages)
Productivity	Conversion of potential to actual output	[0, 1] (1 = perfect efficiency)
Stability	Composite operational effectiveness	[0, 1] (1 = no weaknesses)
Reputation	Communication consistency (cyber-aware)	[0, 1] (1 = expected messages)

3.1. Trust Metric Definitions

To develop an effective system for assessing the trustworthiness of DERs, a practical and essential first step is to implement a dedicated data extraction pipeline. This research adopts a multi-dimensional framework comprised of five core trust metrics: availability, reliability, productivity, stability, and reputation. The definitions for these metrics are drawn from established IEEE standards [18] for electric generating unit reporting and recent advances in distributed trust modeling for energy grids. The IEEE standard 762-2023 [18] is useful as it adjusts metric calculations for Variable Energy Resources (VERs). VERs are power generation units that can vary their electricity output depending on outside factors. This includes systems such as solar or wind generators. VER metrics should reflect aspects of the unit that are constant across many different resource availabilities.

3.2. Metric Descriptions

3.2.1. Availability

Availability measures the extent to which a DER is ready and capable of generating power when needed. Following the IEEE Standard 762-2023 [18] VER specific definition, availability is quantified through the Availability Generation Factor (AGF), which represents the fraction of expected generation (EG) remaining after accounting for generation loss due to unit outages. The AGF is calculated where EG represents expected generation based on available resource conditions and PUG, MUG, and FUG represent planned, maintenance, and forced unavailable generation, respectively. VER-adjusted metrics provide insight into whether a DER can contribute to grid operations during periods when the energy resource (sunlight, wind) is actually available, rather than penalizing the system for conditions beyond its control. Equation for availability is presented in Equation (1), which outputs a value between 0 and 1, where 1 is the best score.

$$AGF = \frac{EG - (PUG + MUG + FUG)}{EG} \quad (1)$$

3.2.2. Reliability

Reliability captures the ability of a DER to operate without forced interruptions during periods when generation is possible. This is measured as the Forced Outage Rate (FOR), where FOR represents the proportion of time a unit was in a forced outage state relative to the time it was in service or on forced outage. Following IEEE 762-2023 VER formulations [18], the FOR calculation is presented in Equation (2), where FOH represents forced outage hours, SH represents service hours, EPRRH represents equivalent partial reserve reduction hours, and RUH represents resource unavailable hours. High reliability values indicate that a DER experiences minimal unplanned disruptions during periods when it could otherwise be generating. This metric is from 0–1 where 0 is the best score, but for consistency, this metric subtracts the FOR from 1 to get a "reliability" score, where 1 is the best score.

$$FOR = \frac{FOH}{SH - EPRRH + FOH + RUH}$$
$$Reliability = 1 - FOR \quad (2)$$

3.2.3. Productivity

Productivity assesses how effectively a DER converts its generation potential into actual energy output. Using the Performance Index (PI) defined in IEEE Standard 762-2023 [18], productivity is calculated, where GAAG represents gross actual aggregate generation and EG represents the expected generation. For VERs, productivity values should be high where conversion efficiency is high. This metric is particularly important for evaluating whether a DER is meeting its anticipated contribution to the microgrid given the actual resource conditions it experienced. This equation also outputs a value between 0 and 1, where 1 is the best score. Equation (3) presents the productivity equation:

$$PI = \frac{GAAG}{EG} \quad (3)$$

3.2.4. Stability

Stability is a metric created specifically for this research. It is a complex aggregated measure of overall DER effectiveness by combining availability, reliability, and productivity into a single composite metric. Stability is calculated by multiplying the three previous metrics together. This multiplicative formulation ensures that weakness in any individual dimension appropriately reduces the overall stability assessment, providing a conservative estimate of trustworthiness that reflects the compound effect of multiple performance limitations. This is a complex metric that is mainly to stress test the later trust metric prediction methods. Equation (4) shows the stability equation:

$$Stability = AGF * FOR * PI \quad (4)$$

3.2.5. Reputation

Reputation introduces a communication-based dimension to trust assessment, measuring the quality and consistency of messages exchanged between the DER and grid control systems. Drawing from the distributed trust model framework developed by Fernando et al. [19], this reputation calculation has been modified in order to fit the 0–1 range. Reputation is calculated based on the count of messages classified as "expected" (CExMSG) relative to the total message count weighted by unexpected messages (CUxMSG), where alpha (α) is an adjustable parameter determining the relative penalty for unexpected communications. Well-behaved DERs are ones that have a low number of unexpected messages. This metric is particularly important for detecting potential security threats, as anoma-

lous communication patterns may indicate compromised systems even when operational metrics appear normal. Equation (5) presents the reputation equation:

$$Reputation = \frac{CExMSG}{CExMSG + (a * CUxMSG)} \quad (5)$$

Together, these five metrics provide a comprehensive characterization of DER trustworthiness, which encompasses operational performance, communication reliability, and overall system stability. The use of VER-specific calculations ensures that the metrics fairly assess renewable energy resources without penalizing them for environmental conditions beyond their control, while the inclusion of communication-based reputation scoring adds a layer of cybersecurity awareness essential for modern networked microgrid environments.

3.3. Synthetic Log Ranges

Synthetic data generation is employed in this research due to several constraints rooted into DER studies. Real-world DER operational logs (e.g., SCADA or JSON) are often proprietary, restricted by policies, or limited in publicly available volume, making large-scale labeled datasets scarce. Moreover, real datasets frequently lack sufficient diversity in data formats, geographic variations, or explicit representations of anomalous behaviors. Synthetic generation addresses these issues by enabling (1) precise control over trust metrics for supervised training, (2) systematic inclusion of normal/anomalous cases, and (3) scalable production of diverse samples. This approach aligns with the established practices in energy systems research, where synthetic datasets are widely used to overcome data access barriers while preserving validity (e.g., NREL's synthetic electric grid models).

For this research, this synthetic data is put into both a structured and a semi-structured data format to replicate the kinds of data that real DER systems would generate. Structured data is data that has a predefined rigid schema that fully defines its structure. Semi-structured data is data that lacks the predefined rigid schema of structured data, but still has some predefined format. The data generated by the ranges shown in Tables 2 and 3 were modified for these log generations, in order to more thoroughly test the model. The ranges obtained from the data were heavily centralized around a few points. This means that, when plugged into the equations, they would create values that were heavily centralized around 0.95. So, the ranges were expanded in order to more thoroughly test the language model's capabilities and make sure that it was not just memorizing the correct value. Also, due to the nature of the generation method, the training data outputs were clamped to make sure that the training value could be 1 at most.

Table 2. Parameter ranges under normal and anomalous conditions for Solar DER [20,21].

