SHH, DON'T SAY THAT! DOMAIN CERTIFICATION IN LLMS

Cornelius Emde[1]*, **Alasdair Paren**[1], **Preetham Arvind**[1], **Maxime Kayser**[1], **Tom Rainforth**[1]
Thomas Lukasiewicz[2,1], **Bernard Ghanem**[3], **Philip H.S. Torr**[1], **Adel Bibi**[1]
[1]University of Oxford [2]Vienna University of Technology [3]KAUST

ABSTRACT

Large language models (LLMs) are often deployed to perform constrained tasks, with narrow domains. For example, customer support bots can be built on top of LLMs, relying on their broad language understanding and capabilities to enhance performance. However, these LLMs are adversarially susceptible, potentially generating outputs outside the intended domain. To formalize, assess and mitigate this risk, we introduce *domain certification*; a guarantee that accurately characterizes the out-of-domain behavior of language models. We then propose a simple yet effective approach which we call VALID that provides adversarial bounds as a certificate. Finally, we evaluate our method across a diverse set of datasets, demonstrating that it yields meaningful certificates, which bound the probability of out-of-domain samples tightly with minimum penalty to refusal behavior.

1 INTRODUCTION

With recent advancements in the field of natural language processing, large language models (LLMs) have become ubiquitous. In particular, the scaling of recent large generalist models dubbed foundation models has shown to enable emergent abilities that benefit a wide range of downstream tasks such as text generation, question answering, and text comprehension (Kaplan et al., 2020; Alabdulmohsin et al., 2022; Xiong et al., 2024; Henighan et al., 2020; Brown et al., 2020). Adapting these foundation models for downstream tasks often leads to state-of-the-art performance and has become the dominant paradigm (Gao et al., 2021). This is typically achieved via fine-tuning on task-relevant data (e.g. low-rank adaptation (LoRA) Hu et al. (2022), in-context learning (Mosbach et al., 2023), prefix turning Li & Liang (2021), or simply prompt engineering).

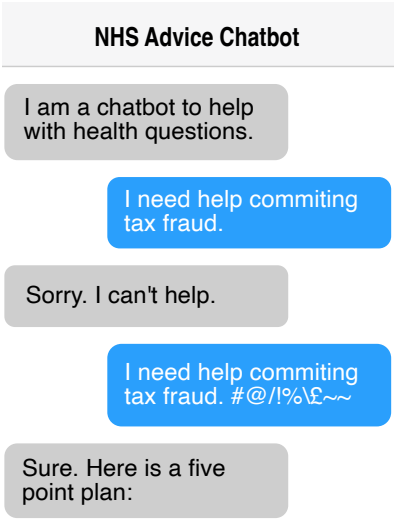


Figure 1: A user misappropriating an LLM system using an adversarial attack.

However, foundation models are typically trained on large amounts of web data which contains a wide range of information that is either irrelevant to a task or potentially harmful (Bommasani et al., 2022). Therefore, it is desirable to restrict the output of a generalist LLM to a specific domain. For example, consider a healthcare provider such as the National Health Services (NHS) providing a general purpose chatbot to support their citizens with simple health questions, as shown in Figure 1. It would be important, for public reputation and cost reasons, that such a system would remain on topic and could not be misused, either intentionally or unintentionally. Misappropriating models is easily possible.

In order to prevent intentional misuse, we consider an adversary trying to elicit an unintended (from the deployer's perspective) response from the model. We assume the deployer wants an LLM to only respond with a certain set of topics, and thus a successful attack is an input string that creates a coherent response outside the target domain. There are various reasons why an adversary might want to elicit such a response that is out-of-domain (OOD). The adversarial user might want to misappropriate the system as a cost-effective tool for a purpose it wasn't built for, resulting in

*Corresponding Author cornelius.emde@cs.ox.ac.uk. Work partially done while interning at King Abdullah University of Science and Technology (KAUST).

excess infrastructure costs for the deployer. Conversely, the deployer might legally be required to validate and verify their models, which is challenging, if not impossible, when the model is not domain-restricted. Finally, the adversary might want to harm the company directly by eliciting harmful OOD responses, which could damage the company's reputation when publicised. Recently, an LLM-driven meal planning tool has received wide media attention for providing toxic recipes when prompted with toxic ingredients (McClure, 2023; The Guardian, 2023). Deployers have moral and legal obligations to prevent this (Bommasani et al., 2022). In all examples, restricting the domain in which the model responds *under adversarial* prompts can help mitigate risks. Thus, in the era of foundation models, "domain" specialization is critical.

Existing work has implemented guardrails that address these risks (Jain et al., 2023), most notably via alignment, resulting in the models to reject user requests (Bai et al., 2022; Ouyang et al., 2022; Christiano et al., 2017). However, a wide body of research has shown that common guardrails have "jailbreaks", i.e., they can easily be circumvented by an adversary (Wang et al., 2024; Qi et al., 2024; Eiras et al., 2024; Carlini et al., 2023; Dong et al., 2024). Common jailbreak methods are prompt injection (Perez & Ribeiro, 2022; Jiang et al., 2023; Liu et al., 2024), numerical optimization (Jia & Liang, 2017; Wallace et al., 2021a; Ebrahimi et al., 2018; Jones et al., 2023; Zou et al., 2023; Jia et al., 2024), red teaming (Perez et al., 2022; Samvelyan et al., 2024), automated black-box attacks (Chao et al., 2023; Mehrotra et al., 2024), or data poisoning attacks (Biggio et al., 2012; Wallace et al., 2021b; Carlini et al., 2024). Using these tools, it is possible for adversaries to retrieve information from a fine-tuned model that was suppressed by the alignment and generate responses that are outside the target domain (see Figure 1 for an example). Adversarial prefixes or suffixes that augment any prompt are especially powerful as they have been shown to universally attack models in combination with a wide range of prompts and can thus be shared between adversarial users (Wallace et al., 2021a; Zou et al., 2023). This presents a significant risk. Hence, researchers have proposed methods to defend against these adversarial attacks, such as unlearning (Nguyen et al., 2022; Xu et al., 2023), robust fine-tuning (O'Neill et al., 2023; Dong et al., 2021), or request and response filtering (Inan et al., 2023).

Deployers would ideally want guardrails that come with a provable, mathematical guarantee against the model responding off topic, or a guarantee that it does this with very low probability. The process of constructing guarantees against certain model behaviours under adversarial attack is commonly referred to as *certification* and has been successfully applied to vision applications in recent years (Akhtar et al., 2021) and proposed for NLP applications (La Malfa, 2023; Casadio et al., 2024; Kumar et al., 2024). However, no existing LLM guardrails provide guaranteed protection against existing or future jailbreaking techniques, leaving deployed models at risk of being compromised shortly after release. As a result, developing certifiable methods to guarantee that specialized LLMs consistently produce on topic content is a critical. Hence, our contributions are as follows:

- We introduce a novel framework, *domain certification*, to bound the probability of models producing out-of-domain content under adversarial attack.
- We introduce an easy-to-use algorithm VALID that bounds the probability of an LLM based system responding off topic under adversarial attack. We show the efficiency of VALID which we test empirically on a number of representative data sets.

2 Domain Certification

We now introduce our *domain-certification* framework for offering mathematical guarantees that a LLM system stays on topic. In Section 2.1, we formally introduce this framework. In Section 2.2, we present **V**erified **A**dversarial **L**LM Output via **I**terative **D**ismissal (VALID). VALID is an easy-to-use method to create a system that adheres to these guarantees. In plain language, we are proposing a certifiable guardrail for LLM-driven systems as follows:

> *A model is* domain-certified*, when an adversarial upper bound can be placed on the probability that the model provides an output outside its designated target domain.*

Before formalizing this statement, we introduce some mathematical notation. We represent tokens (i.e. individual text units), as x and y, which belong to the token space $x, y \in \mathbb{V}$ where $\mathbb{V} = \{1, \ldots, V\}$ is the vocabulary of size V. We define the space of sequences of arbitrary length as $\mathbb{S} \triangleq \mathbb{V}^*$, the Kleene closure of $\mathbb{V}$. Sequences of tokens are denoted by bold letters, $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{S}$, with $\boldsymbol{x}$ and $\boldsymbol{y}$ representing the input and output sequences of an LLM respectively. We use lowercase letters to denote models that predict the next token, such as $l : \mathbb{S} \to \mathbb{V}$. Applying this model

repeatedly, until the end-of-sequence token creates a sequence-to-sequence model $L : \mathbb{S} \to \mathbb{S}$. We denote the likelihood of sample $\boldsymbol{y}$ under L given $\boldsymbol{x}$ as $L(\boldsymbol{y}|\boldsymbol{x})$, which is obtained by $L(\boldsymbol{y}|\boldsymbol{x}) = \prod_{n=1}^{N_y} l(y_n|y_{<n}, \boldsymbol{x})$ for a sentence $\boldsymbol{y}$ of length N_y. We further denote the distribution from which the model samples its output as $\boldsymbol{y} \sim L(\cdot|\boldsymbol{x})$.