Parameter	Methodology Range	Normal Expanded Range	Anomalous Expanded Range
EG (kWh)	1304–4500	2500–3500	3500–4500
PUG (kWh)	0–39.67	0–125	150–350
MUG (kWh)	0–39.67	0–125	150–350
FUG (kWh)	0–39.67	0–125	150–350
FOH (Hours)	0–7.44	0–98	160–893
SH (Hours)	244–450	300–450	200–300
EPRRH_VU (Hours)	0–20	0–15	15–20
RUH (Hours)	270–465	270–350	350–465
GAAG (kWh)	1264–4000	2550–3500	3500–4000
CExMSG	4032–5000	4000–4500	4500–5300
CUnMSG	0–65	0–8	10–65

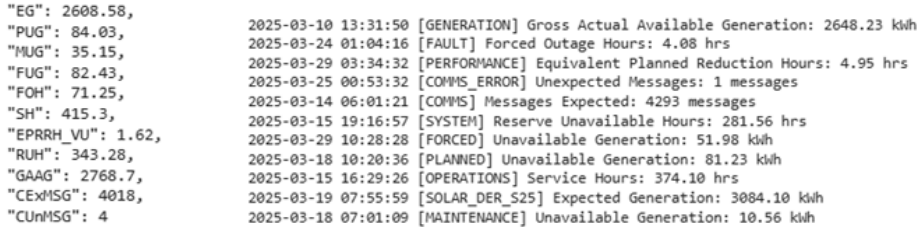
To support effective trustworthiness assessment of DERs, the system should also be capable of generating representative synthetic data that simulates both normal and anomalous operational behavior. Normal logs simulate expected operation, with all key raw metrics staying within predefined acceptable ranges. Anomalous logs, by contrast, feature one or more metrics deliberately falling outside those ranges to mimic faults, misconfigurations, attacks, or degradation [22]. These anomalous metrics will have scores that translate to low-trust metric scores. This approach enables the creation of balanced datasets where the model must learn to distinguish between healthy and degraded system performance. Table 2 shows the ranges for the solar DER data and Table 3 shows the ranges for the wind DER data.

Table 3. Parameter ranges under normal and anomalous conditions for wind DER [23,24].

Parameter	Methodology Range	Normal Expanded Range	Anomalous Expanded Range
EG (kWh)	3748.72–11,715.66	10,000–11,715.66	11,748.72–13,715.66
PUG (kWh)	0–117.16	0–117.16	300–800
MUG (kWh)	0–117.16	0–117.16	300–800
FUG (kWh)	0–117.16	0–117.16	300–800
FOH (Hours)	0–7.44	0–7.44	50–200
SH (Hours)	522.2–699	522.2–699	300–500
EPRRH_VU (Hours)	0–36.35	0–35	36–60
RUH (Hours)	37–172	37–172	100–300
GAAG (kWh)	3636.26–11,715.66	10,000–11,500	1000–4000
CExMSG	4032–4464	4032–4464	3000–3900
CUnMSG	0–10	0–10	10–120

3.4. Synthetic Log Formats

For this research, we used a JSON Log format for the structured data format and a SCADA (Supervisory Control and Data Acquisition) Log format for the semi-structured data. The JSON Log format is a simple dictionary type format that always has some order and structure. It is important to run model tests with this format, as it is used for structured reporting and API communications for these types of systems. These logs represent structured monthly summarization logs. Figure 2a shows an example of the JSON format.



(**a**) JSON Log (**b**) SCADA Log

Figure 2. Comparison of the JSON and SCADA log formats used in the study.

The SCADA Log format is more complex. The lines of the logs are timestamped entries that arrive in non-sequential order, mimicking the real-world asynchronous nature of SCADA system communications. Each log entry contains a timestamp, a system identifier, a raw metric label, and the actual raw metric. The timestamp is a random point across the monitoring period. The system identifier designates the DER system being monitored. The metric label describes which metric is in the line. This format has no fixed schema, is human-readable, and the lines are put in a random order. This allows for the monthly

summarization log to represent a semi-structured log type. Figure 2b shows an example of the SCADA format.

4. Regression Based Small Language Model

For this task of constructing a language model architecture capable of processing structured or semi-structured text data, a decoder transformer architecture [25] was chosen. Transformers have proven to be useful tools for text processing tasks and hence an obvious choice for processing the structured and semi-structured logs. The multi-task regression head was chosen as the output head, as it is a common machine learning concept useful for generating numeric outputs [26]. The system is designed to produce five numeric outputs, each between 0 and 1, representing different aspects of trustworthiness, as illustrated in Figure 3.

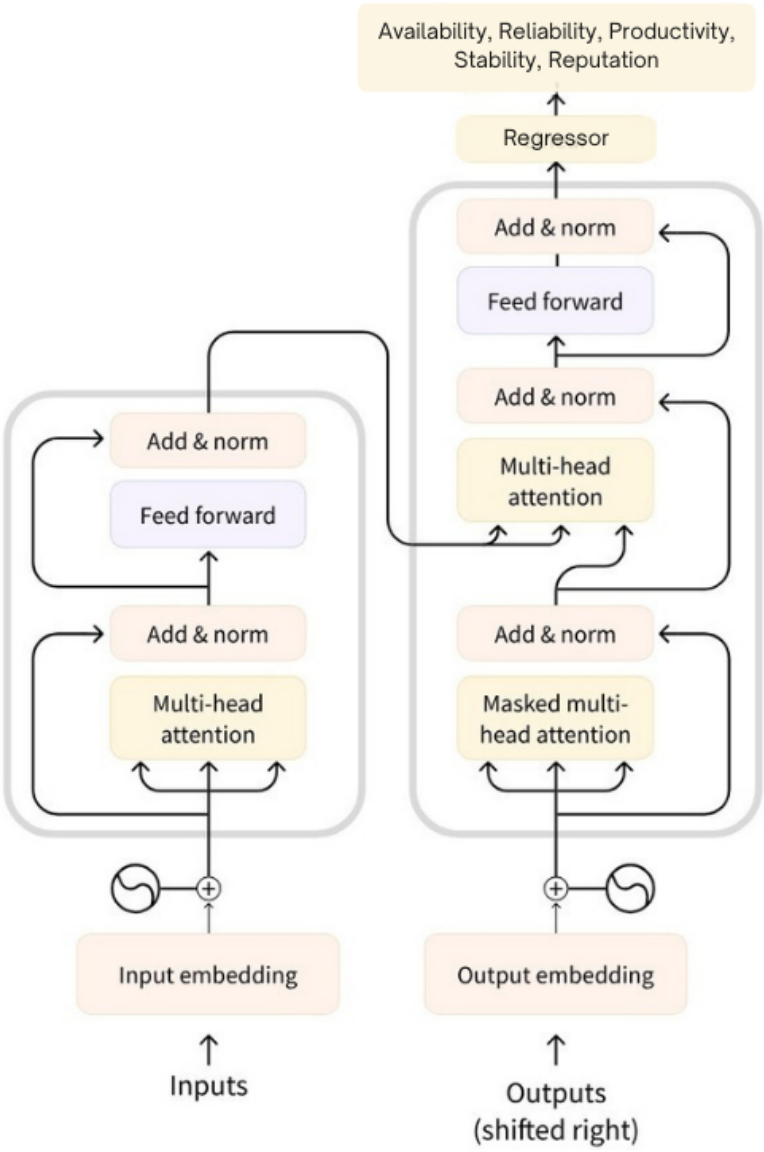


Figure 3. SLM architecture based on original Transformer architecture [25].

The language model architectural components, including the input processing and embedding layer, the multi-head self-attention mechanism, the feed-forward network, and the multi-task regression head, have all been combined to form the predictive model developed by this research. This model is composed of four stacked transformer blocks,

each with four attention heads that operate on 256-dimensional token embeddings [27]. The model takes SCADA or JSON logs, depending on training, and can be up to 512 tokens long. These inputs are integrated with token and positional embeddings to provide the initial representations. These representations are refined through the transformer layer's causal attention patterns. Due to this system, the sequential patterns of the data are factored into the processing. Next, a normalization step is applied to stabilize the representations before the regression head. The regression head itself consists of a linear projection followed by a sigmoid activation function. This linear projection transforms the last token embedding into the five trust metrics. We evaluated a standard large language model setup (without a dedicated regression head), as well as a variant that incorporated pooling attention to better aggregate sequence information. While both approaches showed reasonable performance, the small language model (SLM) equipped with a tailored regression head consistently delivered the highest accuracy across our evaluation metrics. For this reason, we selected the SLM regression model as the final architecture and proceeded with comprehensive testing and validation. Algorithm 1 is adopted from [5].