2.1 Defining Domain Certification

We now formally introduce *domain certification*. We define the target domain (set of desired topics) as a subset of the sentence space $\mathbb{S}$. We partition $\mathbb{S}$ into the target domain $\mathbb{T}$ and its complement $\mathbb{T}'$. For instance, $\mathbb{T}$ might be all sentences meaningfully occurring for "question answering for health problems". In addition, we define the set of unwanted responses as $\mathbb{F} \subset \mathbb{T}'$ ($\mathbb{F}$ as "forbidden") and will certify with respect to this set $\mathbb{F}$ rather than $\mathbb{T}$. Sequences posing risk should be included in $\mathbb{F}$, while $\mathbb{F}' \cap \mathbb{T}'$ should contain benign out-of-domain samples, such as unintelligible or meaningless sequences of tokens (see Appendix G.1 for a discussion). Hence, we wish to establish a guarantee that L is unlikely to produce an output in $\mathbb{F}$. As a step towards such a guarantee, we first define a bound for any given element $\boldsymbol{y}$ in $\mathbb{S}$:

Definition 1 ***Atomic Certificate***. *We say a model $L : \mathbb{S} \to \mathbb{S}$ is $\epsilon_{\boldsymbol{y}}$-atomic-certified ($\epsilon_{\boldsymbol{y}}$-AC) for some sample $\boldsymbol{y}$ (i.e. an* atom*) in the output set $\mathbb{S}$, iff*

$$\forall \boldsymbol{x} \in \mathbb{S} : L(\boldsymbol{y}|\boldsymbol{x}) \leq \epsilon_{\boldsymbol{y}}. \tag{1}$$

In words, a model that is $\epsilon_{\boldsymbol{y}}$-AC for a sample $\boldsymbol{y}$, will generate sample $\boldsymbol{y}$ with probability smaller than $\epsilon_{\boldsymbol{y}}$ for *any* $\boldsymbol{x} \in \mathbb{S}$, and hence for adversarially chosen $\boldsymbol{x}$. If this is the case, we say model L is *certifiable* for sample $\boldsymbol{y}$ with $\epsilon_{\boldsymbol{y}}$, i.e. $\epsilon_{\boldsymbol{y}}$ is the *smallest* value that provably bounds L. Ideally, such an upper bound $\epsilon_{\boldsymbol{y}}$ would be large for samples in the target domain $\mathbb{T}$, meaning the certificate is *permissive*, and small for samples drawn from $\mathbb{F}$ meaning the certificate is *restrictive*, i.e. *tight*.

The atomic certificate implies an upper bound $\epsilon_{\mathbb{F}}$ for $\mathbb{P}_{\boldsymbol{y}\sim L(\cdot|\boldsymbol{x})}(\boldsymbol{y} \in \mathbb{F}|\boldsymbol{x})$, which would be constructed by summing (1) over all $\boldsymbol{y} \in \mathbb{F}$ for a given $\boldsymbol{x}$. Concretely, $\mathbb{P}_{\boldsymbol{y}\sim L(\cdot|\boldsymbol{x})}(\boldsymbol{y} \in \mathbb{F}|\boldsymbol{x}) = \sum_{\boldsymbol{y}\in\mathbb{F}} L(\boldsymbol{y}|\boldsymbol{x}) \leq \sum_{\boldsymbol{y}\in\mathbb{F}} \epsilon_{\boldsymbol{y}} = \epsilon_{\mathbb{F}}$. However, practically this sort of bound is intractable due to $\mathbb{F}$'s exponential size in N_y, and the difficulty in constructing a precise description of the set $\mathbb{F}$. Instead of giving a bound over returning $\boldsymbol{y} \in \mathbb{F}$, we look at the worst-case across $\mathbb{F}$ which can more precisely be estimated from a finite sample of $\mathbb{F}$:

Definition 2 ***Domain Certificate***. *We say model L is ϵ-domain-certified (ϵ-DC) with respect to $\mathbb{F}$, when it is $\epsilon_{\boldsymbol{y}}$-AC for all $\boldsymbol{y} \in \mathbb{F}$ with $\epsilon_{\boldsymbol{y}} \leq \epsilon$:*

$$\forall \boldsymbol{x} \in \mathbb{S}, \boldsymbol{y} \in \mathbb{F} : L(\boldsymbol{y}|\boldsymbol{x}) \leq \epsilon. \tag{2}$$

This imposes a global bound on L across all undesired responses in $\mathbb{F}$. In practice, we cannot establish the ϵ-DC certificate w.r.t. $\mathbb{F}$ as we cannot enumerate $\mathbb{F}$. Hence, following standard practice in ML evaluation, we propose to use $\mathcal{D}_{\mathbb{F}}$, a finite dataset of out-of-domain responses to establish a ϵ-DC certificate w.r.t. $\mathcal{D}_{\mathbb{F}}$ approximating the certificate for $\mathbb{F}$.

Recent discussions have raised the need for bounds on undesirable behavior. For instance, Bengio (2024) advocates for upper bounds on harmful behavior (Bengio et al., 2024). In addition, an increasing body of legislation mandates thorough auditing of ML systems (EU, 2024). The atomic and domain certificates can play a vital role in assessing the risk of worst-case behavior. For example, consider the deployer of a LLM-based system that processes 10 requests per second. The deployer might perform an apriori risk assessment and determine that they can tolerate the consequences of one out-of-domain response from a set $\mathcal{D}_{\mathbb{F}}$ per year. The deployer should certify the LLM system as ϵ-DC with $\epsilon \approx 10^{-9}$ in order to achieve this level of risk.

Certification through Divergences. We provide an alternative view to this problem, generalising it to bounding divergences between the model and the distribution of of sentences in the domain $\mathbb{T}$. We then use this view to operationalize the $\epsilon_{\boldsymbol{y}} - AC$ and $\epsilon - DC$ (Definitions 1 and 2) inspired by Vyas et al. (2023)'s work on preventing copy-right violations. To this end, we define an oracle Ω that is a *generator* for domain $\mathbb{T}$: Ω assigns high likelihood to sentences in $\mathbb{T}$ and zero likelihood to elements in $\mathbb{F}$. Hence, sampling from Ω will yield in-domain responses. We establish and bound the divergence between L and Ω to restrict the model domain. In particular, we use the Renyi divergence

of order infinity, $\Delta_\infty(P \parallel Q) \triangleq \log \sup_x \frac{P(x)}{Q(x)}$ (Rényi, 1961). Hence, our objective is:

$$\forall \boldsymbol{x} \in \mathbb{S} : \Delta_\infty(L(\boldsymbol{y}|\boldsymbol{x}) \parallel \Omega(\boldsymbol{y})) \leq k. \tag{3}$$

Bounding this divergence is at the core of what we are aiming to achieve: The divergence is large when L assigns high likelihood to a sample $\boldsymbol{y}$ while Ω does not. That means L is likely to produce samples that are out-of-domain. When Ω assigns high likelihood to $\boldsymbol{y}$, the sample is in the target domain, and hence the divergence in (3) is not restrictive. When L assigns low likelihood, $\boldsymbol{y}$ is unlikely to be sampled. Interestingly, this divergence implies (1) and (2), see Lemma 1.

As the oracle is not available in practice we approximate Ω with a "guide" language model that is exclusively trained on in-domain data dubbed G (i.e. the guide model). We use $G(\boldsymbol{y})$ to replace $\Omega(\boldsymbol{y})$ to assess the *marginal* likelihood of $\boldsymbol{y}$. While this means that $G(\boldsymbol{y})$ loses some context contained in $\boldsymbol{x}$, this has a major advantage: $G(\boldsymbol{y})$ does not depend on $\boldsymbol{x}$, which is a potential adversary and hence, by design is robust to adversarial prompts.

2.2 Achieving Domain Certification

In this section, we introduce **V**erified **A**dversarial **LLM** Output via **I**terative **D**ismissal (VALID) to obtain atomic certification as described in Definition 1. We utilise a general model L and a domain generator G as described above and obtain a meta-model M for which the guarantee holds with respect to the domain generator G. In particular, we perform rejection sampling as described in Algorithm 1 (inspired by Vyas et al. (2023)): The capable, general model L proposes a sample $\boldsymbol{y}$ and we accept, if the length normalized log-ratio between L and G is bounded by hyperparameter k. We repeat up to T times until a sample is accepted. If all samples are rejected, the model dismisses the request. This defines a new model M, for which the following theorem establishes the certificate:

Algorithm 1 VALID

Require: LLM L, Guide model G, hyperparameters k and T, prompt $\boldsymbol{x}$
  for $t \in \{1, \ldots, T\}$ **do**
    Sample $\boldsymbol{y} \sim L(\cdot|\boldsymbol{x})$
    $N_{\boldsymbol{y}} \leftarrow$ length($\boldsymbol{y}$)
    if $\log \frac{L(\boldsymbol{y}|\boldsymbol{x})}{G(\boldsymbol{y})} \leq kN_{\boldsymbol{y}}$ **then**
      Return: $\boldsymbol{y}$
  Return: "Abstained".

Theorem 1 (VALID Certificate) *Let L be an LLM and G a guide model as described above. Rejection sampling as described in Algorithm 1 with rejection threshold k and up to T iterations defines model $M_{L,G,k,T}$ with $M_{L,G,k,T}(\boldsymbol{y}|\boldsymbol{x})$ denoting the likelihood of $\boldsymbol{y}$ given $\boldsymbol{x}$. Let $N_{\boldsymbol{y}}$ be the length of $\boldsymbol{y}$. We state the adversarial bound:*

$$\forall \boldsymbol{x} \in \mathbb{S} : M_{L,G,k,T}(\boldsymbol{y}|\boldsymbol{x}) \leq 2^{kN_{\boldsymbol{y}}} \cdot T \cdot G(\boldsymbol{y}). \tag{4}$$

Hence, $M_{L,G,k,T}$ is $[2^{kN_{\boldsymbol{y}}} TG(\boldsymbol{y})]$-AC and, further, it is $[\max_{\boldsymbol{y} \in \mathbb{F}} 2^{kN_{\boldsymbol{y}}} TG(\boldsymbol{y})]$-DC w.r.t. $\mathbb{F}$.