Algorithm 1 Transformer-Based Log-to-Metric Conversion Framework

Input: Structured to semi-structured log, context size $C = 512$
Output: Structured metrics $\{\text{availability}, \text{reliability}, \text{productivity}, \text{stability}, \text{reputation}\}$
Step 1: Tokenization
Convert log text to token IDs using tokenizer
Resulting token sequence: $\mathbf{t} \in \mathbb{Z}^L$, where $L \leq C$
Step 2: Embeddings
Compute token embeddings: $\mathbf{E}_{\text{tok}} = \mathbf{W}_{\text{tok}} \cdot \mathbf{t}$
Add positional embeddings: $\mathbf{E}_{\text{pos}} = \mathbf{W}_{\text{pos}} \cdot \mathbf{p}$, where $\mathbf{p}$ denotes position indices
Combine and apply dropout: $\mathbf{E} = \text{Dropout}(\mathbf{E}_{\text{tok}} + \mathbf{E}_{\text{pos}})$
Step 3: Transformer Blocks (Repeat for 4 layers)
for each transformer block **do**
    Normalize input: $\mathbf{x}_{\text{norm1}} = \text{LayerNorm}(\mathbf{x})$
    Compute multi-head attention (4 heads): $\mathbf{A} = \text{MultiHeadAttention}(\mathbf{x}_{\text{norm1}})$
    Add residual connection: $\mathbf{x} = \mathbf{x} + \text{Dropout}(\mathbf{A})$
    Normalize again: $\mathbf{x}_{\text{norm2}} = \text{LayerNorm}(\mathbf{x})$
    Apply feedforward network: $\mathbf{F} = \text{GELU}(\mathbf{W}_1\mathbf{x}_{\text{norm2}} + \mathbf{b}_1)$
    Linear projection: $\mathbf{O} = \mathbf{W}_2\mathbf{F} + \mathbf{b}_2$
    Add residual connection: $\mathbf{x} = \mathbf{x} + \text{Dropout}(\mathbf{O})$
end for
Step 4: Final Normalization
Normalize output representations: $\mathbf{h}_{\text{norm}} = \text{LayerNorm}(\mathbf{h})$
Step 5: Last Token Selection
Extract final token embedding: $\mathbf{h}_{\text{last}} = \mathbf{h}_{\text{norm}}[:, -1, :]$
Step 6: Regression Output
Compute metric vector:

$$\mathbf{M} = \text{sigmoid}(\mathbf{W}_{\text{head}}\mathbf{h}_{\text{last}} + \mathbf{b}_{\text{head}})$$

$\mathbf{M} \in [0, 1]^5$ corresponds to $\{\text{availability}, \text{reliability}, \text{productivity}, \text{stability}, \text{reputation}\}$
Step 7: Metric Assignment
metrics $= \mathbf{M}$

The architectural novelty lies in its extreme miniaturization and task-specific design: unlike standard approaches that rely on fine-tuning large pretrained transformers (e.g., with 1–7 billion parameters) for downstream tasks, this SLM is trained entirely from scratch on domain-specific synthetic DER data for direct regression of bounded trust met-

rics from semi-structured logs. This eliminates the substantial parameter overhead and generalization biases inherent in large pretrained models, which are optimized for broad language understanding rather than precise numeric extraction from energy system logs. Comparative experiments (Section 6) confirm that this lightweight from-scratch design outperforms fine-tuned larger models in both accuracy and efficiency for the targeted DER trust assessment.

5. Training and Testing Methodologies

5.1. Training Methodology

The SLM was trained on a NVIDIA T4 GPU. Table 4 illustrates the specifications of this model of GPU [28].

Table 4. Hardware specifications of the training environment.

Specification	Value
Tensor Cores	320 Turing Tensor Cores
CUDA Cores	2560 NVIDIA CUDA Cores
Memory	16 GB
Bandwidth	320+ GB/s

The dataset simulates diversity across

- DER types: Solar and wind (two primary renewable classes, with distinct parameter distributions reflecting real performance differences, e.g., higher variability in wind RUH/FOH).
- Log formats: Structured (JSON: fixed key-value schema for monthly summaries) and semi-structured (SCADA: timestamped, unordered lines mimicking asynchronous real-time telemetry).
- Conditions: Normal (parameters in realistic ranges yielding high trust scores) and anomalous (expanded ranges simulating faults, degradation, or tampering, yielding low scores) balanced for binary trustworthiness distinction.

For each implementation, wind, solar, and JSON, SCADA, the model was trained on 100,000 synthesized samples. These samples were divided into 80,000 training samples and 20,000 validation samples. The training was conducted over ten epochs with a batch size of 32. This training took roughly 50 min to complete. Table 5 presents configuration and hyperparameters of this model.

Table 5. Model configurations and hyperparameters.

Parameter	Value
Vocabulary size	50,257 tokens (GPT-2 tokenizer)
Context length	512 tokens
Embedding dimension	256
Number of attention heads	4
Dropout rate	0.1
Total parameters	16 million

5.2. Comparison Models for Evaluation

To effectively evaluate the utility of this work, the developed model was compared with several pretrained small models. This was to see whether other small, pre-trained models had an intuitive understanding of what makes a system reliable or trustworthy. No publicly available small language models or compact encoders pretrained on domain-specific energy data (e.g., SCADA/JSON logs for DER trustworthiness assessment in

networked microgrids) exist to date. Therefore, we benchmarked against leading general-purpose language models, which represent strong baselines for textual processing tasks. For this work, three models were chosen: Gemma 2 2B [29], Phi 3 3.8B [30], and DeepSeek R1 1.5B [31]. These models were chosen for several reasons. First, these models are developed by industry leaders; thus, it is believed that the lead would be reflected in their performance. These models are also recent: Gemma 2 2B and Phi 3 3.8B were released in 2024, and DeepSeek R1 1.5B was released in 2025. Also, these models are relatively small, with only a few billion parameters. These models are also publicly available with permissive licenses. The goal of these comparisons is to determine whether the models have any pre-existing understanding of the relevant concepts for this work and to compare their understanding with that of the custom model. For this work, DeepSeek R1 1.5B was also fine-tuned. Fine-tuning is the process of training a model after it has already been trained. It is a method for improving the model's effectiveness in a specific task. For each test, it was fine-tuned on 800 samples. The fine-tuning process used the same type of synthetic data that the custom SLM model was trained on for each test. It used the same T4 GPU that was utilized for training the SLM model. We selected 800 samples for fine-tuning the DeepSeek R1 1.5B model, as it takes roughly the same amount of time as training our custom model from scratch on its entire training dataset.

5.3. Testing Methodology

For each of the four applications, SCADA solar, SCADA wind, JSON solar, and JSON wind, we created a balanced test set containing 50 samples: 25 normal (representing trustworthy behavior) and 25 anomalous (simulating untrustworthy or faulty behavior).

6. Results and Discussion

6.1. Performance in Terms of Coefficient of Determination (R^2)

To evaluate the model's performance on the test dataset, we used the coefficient of determination (R^2) as the primary metric. The R^2 (also referred to as correlation) measures the proportion of variance in a dependent variable that is predictable from one or more independent variables in a regression model. R^2 measures how well the regression model captures the variability in the data. A value of 1 indicates a perfect fit (the model explains all the variation), while 0 means that the model performs no better than simply predicting the mean value for every sample. R^2 can dip below zero when the model's predictions are actually worse than using the mean, though this is uncommon with well-trained models. During inference, our standard models sometimes produced trustworthiness scores slightly below 0 or above 1. Since these values fall outside the valid [0, 1] range, we treated them as invalid and excluded them from the final evaluation.

Figures 4 and 5 are bar graphs showing the correlation values of all the models. These figures clearly demonstrate how the SLM and fine-tuned models are significantly more accurate than the other models, with the SLM model being more accurate than the fine-tuned model. The standard models were unable to consistently produce valid outputs. This indicates that these models did not have an understanding of the underlying material. The model's predictions also reveal interesting patterns across log types. It performed noticeably better on the JSON logs than on the SCADA logs. This difference is not surprising: the JSON logs are generally shorter and more structured, which likely makes it easier for the model to capture relevant patterns and produce more accurate trustworthiness scores.

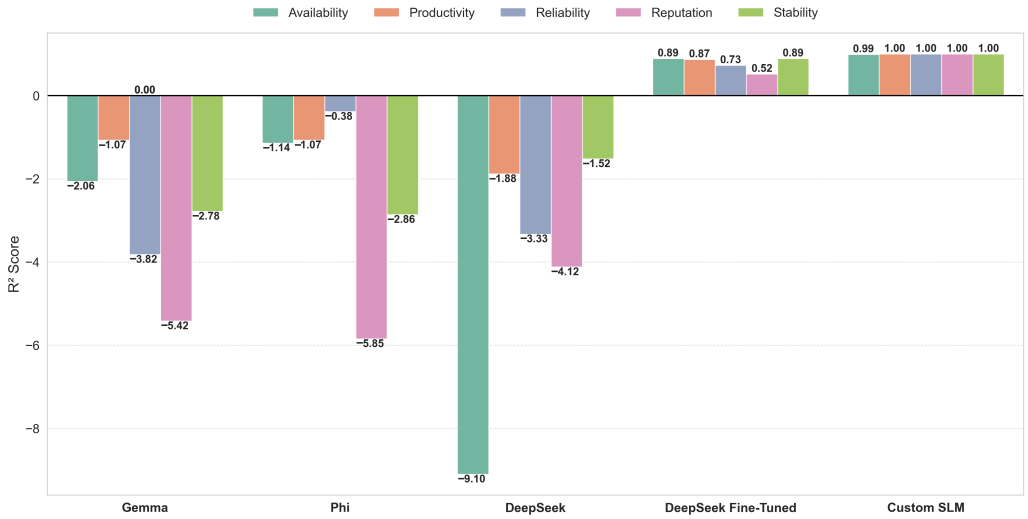


Figure 4. Correlation for JSON Log.

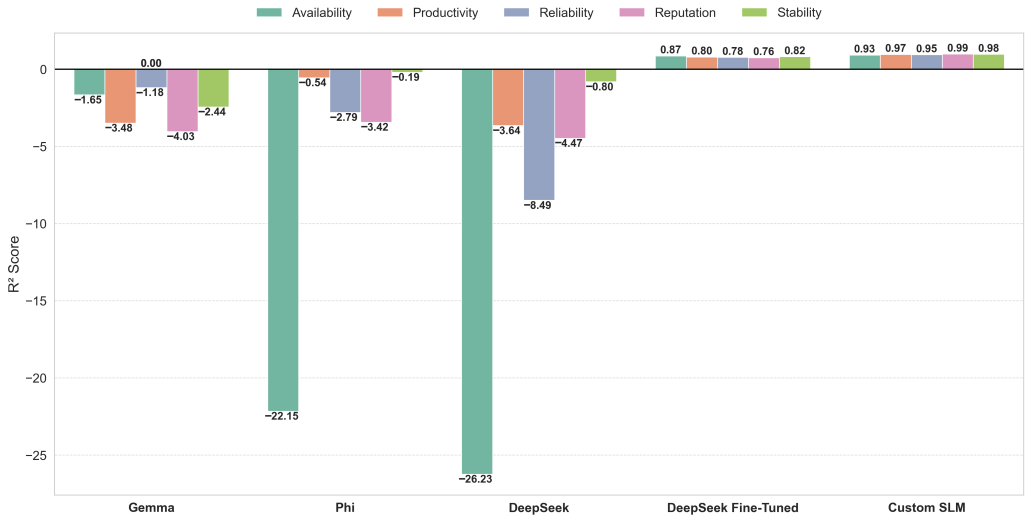


Figure 5. Correlation for SCADA Log.

6.2. Performance in Terms of Mean Absolute Error (MAE)

To complement the R^2 correlation metric we performed Mean Absolute Error (MAE) a standard regression metric measuring the average magnitude of prediction errors on the bounded [0,1] trust score scale. MAE provides an interpretation of absolute accuracy: lower values indicate predictions closer to true metrics, with 0 representing perfect alignment. This is very valuable for trust assessment, where small deviations can impact downstream decisions (e.g., DER integration reliability). Figures 6 and 7 illustrate that the proposed SLM achieves significantly low MAE (0.01–0.04 across metrics and formats), outperforming baselines (0.11–0.51 for larger models) and even the fine-tuned DeepSeek R1 (0.03–0.11). These MAE results confirm the SLM's superior regression performance, enabling highly accurate trust assessment in practical NMG scenarios.

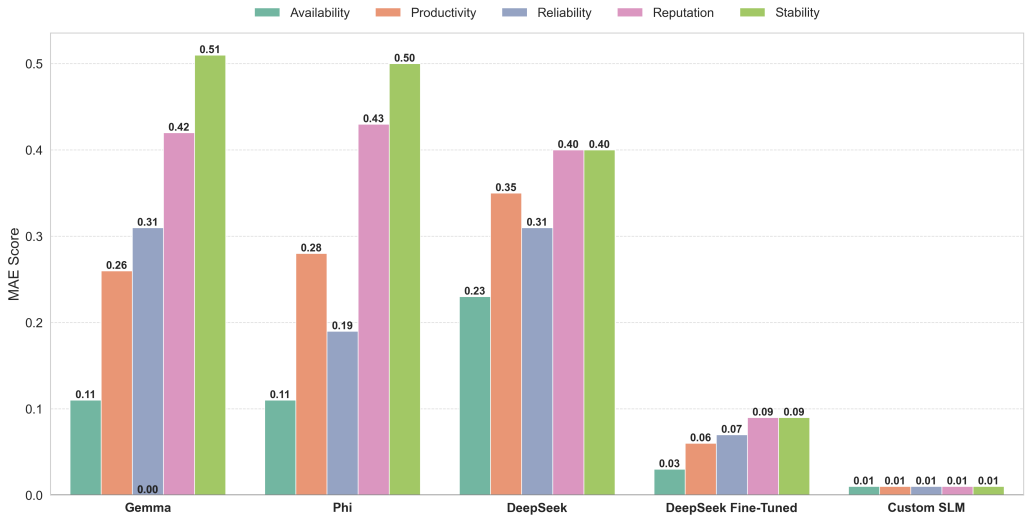


Figure 6. MAE for JSON Log.