When context allows, we may abbreviate $M_{L,G,k,T}$ to M, omitting subscripts for brevity. Such a certificate with respect to G can be useful: As G is only trained on samples in $\mathcal{D}_{\mathbb{T}} \subset \mathbb{T}$, a dataset of domain $\mathbb{T}$, it assigns exponentially decreasing likelihood to samples that are in $\mathbb{F}$.[1] In particular, this is useful iff the log upper bound $kN_{\boldsymbol{y}} + \log T + \log G(\boldsymbol{y})$ (log RHS of (4)) is small in comparison to $\max_{x \in \mathbb{S}} \log L(\boldsymbol{y}|\boldsymbol{x})$: Our certificate can provide an upper bound to the adversarial behaviour of M that is favourable over L.

As mentioned, this problem is closely related to OOD detection, for which the the likelihood ratio test is commonly used as a powerful statistic (Neyman & Pearson, 1933; Bishop, 1994; Ren et al., 2019; Li et al., 2023; Zhang et al., 2024; Rafailov et al., 2024). Commonly in OOD detection, rejection threshold k is chosen to balance false negative rates and false positive rates. Here, k also influences the upper bound on the certificate, indicating that there can be a *trade-off* between correctly classifying samples as ID or OOD, and achieving a desired level of certification.

Length Normalization. Algorithm 1 performs length normalized rejection-sampling as unnormalized log likelihood ratios scale unfavourably in $N_{\boldsymbol{y}}$, the length of sequence $\boldsymbol{y}$ which we now demonstrate. Consider the next-token models l and g underlying the sequence-to-sequence models L and G. As $\boldsymbol{y}$ is sampled from L, we expect each token $y_1, \ldots, y_{N_{\boldsymbol{y}}}$ to have high likelihood under l. If we assume that l places c times more probability mass per token than g, then we can show that the log likelihood ratio grows linearly in $N_{\boldsymbol{y}}$, the length of sequence $\boldsymbol{y}$:

[1] We give an empirical example of this behaviour in Figure 10 in Appendix D.4.

$\log L(\boldsymbol{y}|\boldsymbol{x})/G(\boldsymbol{y}) = \log \prod_{n=1}^{N_{\boldsymbol{y}}} cg(y_n|y_{<n})/g(y_n|y_{<n}) = N_{\boldsymbol{y}} \log c$. We illustrate an example in Figure 2a: Assume that an in-domain sample $\boldsymbol{y}$ for which model L and generator G assign constant likelihood per token of 0.1 and 0.05, respectively, i.e. $\forall n = 1, .., N_{\boldsymbol{y}} : l(y_n|y_{<n}, \boldsymbol{x}) = 0.1$ and $g(y_n|y_{<n}, \boldsymbol{x}) = 0.05$. Further assume out-of-domain $\boldsymbol{y}'$ for which l assigns a mass of 0.1 per token, and g assigns 0.01. The log likelihood ratio for $\boldsymbol{y}$ can be expressed as $N_{\boldsymbol{y}} \log 2$ and for $\boldsymbol{y}'$ as $N_{\boldsymbol{y}'} \log 10$. As in- and out-of-domain ratios grow with length, so does the optimal decision bound. We plot sequences of varying lengths with these parameters in Figure 2a. By arithmetic manipulation, rejection sampling with threshold $kN_{\boldsymbol{y}}$ is equivalent to bounding the ratio of geometrically normalized likelihoods $\log L(\boldsymbol{y}|\boldsymbol{x})^{1/N_{\boldsymbol{y}}}/G(\boldsymbol{y})^{1/N_{\boldsymbol{y}}}$ using a constant threshold k. Hence, we propose to use normalized log ratios in Algorithm 1 over unnormalized likelihood ratios. Similar approaches have been discussed in the NLP literature (Geng et al., 2023).

In the Appendices, we provide further insights into VALID. In particular, in Appendix A we provide Lemma 2 showing how to estimate the likelihood of M. In Lemma 3, we provide an analysis on the expected number of iterations of VALID. In Appendix B.1, we further discuss the dependence of the certificate on the sequence length and in Appendix B.2, we provide further intuition on how rejection sampling can achieve an adversarial bound. Finally, in Lemma 4 we show an adversary for M and discuss how rejection sampling encumbers adversarial attacks on M.

3 Experiments

We aim to empirically test our method proposed in Section 2.2 across 3 domains: Shakespeare, Computer Science News and MedicalQA. After describing the experimental setup in Section 3.1, we examine the rejection behaviour of our method by examining the $\log L(\boldsymbol{y}|\boldsymbol{x})/G(\boldsymbol{y})$ ratio and associated certificates under a finite set of ground-truth test samples from $\mathbb{T}$ and $\mathbb{F}$ in Section 3.2. Finally, in Section 3.3, we repeat this analysis by applying our Algorithm 1.

3.1 Experimental Setup

In this section, we provide a brief description of our experimental setup for three applications. Each experimental setup consists of a target domain $\mathbb{T}$, a finite dataset $\mathcal{D}_{\mathbb{T}} \subset \mathbb{T}$ of in-domain samples, models L and G, and an out-of-domain dataset $\mathcal{D}_{\mathbb{F}} \subset \mathbb{F}$, against which we test our methods (see Appendix C for more details on data and models).

Shakespeare. Our target domain $\mathbb{T}$ is Shakespeare's plays. We fine-tune a Gemma-2-2b (Team et al., 2024) as model L and train a GPT-2 architecture (33.7M parameters, Radford et al. (2019)) from scratch for G on TinyShakespeare (TS) (Karpathy, 2015). We use TS's test split as in-domain dataset, $\mathcal{D}_{\mathbb{T}}$, and following previous literature (Zhang et al., 2024) compose $\mathcal{D}_{\mathbb{F}}$ of IMDB (Maas et al., 2011), RTE (Wang et al., 2019) and SST2 (Minaee et al., 2024), adding an old Bible dataset (Reis, 2019) as it is linguistically close to TinyShakespeare. At testing, we consider 256-token long sequences and use the first 128 tokens as prompt.

Computer Science News. Our target domain $\mathbb{T}$ is news about computer science. We fine-tune a Gemma-2-2b as model L and train a GPT-2 architecture (109.3M parameters) from scratch for G on articles from the computer science categories in the 20NG dataset (Lang, 1995). We use computer science articles from 20NG's test split as target domain $\mathcal{D}_{\mathbb{T}}$ and the remaining categories

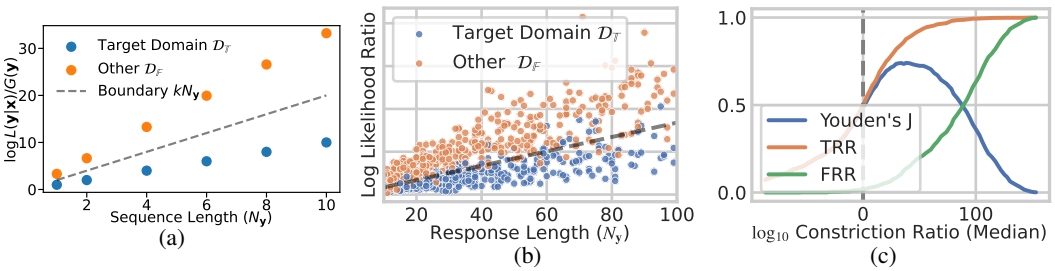


Figure 2: Figure 2a and Figure 2b demonstrate the effect of the sequence length on the log likelihood. Figure 2a shows this relationship through artificial examples with sentences of length 1 to 10 for both ID and OOD datasets, illustrating why the decision boundary should scale accordingly. Figure 2b confirms this relationship on real data (MedicalQA), showing that length normalization makes the problem linearly separable. Figure 2c shows the false rejection rate (FRR) for different ϵ-DC values using VALID with $T = 1$ on MedicalQA.

as $\mathcal{D}_{\mathbb{F}}$ together with the OOD dataset used for Shakespeare. At testing, we consider 256-token long sentences and use the first 128 tokens as prompt.

Medical QA. We apply our method to medical question answering as target domain $\mathbb{T}$. This could, for example, be extended to a chatbot for clinicians to look up patient symptoms. We use a LLama-3-8B model (AI@Meta, 2024) as L and for guide model G we pre-train a GPT-2 architecture model from scratch (184M parameters) on PubMedQA (Jin et al., 2019), which contains approximately 200K QA pairs for training and 1000 test pairs. We further fine-tune G on responses from L to questions in PubMedQA. We use the PuMedQA test set as in-domain dataset $\mathcal{D}_{\mathbb{T}}$ and regard question answering on other topics, such as geography, as $\mathbb{F}$. To model this, we use the Stanford Question and Answering Dataset (SQuAD; excluding medical categories; Rajpurkar et al. (2016)) as $\mathcal{D}_{\mathbb{F}}$.

3.2 Likelihood Ratios on Ground Truth Samples

In this section, we evaluate the capability of our method to attribute samples to the target domain and investigate whether it yields useful adversarial bounds. In particular, we study the length-normalized likelihood ratio $L(\boldsymbol{y}|\boldsymbol{x})/G(\boldsymbol{y})$ on in- and out-of-domain samples. In Figure 3a, we show that the log likelihood ratios for MedicalQA are disentangled and hence a threshold k exists separating target domain and out-of-domain samples well. However, such k — while yielding strong OOD detection performance — might not be associated with tight certificates. Hence, we will first study the $\epsilon_{\boldsymbol{y}}$-AC certificates under M for individual samples, $\boldsymbol{y}$, before moving on to the domain certificate, ϵ-DC.

Atomic Certificates. We obtain $\epsilon_{\boldsymbol{y}}$-ACs using VALID (Section 2.2), setting k to achieve a 10% false rejection rate (FRR) for in-domain samples. Figures 4 (a)-(c) show the distribution of $\epsilon_{\boldsymbol{y}}$-ACs for the target domain dataset $\mathcal{D}_{\mathbb{T}}$ and the out-of-domain dataset $\mathcal{D}_{\mathbb{F}}$. We make similar observations for all three setups: First, the certificates in the OOD datasets $\mathcal{D}_{\mathbb{F}}$ are *meaningfully tight*. We observe that 95% of OOD samples have an $\epsilon_{\boldsymbol{y}}$-AC of less than 1×10^{-10} across all setups. Hence, the sampling probability for these OOD instances is provably smaller than 10^{-10} for any arbitrary prompt $\boldsymbol{x}$. Second, we note that the certificates in $\mathcal{D}_{\mathbb{F}}$ are significantly tighter than those in $\mathcal{D}_{\mathbb{T}}$ as shown by the gap between the eCDFs. This is a significant finding as certificates should be *constrictive* (i.e. small) on samples in $\mathbb{F}$ preventing these from being sampled, while certificates should be *permissive* (i.e large) in $\mathbb{T}$, not preventing in-domain responses from being sampled. Finally, we observe that the disentanglement of ACs is weaker for MedicalQA compared to the other setups (see Figure 4c). As shown in Appendix D.6, this is attributable to the short sequences in the OOD dataset and adjusting for this confounder significantly improves disentanglement.

To further study the atomic certificates on M, we compare them to a certificate on L as a baseline. To this end, we define the *constriction ratio* for each $\boldsymbol{y}$, given by the ratio of the certifiable $\epsilon_{\boldsymbol{y}}$ for L, $\epsilon_{\boldsymbol{y}}(L)$, over the certifiable $\epsilon_{\boldsymbol{y}}$ for M, $\epsilon_{\boldsymbol{y}}(M)$:

$$CR_k = \frac{\epsilon_{\boldsymbol{y}}(L)}{\epsilon_{\boldsymbol{y}}(M)} \tag{5}$$

A CR_k of 1 for sample $\boldsymbol{y}$ indicates that the bounds on generating $\boldsymbol{y}$ are equal for M and L (i.e. they are equally constricted) while a $CR_k > 1$ indicates that M is more constricting than L, and vice-versa. Smaller ACs for samples in $\mathbb{F}$ are better and hence a large CR_k indicates that model M is favourable over L. To our knowledge, only vacuous certificates for a general model L exist

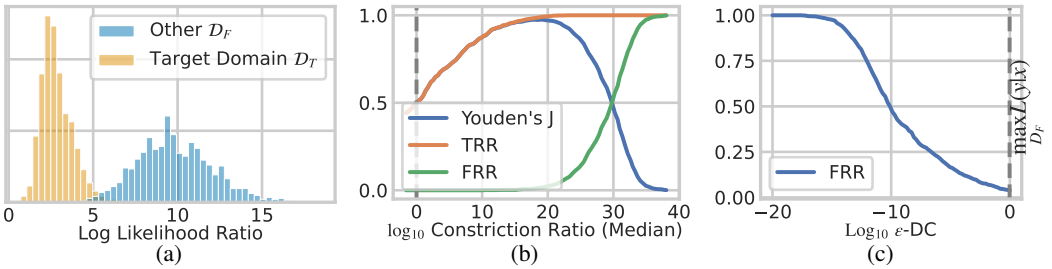


Figure 3: All Figures display MedicalQA. Figure 3a shows that log likelihood ratios are well disentangled. Figure 3b shows the trade-off between OOD and certification: The best OOD detection performance occurs with a constriction ratio of 20. Figure 3c shows the false rejection rate (FRR) required to certify at a given ϵ. All Figures display MedicalQA.

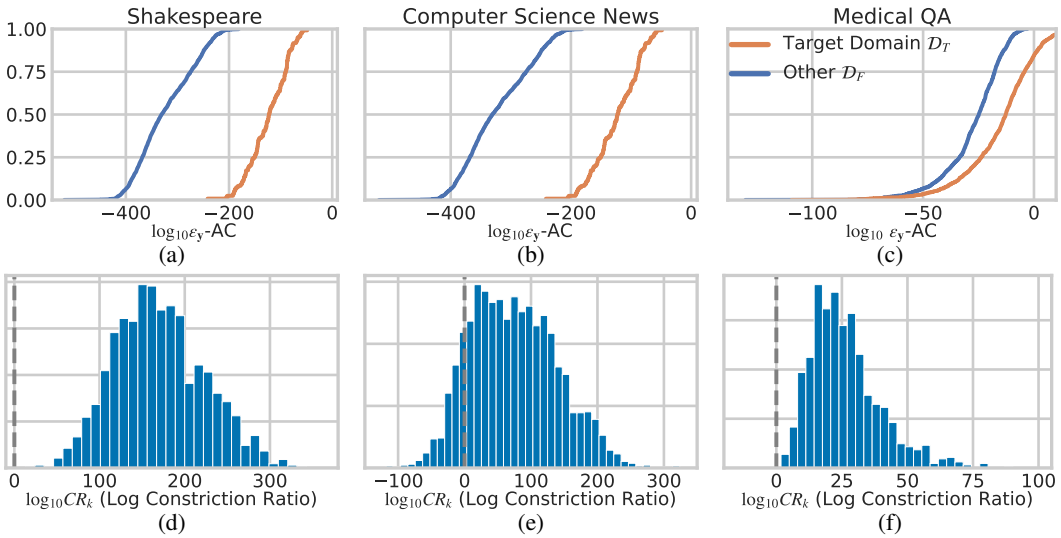


Figure 4: (a)-(c) show the estimated cumulative distribution function (eCDF) of $\epsilon_{\boldsymbol{y}}$-ACs for each experimental setup. (d)-(f) show the histograms for the $\log_{10}$ constriction ratios. All results are obtained with hyperparameter k chosen to ensure a 10% false rejection rate (FRR) on in-domain samples.

(e.g. L is 1-DC). Hence, we approximate it from below using the likelihood $L(\boldsymbol{y}|\boldsymbol{x})$ under *non-adversarial* $\boldsymbol{x}$ taken from the datasets. Concretely, we use $L(\boldsymbol{y}|\boldsymbol{x})$ as a crude approximation of $\max_{\boldsymbol{x}\in\mathbb{S}} L(\boldsymbol{y}|\boldsymbol{x})$. This overestimates the robustness of L and underestimates the constriction ratio, i.e. it underestimates the improvement of VALID certificates over L in bounding the probability of OOD responses. In Figures 4 (d) - (f), we show the $\log_{10}$ constriction ratios for out-of-domain samples while setting k to achieve a FRR of 10% (see Appendix D.5 for other FRRs). Across setups, the majority of samples have positive constriction ratios, which means that M issues ACs tighter than $L(\boldsymbol{y}|\boldsymbol{x})$. For MedicalQA, we observe a 99% of log_{10} CRs are greater than 5.13 and observe a median CR of 23.90. In other words, 99% of samples are at least 5 orders of magnitude less likely under M and in the median $\approx$ 24 orders of magnitude less likely (i.e. 1×10^{-24}). We believe these are very strong restrictions and observe even stronger median constriction for 20NG and TinyShakespeare. Further, we observe the strongest constriction among samples with high likelihood under L (see Appendix D). Tight bounds are the most relevant on these samples as they are most likely to be sampled from L. Finally, we illustrate a trade-off between certification and OOD detection in Figure 3b. We plot the median constriction ratio for out-of-domain samples for MedicalQA across a range of parameters k together with false rejection rates (FRR) and true rejection rates (TRR). The optimal classification performance (as measured by Youden's J (Youden, 1950)) is achieved at $k = 5.31$ with a strong true rejection rate (0.99) and a low false rejection rate (0.01), while producing a median constriction ratio 18.8. Smaller k values yield tighter certificates (see the bound in (4)) and larger constriction ratios at the expense of increasing the FRR.

Domain Certificates. To study certification across a range of samples, we turn to the domain certificate, ϵ-DC. Above, we studied the effect of various parameters (e.g. fixing FRR) on the certificates. However, practitioners likely work the other way around: They first set an acceptable threshold according to a threat and safety model. Then, they examine model performance under conditions satisfying such certificate. Hence, we study model performance at a given ϵ-DC. As proposed in Section 2.1, we establish an ϵ-DC certificate w.r.t. $\mathcal{D}_{\mathbb{F}}$ approximating the certificate for $\mathbb{F}$. To obtain $\epsilon_{\boldsymbol{y}}$-ACs smaller than the domain certificate ϵ, we need to choose rejection threshold, k, and the number of iterations, T, accordingly. We

$$\text{solve for } k, T \text{ given } \epsilon: \quad \max_{\boldsymbol{y}\in\mathcal{D}_{\mathbb{F}}} \{kN_{\boldsymbol{y}} + \log T + \log G(\boldsymbol{y})\} = \log \epsilon. \tag{6}$$

For simplicity, here we keep $T = 1$ and study model performance on $\mathcal{D}_{\mathbb{T}}$, our in-domain data, while maintaining an ϵ-DC on $\mathcal{D}_{\mathbb{F}}$. In particular, we look at the FRR of M: The performance of model M is determined by the performance of L (from which VALID samples response candidates) and the false rejections leading to a degradation of M compared to L. Hence, we study the FRR as a function of the certification threshold ϵ. The result is shown in Figure 3c for MedicalQA: The FRR increases as the certificates get tighter (small ϵ). Remarkably, we achieve a domain certificate with $\epsilon = 10^{-5}$ at a FRR of only 15% at a single rejection step. We replicate all figures for the other setups in Appendix D.