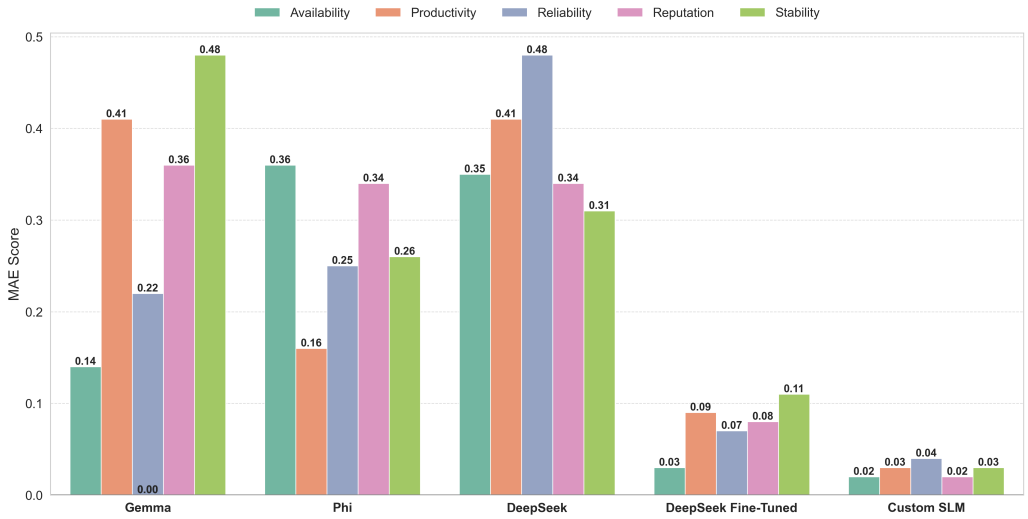


Figure 7. MAE for SCADA Log.

6.3. Model Size and Resource Efficiency

Table 6 clearly shows that our SLM is dramatically smaller and more resource-efficient than the other models considered. Compared to the next smallest model, DeepSeek R1 (1.5B parameters), a custom SLM with only 16 million parameters, requires less than 6% of the GPU memory and under 2% of the disk space. This substantial reduction in size and resource demands makes the SLM particularly well-suited for deployment in resource-constrained environments while still delivering strong performance on the trustworthiness evaluation task.

Table 6. Comparison of the models' resource requirements.

Model	Disk Space	GPU RAM (FP16)	Parameters Info
DeepSeek R1	3.4 GB	3.7 GB	1.5 Billion
gemma-2-2b-it	9.8 GB	5.2 GB	2.6 Billion
Phi-3-mini-4k-instruct	7.2 GB	7.9 GB	3.8 Billion
DeepSeek R1 Fine-tuned	3.4 GB	3.7 GB	1.5 Billion
SLM	64.6 MB	224 MB	16 million

These results, seen in Figure 8, show that while the fine-tuned model is somewhat comparable to the SLM, it requires far more resources than the SLM, and is thus more suited for tasks in resource limited contexts.

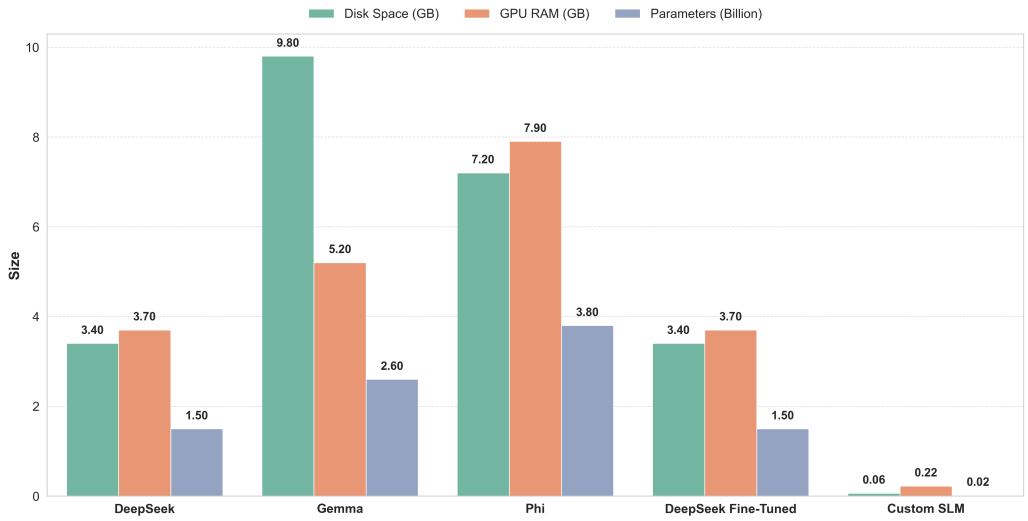


Figure 8. Model sizes, memory requirements, and parameter counts.

6.4. Model Performance

This section presents scatter plots that compare the performance of three predefined large language models, the fine-tuned model, and our custom SLM across the five trust metrics: Availability, Reliability, Productivity, Stability, and Reputation. The three predefined models, Gemma, Phi, and DeepSeek, showed very similar behavior in our tests, so to keep things clear and avoid repetition, only the results for Gemma as a representative example are displayed here. Each plot is based on predictions for 50 test SCADA logs, with red dots marking anomalous (potentially untrustworthy) logs and green dots showing unanomalous (normal). These visualizations make it easy to see how well each model separates the two classes, and they highlight the SLM's stronger ability to distinguish between normal and anomalous cases.

These scatter plots in Figure 9 demonstrate poor predictive capabilities. There are many points in the data that are label as having similar values despite being comprised of wildly different data inputs. The model does get some values correct, but it also gets more values significantly wrong. It is also worth noting that there are points missing from this data, because some of the model's predictions were outside of the accepted [1, 0] range. These scatter plots in Figure 9 demonstrate that Gemma 2 does not have an understanding of these metrics and that it is highly unreliable for these predictions.

The scatter plots in Figure 10 demonstrate decent predictive capabilities. The model is generally able to give a value that is close to the actual output. Some of the data points are very close, within 0.1 of the actual value, and none of the data points are more than 0.3 away from the actual value. The scatter plots in Figure 10 demonstrate that the fine-tuned model has a moderate understanding of these metrics and that it is decently reliable for these predictions.