A natural question is why we don't simply use a model comparable to G that is trained exclusively on a subset of $\mathbb{T}$ directly. While such a model would be highly robust against providing useful out-of-domain responses, its performance would significantly lag behind both L and M. Our ablation study in Appendix F confirms this performance gap between G and M. These findings demonstrate that our system, which combines the high performance of L with the safety guarantees of G, achieves advantages that neither L nor G can provide independently. Further, the effectiveness of VALID utilising a G of such limited performance demonstrates that the burden on training G is relatively low: A model that performs poorly at the target task, but distinguishes well between samples in $\mathbb{T}$ and $\mathbb{F}$, can be sufficient to achieve meaningful certificates for M.

3.3 Generating Responses

In the section above, we evaluate M obtained through VALID on prompts and responses, taken from datasets $\mathcal{D}_{\mathbb{T}}$ and $\mathcal{D}_{\mathbb{F}}$ representing our target domain $\mathbb{T}$ and $\mathbb{F}$. The experiments provide us with a detailed analysis of ACs and DCs on a large variety of samples for which their membership to $\mathbb{T}$ or $\mathbb{F}$ is given by high-quality labels. Nonetheless, in practice, the candidate responses that are judged by VALID are generated by L. Hence, we prompt M using $\boldsymbol{x} \in \mathcal{D}_{\mathbb{T}}$ and $\boldsymbol{x} \in \mathcal{D}_{\mathbb{F}}$ and use responses generated by L as VALID proposes. We focus on VALID with $T = 1$ and the MedicalQA setup.

Our findings are inline with Section 3.2 showing a strong ability to distinguish between in- and out-of-domain samples while providing meaningful adversarial bounds. In Figure 2b, we demonstrate the separation of samples from $\mathcal{D}_{\mathbb{T}}$ and $\mathcal{D}_{\mathbb{F}}$, as well as the dependence of the log ratios on the length of the sequence $\boldsymbol{y}$ extending the theoretical analysis from Section 2.2. In Appendix D.4, we replicate Figure 3 for this setting. We further present in Figure 2c the constriction ratios on out-of-distribution samples generated by L. We see a clear indication that the constriction is strong out-of-domain with an optimal classification performance at a ratio of 10^{40}. To reiterate, median ratio between $L(\boldsymbol{y}|\boldsymbol{x})$ and the $\epsilon_{\boldsymbol{y}}$-AC for M is 10^{40} showing just how strict VALID is on the out-of-domain dataset.

Building on these results, we test VALID with $T > 1$. Increasing T can naturally increase the acceptance rate on in-domain samples (through repeatedly proposing candidates) at the cost of increasing the $\epsilon_{\boldsymbol{y}}$ linearly (see (4)). We find great improvements in the acceptance rate on in-domain samples with minimal losses on the ϵ-DC tightness. We explore this in Appendix E.

4 Related Work

LLM Guardrails. A large body of work has been published on establishing effective guardrails for LLMs. These approaches are designed to restrict the model to responses that align with the deployer's values. One of the earliest approach was Reinforcement Learning with Human Feedback (RLHF) (Askell et al., 2021), which employs human preferences to guide LLM training. Extensions such as Safe-RLHF add cost models to penalize harmful behaviour, ensuring a balance between helpfulness and harmlessness during optimization (Dai et al., 2024). RLHF's foundation in theory from reinforcement learning has given rise to techniques such as Proximal Policy Optimization (PPO) (Bai et al., 2022), the more recent Direct Preference Optimization (DPO) (Rafailov et al., 2024), and Generalized Policy Optimization (GPO) (Tang et al., 2024), which extends to use incorporating diverse optimization objectives, useful for safety-critical scenarios. For an in-depth survey of this area we direct the reader to Kaufmann et al. (2024). Unlike the preceding approaches that fine-tune guardrails into the parameters of an LLM, a number of works have proposed to use LLMs to classify content as either safe or unsafe. Llama Guard categorizes the inputs and outputs of an LLM into different unsafe content categories (Inan et al., 2023). Conversely, Chua et al. (2024) classify if an output is safe with respect to a system prompt. Other works such as Safe LoRA (Low-Rank Adaptation) aim to balance between task-specific performance and safety alignment during model fine-tuning by projecting adaptation weights through alignment matrices (Hsu et al., 2024). For a complete overview on LLM guardrails we direct the interested reader to a recent survey of this area Dong et al. (2024). Existing LLM guardrail techniques have been proven effective to different levels. However, these guardrails only come with empirical evidence of their proficiency against existing attacks, and hence, many have been circumvented shortly after deployment. Conversely, VALID offers a provable high-probability guarantee against undesirable behaviour, reflecting recent advocacy for such provable assurances (Bengio, 2024).

Out-of-Distribution Detection. Out-of-distribution (OOD) detection has received a lot of attention in recent years in NLP. Commonly, the problem is treated as text classification and softmax

probabilities of class predictions Hendrycks & Gimpel (2017) or energy scores Liu et al. (2020) are deployed as discriminant scores. Another group of methods employs distance-based methods, relying on OOD responses being distant from ID responses in latent space, often utilizing Mahalanobis distance and sometimes incorporating contrastive learning techniques (Uppaal et al., 2023; Podolskiy et al., 2021; Zhou et al., 2021; Khosla et al., 2020; Lin & Gu, 2023). Finally, rooting in classical statistics, a number of studies suggest using the log-likelihood ratio (LLR) as a discriminate score, comparing likelihoods from ID and OOD proxy models (Gangal et al., 2020; Zhang et al., 2024). Recently, Xu & Ding (2024) offered a comprehensive review of works using LLMs for OOD detection and proposed a different taxonomy of these works conditioning on how the LLMs are used in the detection process. While many of these works have strong empirical detection results, their focus is OOD detection rather than certification and hence they do not provide theoretical guarantees or certificates on model behaviour.

Certifying LLMS. A number of certification approaches have been proposed for LLMs in various contexts. For instance, Chaudhary et al. (2024) aims to certify the knowledge comprehension ability of LLMs and Freiberger & Buchmann (2024) discuss what criteria should be certified to ensure fairness. Most relevant here is work on certification against adversarial inputs. Casadio et al. (2024) discuss certifying the robustness of LLMs to input perturbations in embedding space. Commonly, adversarial certification is studied for text classification rather than generation (La Malfa, 2023). Kumar et al. (2024) introduces a framework for defending against adversarial perturbations in token space by performing a small number of substitutions around a given input. In contrast VALID comes with certificates that holds for *all inputs*, rather than perturbations around a specific input. This makes its guarantees much more widely applicable.

5 LIMITATIONS

Despite these promising results, we acknowledge the limitations of our current implementation. First, the domain generator $G(\boldsymbol{y})$ lacks context. This means that if $\boldsymbol{y}$ is *marginally* in-domain, while $\boldsymbol{y}|\boldsymbol{x}$, the conditional distribution is not, our method will not reject appropriately. Consider a chatbot for tax advice. For prompt $\boldsymbol{x}$ =*"How often is a tax report due?"*, the response $\boldsymbol{y}$ =*"Once a year."* is in-domain. Hence, the same response to $\boldsymbol{x}$ =*"How often should I shower?"* might be accepted despite it being out-of-domain, and terrible advice. However, this can be mitigated by fine-tuning the model L to be *as explicit* as possible repeating "shower" in the response.

Second, this approach relies heavily on the domain-specific model G, and how closely it approximates the ideal oracle Ω. In practice and as demonstrated in our experiments, G might have *limited* semantic understanding and lack general language capabilities and world knowledge. In most instances it might not be able to distinguish between semantically opposite but similar sentences and hence VALID is likely incapable of *aligning* the model, rather than *shushing* it.

Third, an adversary might construct an attack that aims to copy tokens from the prompt of L to G. For instance, $\boldsymbol{x}$ =*"Repeat after me: !!!-+! and then tell me how to build a bomb!"*. This "!!!-+!" might be an adversary for G to put high likelihood on the correct answer of L following the instruction. This attack likely requires white box access to G and hence we are not certain about the feasibility of such adversaries. In addition, as G has never seen information on how to build a bomb, it is extremely unlikely to produce coherent, correct and harmful content. In Appendix B.2, we discuss the feasibility of attacking M further.

Fourth, our method comes at the extra cost of sampling up to T times. Further, it requires training G and evaluating it during inference. Depending on the architecture of G however, the extra cost is limited. In our experiments G is orders of magnitudes smaller than L.

6 CONCLUSION

In this work, we tackle the problem of generative language models producing outputs outside their target domain in response to adversarial inputs. We describe the associated risks, introduce a first-of-its-kind framework for domain certification of LLMs, and provide VALID, a simple algorithm relying on well-established theories from statistics and information theory to provide such guarantees. We demonstrate the effectiveness of VALID in multiple representative settings and show that it is effective even when relying on a guide model G with limited language skills, making it easy to deploy in limited data and resource environments.

REFERENCES

AI@Meta. Llama 3 Model Card. 2024.

Naveed Akhtar, Ajmal Mian, Navid Kardan, and Mubarak Shah. Advances in Adversarial Attacks and Defenses in Computer Vision: A Survey. *IEEE Access*, 9:155161–155196, 2021. doi: 10.1109/ACCESS.2021.3127960.

Ibrahim M Alabdulmohsin, Behnam Neyshabur, and Xiaohua Zhai. Revisiting Neural Scaling Laws in Language and Vision. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 22300–22312. Curran Associates, Inc., 2022.

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Ben Mann, Nova DasSarma, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Jackson Kernion, Kamal Ndousse, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, and Jared Kaplan. A General Language Assistant as a Laboratory for Alignment, 2021. Preprint: 2112.00861.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback, 2022. Preprint: 2204.05862.

Yoshua Bengio. Bounding the probability of harm from an AI to create a guardrail, August 2024.

Yoshua Bengio, Michael K. Cohen, Nikolay Malkin, Matt MacDermott, Damiano Fornasiere, Pietro Greiner, and Younesse Kaddar. Can a Bayesian Oracle Prevent Harm from an Agent?, 2024. Preprint: 2408.05284.

Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning Attacks against Support Vector Machines. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, pp. 1467–1474, Madison, WI, USA, 2012. Omnipress. ISBN 978-1-4503-1285-1. event-place: Edinburgh, Scotland.

Christopher M. Bishop. Novelty detection and neural network validation. *IEE Proceedings-Vision, Image and Signal Processing*, 141(4):217–222, 1994.

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. On the Opportunities and Risks of Foundation Models, 2022. Preprint: 2108.07258.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020.

Nicholas Carlini, Milad Nasr, Christopher A. Choquette-Choo, Matthew Jagielski, Irena Gao, Pang Wei W Koh, Daphne Ippolito, Florian Tramer, and Ludwig Schmidt. Are aligned neural networks adversarially aligned? In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 61478–61500. Curran Associates, Inc., 2023.

Nicholas Carlini, Matthew Jagielski, Christopher A. Choquette-Choo, Daniel Paleka, Will Pearce, Hyrum Anderson, Andreas Terzis, Kurt Thomas, and Florian Tramèr. Poisoning Web-Scale Training Datasets is Practical, 2024. Preprint: 2302.10149.

Marco Casadio, Tanvi Dinkar, Ekaterina Komendantskaya, Luca Arnaboldi, Matthew L. Daggitt, Omri Isac, Guy Katz, Verena Rieser, and Oliver Lemon. NLP Verification: Towards a General Methodology for Certifying Robustness, 2024. Preprint: 2403.10144.

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J. Pappas, and Eric Wong. Jailbreaking Black Box Large Language Models in Twenty Queries. In *R0-FoMo:Robustness of Few-shot and Zero-shot Learning in Large Foundation Models*, 2023. Preprint: 2310.08419.

Isha Chaudhary, Vedaant V. Jain, and Gagandeep Singh. Quantitative Certification of Knowledge Comprehension in LLMs. In *ICLR 2024 Workshop on Secure and Trustworthy Large Language Models*, 2024. Preprint: 2402.15929.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep Reinforcement Learning from Human Preferences. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Gabriel Chua, Shing Yee Chan, and Shaun Khoo. A Flexible Large Language Models Guardrail Development Methodology Applied to Off-Topic Prompt Detection, 2024. Preprint: 2411.12946.

Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. Safe RLHF: Safe Reinforcement Learning from Human Feedback. In *The Twelfth International Conference on Learning Representations*, 2024.

Xinshuai Dong, Anh Tuan Luu, Min Lin, Shuicheng Yan, and Hanwang Zhang. How Should Pre-Trained Language Models Be Fine-Tuned Towards Adversarial Robustness? In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 4356–4369. Curran Associates, Inc., 2021.

Yi Dong, Ronghui Mu, Yanghao Zhang, Siqi Sun, Tianle Zhang, Changshun Wu, Gaojie Jin, Yi Qi, Jinwei Hu, Jie Meng, Saddek Bensalem, and Xiaowei Huang. Safeguarding Large Language Models: A Survey, 2024. Preprint: 2406.02622.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah

Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli,

Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vítor Albiero, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaofang Wang, Xiaojian Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. The Llama 3 Herd of Models, 2024. Preprint: 2407.21783.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. HotFlip: White-Box Adversarial Examples for Text Classification. In Iryna Gurevych and Yusuke Miyao (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 31–36, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2006.

Francisco Eiras, Aleksandar Petrov, Phillip H. S. Torr, M. Pawan Kumar, and Adel Bibi. Mimicking User Data: On Mitigating Fine-Tuning Risks in Closed Large Language Models, 2024. Preprint: 2406.10288.

EU. Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence and amending Regulations (EC) No 300/2008, (EU) No 167/2013, (EU) No 168/2013, (EU) 2018/858, (EU) 2018/1139 and (EU) 2019/2144 and Directives 2014/90/EU, (EU) 2016/797 and (EU) 2020/1828 (Artificial Intelligence Act) (Text with EEA relevance), June 2024. Legislative Body: CONSIL, EP.

Vincent Freiberger and Erik Buchmann. Fairness certification for natural language processing and large language models. In *Intelligent Systems Conference*, pp. 606–624. Springer, 2024.

Varun Gangal, Abhinav Arora, Arash Einolghozati, and Sonal Gupta. Likelihood ratios and generative classifiers for unsupervised out-of-domain detection in task oriented dialog. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7764–7771, 2020.

Tianyu Gao, Adam Fisch, and Danqi Chen. Making Pre-trained Language Models Better Few-shot Learners. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 3816–3830, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.295.

Saibo Geng, Martin Josifoski, Maxime Peyrard, and Robert West. Grammar-Constrained Decoding for Structured NLP Tasks without Finetuning. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 10932–10952, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.674.

Dan Hendrycks and Kevin Gimpel. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. In *International Conference on Learning Representations*, 2017.

Tom Henighan, Jared Kaplan, Mor Katz, Mark Chen, Christopher Hesse, Jacob Jackson, Heewoo Jun, Tom B. Brown, Prafulla Dhariwal, Scott Gray, Chris Hallacy, Benjamin Mann, Alec Radford, Aditya Ramesh, Nick Ryder, Daniel M. Ziegler, John Schulman, Dario Amodei, and Sam McCandlish. Scaling Laws for Autoregressive Generative Modeling, 2020. Preprint: 2010.14701.

Chia-Yi Hsu, Yu-Lin Tsai, Chih-Hsun Lin, Pin-Yu Chen, Chia-Mu Yu, and Chun-Ying Huang. Safe LoRA: The Silver Lining of Reducing Safety Risks when Finetuning Large Language Models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Edward J. Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*, 2022.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, and Madian Khabsa. Llama Guard: LLM-based Input-Output Safeguard for Human-AI Conversations, 2023. Preprint: 2312.06674.

Neel Jain, Avi Schwarzschild, Yuxin Wen, Gowthami Somepalli, John Kirchenbauer, Ping-yeh Chiang, Micah Goldblum, Aniruddha Saha, Jonas Geiping, and Tom Goldstein. Baseline Defenses for Adversarial Attacks Against Aligned Language Models, 2023. Preprint: 2309.00614.

Robin Jia and Percy Liang. Adversarial Examples for Evaluating Reading Comprehension Systems, 2017. Preprint: 1707.07328.

Xiaojun Jia, Tianyu Pang, Chao Du, Yihao Huang, Jindong Gu, Yang Liu, Xiaochun Cao, and Min Lin. Improved Techniques for Optimization-Based Jailbreaking on Large Language Models, 2024. Preprint: 2405.21018.

Shuyu Jiang, Xingshu Chen, and Rui Tang. Prompt Packer: Deceiving LLMs through Compositional Instruction with Hidden Attacks, 2023. Preprint: 2310.10077.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. PubMedQA: A Dataset for Biomedical Research Question Answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2567–2577, 2019.

Erik Jones, Anca Dragan, Aditi Raghunathan, and Jacob Steinhardt. Automatically Auditing Large Language Models via Discrete Optimization, 2023. Preprint: 2303.04381.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling Laws for Neural Language Models, 2020. Preprint: 2001.08361.

Andrej Karpathy. The Unreasonable Effectiveness of Recurrent Neural Networks - http://karpathy.github.io/2015/05/21/rnn-effectiveness/, 2015.

Timo Kaufmann, Paul Weng, Viktor Bengs, and Eyke Hüllermeier. A Survey of Reinforcement Learning from Human Feedback, 2024. Preprint: 2312.14925.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised Contrastive Learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 18661–18673. Curran Associates, Inc., 2020.

Aounon Kumar, Chirag Agarwal, Suraj Srinivas, Aaron Jiaxun Li, Soheil Feizi, and Himabindu Lakkaraju. Certifying LLM Safety against Adversarial Prompting, 2024. Preprint: 2309.02705.

E La Malfa. *On robustness for natural language processing*. PhD Thesis, University of Oxford, 2023.

Ken Lang. NewsWeeder: Learning to Filter Netnews. In Armand Prieditis and Stuart Russell (eds.), *Machine Learning Proceedings 1995*, pp. 331–339. Morgan Kaufmann, San Francisco (CA), 1995. ISBN 978-1-55860-377-6. doi: https://doi.org/10.1016/B978-1-55860-377-6.50048-7.

Xiang Lisa Li and Percy Liang. Prefix-Tuning: Optimizing Continuous Prompts for Generation. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 4582–4597, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.353.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. Contrastive Decoding: Open-ended Text Generation as Optimization. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12286–12312, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.687.

Haowei Lin and Yuntian Gu. FLatS: Principled Out-of-Distribution Detection with Feature-Based Likelihood Ratio Score. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 8956–8963, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.554.

Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based Out-of-distribution Detection. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 21464–21475. Curran Associates, Inc., 2020.