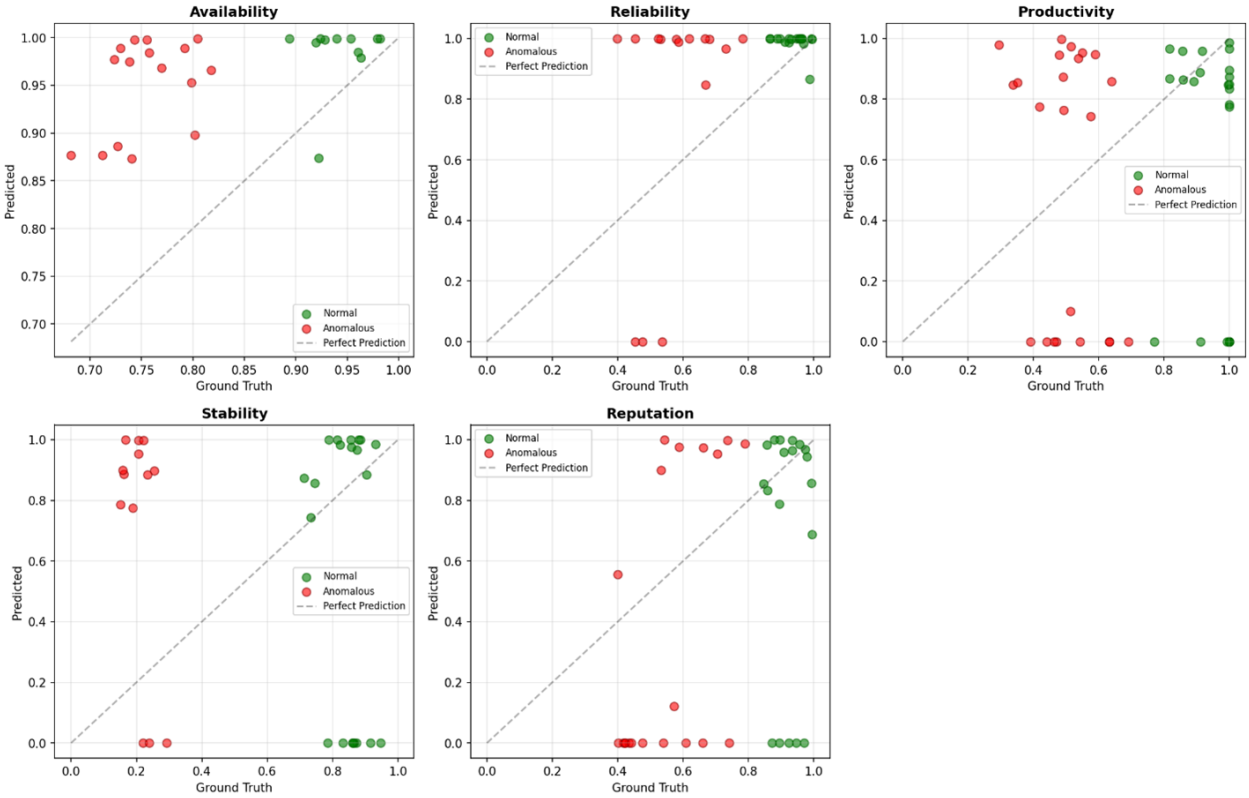


Figure 9. Gemma 2 prediction scatter plots for SCADA data.

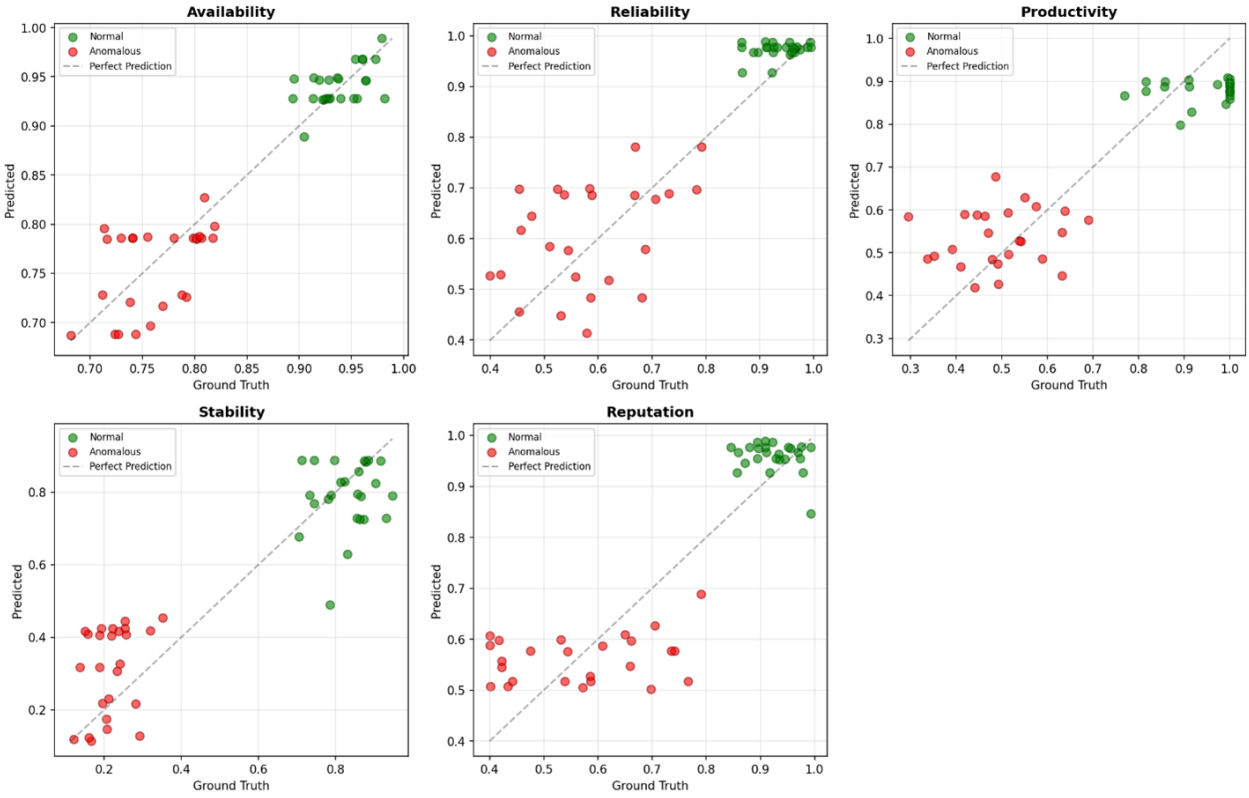


Figure 10. DeepSeek fine-tuned prediction scatter plots for SCADA data.

The scatter plots in Figure 11 demonstrate high predictive capabilities. The model is reliably able to give a value that is close to the actual output. Some of the data points are very close, within 0.1 of the actual value, and none of the data points are more than 0.15 away

from the actual value. These scatter plots in Figure 11 demonstrate that the SLM has excellent understanding of these metrics and that it is highly reliable for these predictions.

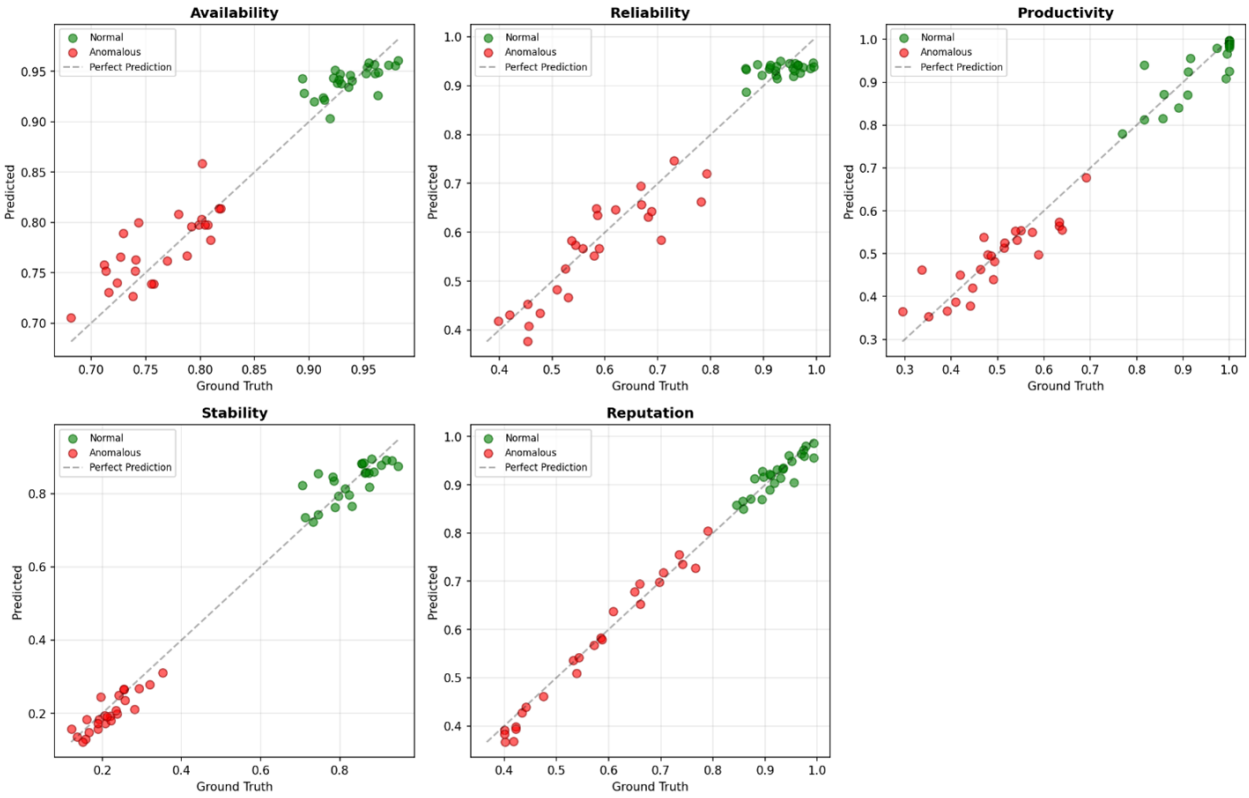


Figure 11. SLM prediction scatter plots for SCADA data.

6.5. Neural Network Test

To evaluate the effectiveness of the generated trust metrics, a basic neural network was trained as a binary classifier for both solar and wind DER data. The network architecture consists of layers with five input nodes (corresponding to the five trust metrics), followed by hidden layers of 64, 32, and 16 neurons, and a single output node. It was trained on 25,000 data points using the same expanded metrics employed in data generation, but applied directly to the trust equations rather than embedded in logs. This approach allowed the neural network to achieve 100% accuracy on correctly labeled inputs, serving as a simplified testbed for trust evaluation. The metrics produced by the SLM and other models were then fed into this classifier to assess whether a neural network-based system could reliably predict DER trustworthiness from the provided values.

The results in Figure 12 show that the SLM and fine-tuned model were able to generate trust metrics that were accurate enough for the neural network to reliably predict if the data belonged to a normal or anomalous DER. The SLM data put into the neural network was 100 percent accurate for all of the tests. The fine-tuned data put into the neural network was almost as accurate, with the wind JSON model being slightly less accurate. The standard models were not accurate, with most of the test results hovering around 50 percent accurate (shown be dotted red line in Figure 12). There were some outputs that were a bit more accurate, but they were at best 15 percent better than guessing randomly, and thus they could not be part of a reliable system. These results demonstrate that the SLM and the fine-tuned model could be part of a full trust evaluation framework, noting that the SLM performs slightly better while requiring a significantly smaller number of computational resources to do so as compared to the fine-tuned model.

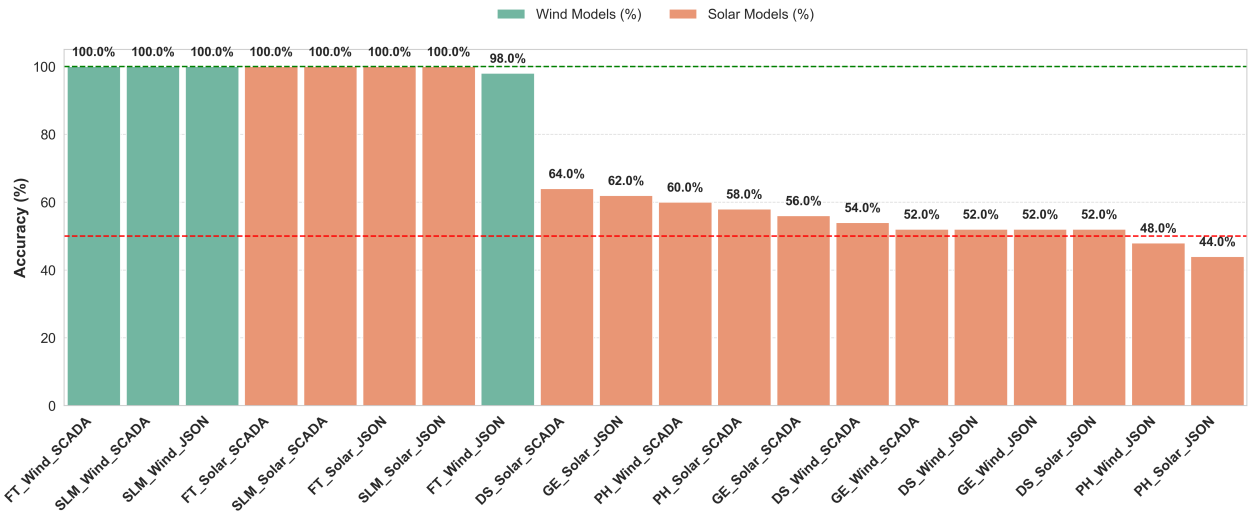


Figure 12. Neural network output accuracy.

6.6. Resource Efficiency and Real-Time Processing

Table 6 and Figure 8 highlight the substantial efficiency advantages of the proposed SLM over larger pretrained models. Compared to the baseline models (DeepSeek R1 1.5B, Gemma-2-2B, and Phi-3-mini-3.8B), the SLM demonstrates improved efficiency in the following specific metrics critical for real-time processing of DER data in resource-constrained environments:

- Disk space (deployment size): 64.6 MB versus 3.4–9.8 GB (a reduction of 98–99%), enabling rapid model loading, storage on low-capacity edge devices, and over-the-air updates without excessive bandwidth or storage demands.
- Runtime memory footprint (GPU RAM in FP16): 224 MB versus 3.7–7.9 GB (a reduction of approximately 94–97%), preventing out-of-memory errors on typical microgrid edge hardware and eliminating the need for cloud offloading, which would significantly reduce latency in real-time scenarios.
- Computational complexity: With only 16 million parameters (versus 1.5–3.8 billion, a 94–99% reduction), the SLM requires fewer floating-point operations per forward pass. This directly translates to significantly lower inference latency per DER log, supporting real-time trust metric generation during dynamic DER integration without delaying NMG control decisions.

6.7. Summary

These results demonstrate that the custom SLM achieves the highest accuracy among the evaluated models. It effectively processes semi-structured input data to generate highly precise trust metric predictions, a capability not matched by the Gemma 2, Phi 3, or DeepSeek R1 models. The fine-tuned model performs reasonably well, but falls short of the SLM's accuracy while requiring substantially greater computational resources. Consequently, the development of this lightweight SLM proves highly valuable, as it efficiently converts structured or semi-structured DER data into reliable trust metrics, ideally suited for subsequent neural network-based trust evaluation in resource-constrained environments.

7. Conclusions

This study develops a computationally efficient system that transforms structured and semi-structured DER operational data into accurate trust metrics: Availability, Reliability, Productivity, Stability, and Reputation. The transformer-based SLM combined with a regression head achieved correlation values of 0.93–1.00 across and MAE values of

0.01–0.04 across all metrics and formats, while using only 16 million parameters, 64.6 MB of disk space, and 200–250 MB of memory, which are significantly lower requirements than larger models such as DeepSeek R1, Gemma 2, and Phi 3. These results confirm that a small, specialized language model can deliver high accuracy with a fraction of the resource demand, making it highly suitable for deployment in resource-constrained NMG environments.

Despite the promising performance, the current evaluation is limited to synthetically generated data. Real-world SCADA logs are rarely publicly available, which restricted validation to controlled scenarios. In future work, two main directions will be pursued. First, the system will be extended to handle fully unstructured data (beyond the semi-structured logs tested here). Preliminary results with variable log sequences indicate that the current attention-based architecture is well positioned for this step. Second, the SLM will be integrated into a complete neural network-based trust framework. Once this integration is complete, the full system will enable end-to-end trust evaluation of DERs in operational NMGs.

With the growing deployment of distributed energy resources in smart cities, such lightweight and interpretable trust assessment systems will play an important role in protecting grid reliability and supporting sustainable urban energy infrastructure.