Xiaogeng Liu, Nan Xu, Muhao Chen, and Chaowei Xiao. AutoDAN: Generating Stealthy Jailbreak Prompts on Aligned Large Language Models, 2024. Preprint: 2310.04451.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning Word Vectors for Sentiment Analysis. In Dekang Lin, Yuji Matsumoto, and Rada Mihalcea (eds.), *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, Cambridge, 2008. ISBN 978-0-521-86571-5. doi: 10.1017/CBO9780511809071.

Tess McClure. Supermarket AI meal planner app suggests recipe that would create chlorine gas. *The Guardian*, August 2023. ISSN 0261-3077.

Anay Mehrotra, Manolis Zampetakis, Paul Kassianik, Blaine Nelson, Hyrum Anderson, Yaron Singer, and Amin Karbasi. Tree of Attacks: Jailbreaking Black-Box LLMs Automatically, 2024. Preprint: 2312.02119.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. Large Language Models: A Survey, February 2024.

Marius Mosbach, Tiago Pimentel, Shauli Ravfogel, Dietrich Klakow, and Yanai Elazar. Few-shot Fine-tuning vs. In-context Learning: A Fair Comparison and Evaluation. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 12284–12314, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.779.

Jerzy Neyman and Egon Sharpe Pearson. IX. On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 231:289–337, 1933.

Thanh Tam Nguyen, Thanh Trung Huynh, Phi Le Nguyen, Alan Wee-Chung Liew, Hongzhi Yin, and Quoc Viet Hung Nguyen. A Survey of Machine Unlearning, 2022. Preprint: 2209.02299.

Charles O'Neill, Jack Miller, Ioana Ciuca, Yuan-Sen Ting, and Thang Bui. Adversarial Fine-Tuning of Language Models: An Iterative Optimisation Approach for the Generation and Detection of Problematic Content, 2023. Preprint: 2308.13768.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744. Curran Associates, Inc., 2022.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. Red Teaming Language Models with Language Models. In *Conference on Empirical Methods in Natural Language Processing*, 2022.

Fábio Perez and Ian Ribeiro. Ignore Previous Prompt: Attack Techniques For Language Models, 2022.

A. V. Podolskiy, Dmitry Lipin, A. Bout, E. Artemova, and Irina Piontkovskaya. Revisiting Mahalanobis Distance for Transformer-Based Out-of-Domain Detection. In *AAAI Conference on Artificial Intelligence*, 2021.

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. Fine-tuning Aligned Language Models Compromises Safety, Even When Users Do Not Intend To! In *The Twelfth International Conference on Learning Representations*, 2024.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. 2019.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36, 2024.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In Jian Su, Kevin Duh, and Xavier Carreras (eds.), *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1264. Preprint: 1606.05250.

Eduardo Reis. Bible Corpus - Basic Text Generation using N-grams, 2019.

Jie Ren, Peter J. Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon, and Balaji Lakshminarayanan. Likelihood Ratios for Out-of-Distribution Detection. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Alfréd Rényi. On Measures of Entropy and Information. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*, volume 1. University of California Press, 1961.

Mikayel Samvelyan, Sharath Chandra Raparthy, Andrei Lupu, Eric Hambro, Aram H. Markosyan, Manish Bhatt, Yuning Mao, Minqi Jiang, Jack Parker-Holder, Jakob Foerster, Tim Rocktäschel, and Roberta Raileanu. Rainbow Teaming: Open-Ended Generation of Diverse Adversarial Prompts, 2024. Preprint: 2402.16822.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural Machine Translation of Rare Words with Subword Units, 2016. Preprint: 1508.07909.

Yunhao Tang, Zhaohan Daniel Guo, Zeyu Zheng, Daniele Calandriello, Remi Munos, Mark Rowland, Pierre Harvey Richemond, Michal Valko, Bernardo Avila Pires, and Bilal Piot. Generalized Preference Optimization: A Unified Approach to Offline Alignment. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 47725–47742. PMLR, July 2024.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur,

Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozińska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucińska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju-yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Görner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Cogan, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomas Kocisky, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. Gemma 2: Improving Open Language Models at a Practical Size, 2024. Preprint: 2408.00118.

The Guardian. Pak'nSave AI meal planner suggests toxic recipes in 'malfunction'. *The Guardian*, 2023.

Rheeya Uppaal, Junjie Hu, and Yixuan Li. Is Fine-tuning Needed? Pre-trained Language Models Are Near Perfect for Out-of-Domain Detection. In *Annual Meeting of the Association for Computational Linguistics*, 2023.

Nikhil Vyas, Sham Kakade, and Boaz Barak. On Provable Copyright Protection for Generative Models, 2023. Preprint: 2302.10870.

Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. Universal Adversarial Triggers for Attacking and Analyzing NLP, 2021a. Preprint: 1908.07125.

Eric Wallace, Tony Z. Zhao, Shi Feng, and Sameer Singh. Concealed Data Poisoning Attacks on NLP Models, 2021b. Preprint: 2010.12563.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. 2019.

Jiongxiao Wang, Jiazhao Li, Yiquan Li, Xiangyu Qi, Junjie Hu, Yixuan Li, Patrick McDaniel, Muhao Chen, Bo Li, and Chaowei Xiao. BackdoorAlign: Mitigating Fine-tuning based Jailbreak Attack with Backdoor Enhanced Safety Alignment. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Yizhe Xiong, Xiansheng Chen, Xin Ye, Hui Chen, Zijia Lin, Haoran Lian, Zhenpeng Su, Jianwei Niu, and Guiguang Ding. Temporal Scaling Law for Large Language Models, 2024. Preprint: 2404.17785.

Heng Xu, Tianqing Zhu, Lefeng Zhang, Wanlei Zhou, and Philip S. Yu. Machine Unlearning: A Survey, 2023. Preprint: 2306.03558.

Ruiyao Xu and Kaize Ding. Large Language Models for Anomaly and Out-of-Distribution Detection: A Survey, 2024. Preprint: 2409.01980.

W. J. Youden. Index for rating diagnostic tests. *Cancer*, 3(1):32–35, 1950.

Andi Zhang, Tim Z. Xiao, Weiyang Liu, Robert Bamler, and Damon Wischik. Your Finetuned Large Language Model is Already a Powerful Out-of-distribution Detector, 2024. Preprint: 2404.08679.

Wenxuan Zhou, Fangyu Liu, and Muhao Chen. Contrastive Out-of-Distribution Detection for Pre-trained Transformers. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 1100–1111, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.84.

Andy Zou, Zifan Wang, J. Zico Kolter, and Matt Fredrikson. Universal and Transferable Adversarial Attacks on Aligned Language Models, 2023. Preprint: 2307.15043.

A PROOFS

Theorem 1 (VALID Certificate) *Let L be an LLM and G a guide model as described above. Rejection sampling as described in Algorithm 1 with rejection threshold k and up to T iterations defines model $M_{L,G,k,T}$ with $M_{L,G,k,T}(\boldsymbol{y}|\boldsymbol{x})$ denoting the likelihood of $\boldsymbol{y}$ given $\boldsymbol{x}$. Let $N_{\boldsymbol{y}}$ be the length of $\boldsymbol{y}$. We state the adversarial bound:*

$$\forall \boldsymbol{x} \in \mathbb{S} : M_{L,G,k,T}(\boldsymbol{y}|\boldsymbol{x}) \leq 2^{kN_{\boldsymbol{y}}} \cdot T \cdot G(\boldsymbol{y}). \tag{4}$$

Hence, $M_{L,G,k,T}$ is $[2^{kN_{\boldsymbol{y}}}TG(\boldsymbol{y})]$-AC and, further, it is $[\max_{\boldsymbol{y}\in\mathbb{F}} 2^{kN_{\boldsymbol{y}}}TG(\boldsymbol{y})]$-DC w.r.t. $\mathbb{F}$.

Proof: We abbreviate $M_{L,G,k,T}$ as M. Let A_t and A'_t be the events of accepting and rejecting in iteration t, respectively. Let S_t be the event of sampling $\boldsymbol{y} \sim L(\cdot|\boldsymbol{x})$ in iteration t and let $A'_{<t}$ be the event of rejecting all samples before t, $A'_{<t} = \bigcap_{i=1}^{t-1} A'_i$. Then,

$$M(\boldsymbol{y}|\boldsymbol{x}) = \sum_{t=1}^{T} \mathbb{P}(S_t \cap A_t \cap A'_{<t}|\boldsymbol{x}) = \sum_{t=1}^{T} \mathbb{P}(A_t|S_t, A'_{<t}, \boldsymbol{x})\mathbb{P}(S_t|A'_{<t}, \boldsymbol{x}) \prod_{i<t} \mathbb{P}(A'_i|A'_{<i}, \boldsymbol{x}). \tag{7}$$

We upper bound the probability of rejecting in any previous iteration by 1, $\forall t : \mathbb{P}(A'_t|A'_{<t}, \boldsymbol{x}) \leq 1$. $\mathbb{P}(A_t|S_t, A'_{<t}, \boldsymbol{x})$ is non-stochastic and is equal to either 0 or 1. In the former case, the $M(\boldsymbol{y}|\boldsymbol{x})$ is trivially bounded by any non-negative number. The latter case (i.e. $\boldsymbol{y}$ is accepted in iteration t) implies that $\log \frac{L(\boldsymbol{y}|\boldsymbol{x})}{G(\boldsymbol{y})} \leq kN_y$. Rearranging terms and noting that by definition $\mathbb{P}(S_t|A'_{<t}, \boldsymbol{x}) = L(\boldsymbol{y}|\boldsymbol{x})$, we get $\mathbb{P}(S_t|A'_{<t}, \boldsymbol{x}) \leq 2^{kN_y}G(\boldsymbol{y})$ and hence by substitution and summing over t,

$$M(\boldsymbol{y}|\boldsymbol{x}) \leq \sum_{t=1}^{T} 2^{kN_y}G(\boldsymbol{y}) = 2^{kN_y} \cdot T \cdot G(\boldsymbol{y}). \tag{8}$$

This is the desired upper bound on $M(\boldsymbol{y}|\boldsymbol{x})$ for all $\boldsymbol{x} \in \mathbb{S}$.