Author Contributions: Conceptualization, R.I. and N.H.; methodology, N.H.; software, N.H.; validation, N.H. and R.I.; formal analysis, N.H.; investigation, N.H.; resources, R.I.; data curation, N.H.; writing—original draft preparation, N.H.; writing—review and editing, R.I.; visualization, N.H.; supervision, R.I.; project administration, N.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Acknowledgments: AI was used in this research. GPT-5 was used for improving some text in the Abstract and Conclusion sections. Claude Sonnet 4.5 was used for some coding work.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Geocaris, M. The Future of Distributed Wind in the United States: Considerations for Unlocking Terawatt-Level Potential. National Renewable Energy Laboratory. 2022. Available online: https://www.nrel.gov/news/detail/features/2022/the-future-of-distributed-wind-in-the-united-states-considerations-for-unlocking-terawatt-level-potential (accessed on 10 November 2025).
2. Behrsin, I. The State(s) of Distributed Solar—2024 Update. Institute for Local Self-Reliance. 2025. Available online: https://ilsr.org/article/energy-democracy/the-states-of-distributed-solar-2024-update/ (accessed on 10 November 2025).
3. Solar Energy Industries Association. Solar Market Insight Report Q4 2023. SEIA. 2023. Available online: https://seia.org/research-resources/solar-market-insight-report-q4-2023/ (accessed on 10 November 2025).
4. Oak Ridge National Laboratory. Networked Microgrids. Available online: https://www.ornl.gov/content/networked-microgrids (accessed on 11 November 2025).
5. Iqbal, R.; Hamill, N.S. Interpretable SLM-Driven Trust Framework for Smart Cities: Managing Distributed Energy Resources in Networked Microgrids. *Smart Cities* **2025**, *8*, 186. [CrossRef]
6. Ding, F.; MacDonald, J.; Pratt, A.; Hagerman, J.; Liu, W.; Ogle, J.; Saha, A.; Baggu, M. *Federated Architecture for Secure and Transactive Distributed Energy Resource Management Solutions (FAST-DERMS): System Architecture and Reference Implementation*; Technical Report NREL/TP-5D00-81566; National Renewable Energy Laboratory: Golden, CO, USA, 2022.

7. Hu, Y.; Xian, X.; Jin, Y. RADM: A Risk-Aware DER Management Framework with Real-Time DER Trustworthiness Evaluation. In Proceedings of the 12th ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS), Nashville, TN, USA, 19–21 May 2021; pp. 77–86. [CrossRef]
8. Zong, M.; Hekmati, A.; Guastalla, M.; Li, Y.; Krishnamachari, B. Integrating Large Language Models with Internet of Things: Applications. *Discov. Internet Things* **2025**, *5*, 2. [CrossRef]
9. Shirali, M.; Sani, M.F.; Ahmadi, Z.; Serral, E. LLM-Based Event Abstraction and Integration for IoT-Sourced Logs. In Proceedings of the International Conference on Business Process Management, Krakow, Poland, 1–6 September 2024.
10. Kök, İ.; Demirci, O.; Özdemir, S. When IoT Meet LLMs: Applications and Challenges. In Proceedings of the IEEE International Conference on Big Data (BigData), Washington, DC, USA, USA, 15–18 December 2024.
11. Ashiwal, V.; Finster, S.; Dawoud, A. LLM-Based Vulnerability Sourcing from Unstructured Data. In Proceedings of the IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Vienna, Austria, 8–12 July 2024.
12. Wang, J.; Feng, J. Unify: An Unstructured Data Analytics System. In Proceedings of the IEEE 41st International Conference on Data Engineering (ICDE), Hong Kong, China, 19–23 May 2025.
13. Joy, D.B.; Bo, N.J.; Zheng, M. A Framework for Resilient Community Microgrids: Review of Operational Strategies and Performance Metrics. *Energies* **2025**, *18*, 405. [CrossRef]
14. Aslam, M.J.; Din, S.; Rodrigues, J.J.P.C.; Ahmad, A.; Choi, G.S. Defining Service-Oriented Trust Assessment for Social Internet of Things. *IEEE Access* **2020**, *8*, 206459–206473. [CrossRef]
15. Madani, S.; Tavasoli, A.; Khoshtarash Astaneh, Z.; Pineau, P.-O. Large Language Models Integration in Smart Grids. *Energy Rep.* **2025**, *14*, 4445–4460. [CrossRef]
16. Li, J.; Yang, Y.; Sun, J. Risks of Practicing Large Language Models in Smart Grid: Threat Modeling and Validation. *arXiv* **2025**, arXiv:2405.06237. [CrossRef]
17. Ferraris, D.; Kotis, K.; Kalloniatis, C. Enhancing TrUStAPIS Methodology in the Web of Things with LLM-Generated IoT Trust Semantics. In *Information and Communications Security (ICICS 2024)*; Lecture Notes in Computer Science; Katsikas, S., Xenakis, C., Kalloniatis, C., Lambrinoudakis, C., Eds.; Springer: Singapore, 2025; Volume 15056, pp. 125–144. [CrossRef]
18. *IEEE Std 762-2023*; IEEE Standard Definitions for Use in Reporting Electric Generating Unit Reliability, Availability, and Productivity. IEEE: New York, NY, USA, 2023.
19. Fernando, N.S.; Acken, J.M.; Bass, R.B. Trust Model Measurements for the Energy Grid of Things. In Proceedings of the IEEE/PES Transmission and Distribution Conference and Exposition (T&D), Anaheim, CA, USA, 6–9 May 2024; pp. 1–5. [CrossRef]
20. National Renewable Energy Laboratory. PVWatts Calculator. Available online: https://pvwatts.nrel.gov/ (accessed on 3 December 2025).
21. NOAA Global Monitoring Laboratory. NOAA Solar Calculator. Available online: https://gml.noaa.gov/grad/solcalc/ (accessed on 11 August 2025).
22. Jordan, D.; Marion, B.; Deline, C.; Barnes, T.; Bolinger, M. PV Field Reliability Status—Analysis of 100,000 Solar Systems. *Prog. Photovolt. Res. Appl.* **2020**, *28*, 739–754. [CrossRef]
23. Gonzalez, E.; Stephen, B.; Infield, D.; Melero, J.J. Using High-Frequency SCADA Data for Wind Turbine Performance Monitoring: A Sensitivity Study. *Renew. Energy* **2019**, *131*, 841–853. [CrossRef]
24. Huskey, A.; Bowen, A.; Jager, D. *Wind Turbine Generator System Duration Test Report for the Gaia-Wind 11 kW Wind Turbine*; Technical Report NREL/TP-500-49069; National Renewable Energy Laboratory: Golden, CO, USA, 2010.
25. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. In *Advances in Neural Information Processing Systems*; Curran Associates, Incorporated: Red Hook, NY, USA, 2017.
26. Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I. Improving Language Understanding by Generative Pre-Training. OpenAI. 2018. Available online: https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf (accessed on 10 November 2025).
27. Raschka, S. *Build a Large Language Model (From Scratch)*; Manning: Shelter Island, NY, USA, 2024.
28. NVIDIA. NVIDIA T4 Tensor Core GPUs for Accelerating Inference. Available online: https://www.nvidia.com/en-us/data-center/tesla-t4/ (accessed on 13 November 2025).
29. Gemma Team. Gemma 2: Improving Open Language Models at a Practical Size. *arXiv* **2024**, arXiv:2408.00118. [CrossRef]
30. Microsoft. Phi-3 Technical Report: A Highly Capable Language Model Locally on Your Phone. *arXiv* **2024**, arXiv:2404.14219. [CrossRef]
31. DeepSeek-AI; Guo, D.; Yang, D.; Zhang, H.; Song, J.; Wang, P.; Zhu, Q.; Xu, R.; Zhang, R.; Ma, S.; et al. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. *arXiv* **2025**, arXiv:2501.12948.