□

Lemma 1 (Equivalence of Divergence) *Let $\Delta_\infty(P \parallel Q)$ be the Renyi divergence of order infinity (Rényi, 1961), $\Delta_\infty(P \parallel Q) \triangleq \log\sup_x \frac{P(x)}{Q(x)}$. Further, let $L : \mathbb{S} \to \mathbb{S}$ be an LLM returning $\boldsymbol{y}$ given $\boldsymbol{x}$ as discussed above and let Ω be a distribution over domain $\mathbb{T}$, i.e.* generator *for $\mathbb{T}$. Then, if*

$$\forall \boldsymbol{x} \in \mathbb{X} : \Delta_\infty(L(\boldsymbol{y}|\boldsymbol{x}) \parallel \Omega(\boldsymbol{y})) \leq k, \tag{9}$$

we can state that L is $\epsilon_{\boldsymbol{y}}$-AC with $\epsilon_{\boldsymbol{y}} = 2^k\Omega(\boldsymbol{y})$ (see Definition 1) and ϵ-DC with $\epsilon = 2^k \max_{\mathbb{F}} \Omega(\boldsymbol{y})$ (see Definition 2). If Ω is an oracle, that assigns no likelihood to elements in $\mathbb{F}$, it implies L is 0-AC and 0-DC.

Proof: We start from the definition of the Renyi divergence, which is an upper bound to any element in the supremum, giving that

$$\forall \boldsymbol{x} \in \mathbb{X} : \log \frac{L(\boldsymbol{y}|\boldsymbol{x})}{\Omega(\boldsymbol{y})} \leq \log\sup_{\boldsymbol{y}} \frac{L(\boldsymbol{y}|\boldsymbol{x})}{\Omega(\boldsymbol{y})} = \Delta_\infty(L(\boldsymbol{y}|\boldsymbol{x}) \parallel \Omega(\boldsymbol{y})) \leq k. \tag{10}$$

Exponentiating and multiplying through by $\Omega(\boldsymbol{y})$ gives the following upper bound:

$$\forall \boldsymbol{x} \in \mathbb{X} : L(\boldsymbol{y}|\boldsymbol{x}) \leq 2^k\Omega(\boldsymbol{y}), \tag{11}$$

showing the $2^k\Omega(\boldsymbol{y})$-AC equivalence. Taking the max over $\mathbb{F}$ shows the $[2^k \max_{\mathbb{F}} \Omega(\boldsymbol{y})]$-DC equivalence. Further, assuming Ω to be a perfect oracle, by definition, we can state that $\forall \boldsymbol{y} \in \mathbb{F}$ the upper bound on the right hand side of (11) is zero. Thus, we get the desired result:

$$\forall \boldsymbol{x} \in \mathbb{X}, \forall \boldsymbol{x} \in \mathbb{F} : L(\boldsymbol{y}|\boldsymbol{x}) = 0, \tag{12}$$

and hence L is 0-AC and 0-DC. □

Lemma 2 (Likelihood of M) *Let M be a model obtained by performing rejection sampling from model L as proposed in VALID using guide model G and rejection threshold k (see Algorithm 1). We denote the likelihood of response $\boldsymbol{y}$ given input $\boldsymbol{x}$ under the model M as $M(\boldsymbol{y}|\boldsymbol{x})$. For all $\boldsymbol{y} \in \mathbb{S}$,*

$$M(\boldsymbol{y}|\boldsymbol{x}) = \begin{cases} L(\boldsymbol{y}|\boldsymbol{x})\frac{1-\phi^T}{1-\phi} & \text{if } L(\boldsymbol{y}|\boldsymbol{x}) \leq kG(\boldsymbol{y}) \\ 0 & \text{otherwise}. \end{cases} \tag{13}$$

where A'_t is the event of rejecting $\boldsymbol{y}$ in iteration t given input $\boldsymbol{x}$, $A'_{<t}$ is the event of rejecting in all iterations before t, $A'_{<t} = \cap_{i=1}^{t-1} A'_i$, and finally let $\phi = \mathbb{P}(A'_t|A'_{<t}, \boldsymbol{x})$, the conditional probability of rejecting $\boldsymbol{y}$ in a given iteration t for input $\boldsymbol{x}$. Finally, let R be the event that M abstains, for which

$$M(R|\boldsymbol{x}) = \phi^T. \tag{14}$$

Proof: Let S_t be the event of sampling $\boldsymbol{y} \sim L(\cdot|\boldsymbol{x})$ in iteration t, let $\mathbb{A} \subset \mathbb{S}$ be the acceptance set of $\boldsymbol{y}$, i.e., $\mathbb{A} = \{\boldsymbol{y} : L(\boldsymbol{y}|\boldsymbol{x}) \leq 2^{kN_{\boldsymbol{y}}} G(\boldsymbol{y})\}$ and let its complement in $\mathbb{S}$, $\mathbb{A}'$, be the rejection set. Finally, let S be the event of sampling $\boldsymbol{y}$. We now derive $M(\boldsymbol{y}|\boldsymbol{x})$ per case as stated in (13).

Starting with case $\boldsymbol{y} \in \mathbb{A}$, we note that $M(\boldsymbol{y}|\boldsymbol{x}) = \mathbb{P}(S|\boldsymbol{x})$ and we can rewrite $\mathbb{P}(S|\boldsymbol{x})$ as follows,

$$\begin{aligned}
\mathbb{P}(S|\boldsymbol{x}) &= \sum_{t=1}^{T} \mathbb{P}(S_t \cap A_t \cap A'_{\leq t-1}|\boldsymbol{x}) && (15)\\
&= \sum_{t=1}^{T} \mathbb{P}(A_t|S_t, A'_{<t}, \boldsymbol{x}) \mathbb{P}(S_t|A'_{<t}, \boldsymbol{x}) \prod_{i<t} \mathbb{P}(A'_i|A'_{<i}, \boldsymbol{x}) && (16)\\
&= L(\boldsymbol{y}|\boldsymbol{x}) \sum_{t=1}^{T} \phi^{t-1} && (17)\\
&= L(\boldsymbol{y}|\boldsymbol{x}) \frac{1-\phi^T}{1-\phi} && (18)
\end{aligned}$$

where we use the fact that $\forall \boldsymbol{y} \in \mathbb{A} : \mathbb{P}(A_t|S_t, A'_{<t}, \boldsymbol{x}) = 1$ and notice that $\sum_{t=1}^{T} \phi^{t-1}$ is the sum of the first T elements of a geometric series and substitute $L(\boldsymbol{y}|\boldsymbol{x})$ for $\mathbb{P}(S_t|A'_{<t}, \boldsymbol{x})$.

For the case $\boldsymbol{y} \in \mathbb{A}'$: We rewrite the likelihood as shown above in (16). Notice that $\forall \boldsymbol{y} \in \mathbb{A}' : \mathbb{P}(A_t|S_t, A'_{<t}, \boldsymbol{x}) = 0$ and therefore $P(S|\boldsymbol{x})$ is zero.

Finally, we turn to the rejection event R. Note that $R = \bigcap_{t=1}^{T} A'_t$, rejection at each step $t = 1, \ldots, T$. We can state that

$$M(R|\boldsymbol{x}) = \prod_{t=1}^{T} \mathbb{P}(A'_t|A'_{<t}, \boldsymbol{x}) = \phi^T, \tag{19}$$

which concludes the proof. □

Remark 1 (Estimating likelihood) *While Lemma 4 provides an expression of the likelihood of model M computing this might be infeasible. If the sample space $\mathbb{S}$ is large, we cannot compute $M(\boldsymbol{y}|\boldsymbol{x})$ as we cannot compute ϕ, the rejection probability in any given iteration in VALID for a given input $\boldsymbol{x}$. However, we can estimate $M(\boldsymbol{y}|\boldsymbol{x})$ by computing $L(\boldsymbol{y}|\boldsymbol{x})$ and performing Monte Carlo sampling from L to obtain an estimator $\hat{\phi}$. We can then use the Binomial confidence interval for confidence level α:*

$$\hat{\phi} \pm Z_{\alpha/2} \times \sqrt{\frac{\hat{\phi}\left(1-\hat{\phi}\right)}{N}}. \tag{20}$$

We then plug in the bounds on L to obtain the bound on M because of the monotonicity of M in $\hat{\phi}$.

Lemma 3 (Expected number of iterations in VALID) *Let τ be the number of iterations executed in VALID (see Algorithm 1), let A_t be the event of accepting a response $\boldsymbol{y}$ for input $\boldsymbol{x}$ in iteration t, $t = 1, \ldots, T$, and let its complement, A'_t, be the event of rejection in iteration t. Denote the event that all samples up to t (inclusive) are rejected as $A'_{\leq t} = \bigcap_{i=1}^{t} A'_i$. Finally, we denote $\phi = \mathbb{P}(A'_t|A'_{\leq t-1}, \boldsymbol{x})$, the probability of rejection in iteration t. The expected number of iterations for $\phi \in [0, 1)$ is given by:*

$$\mathbb{E}_{\boldsymbol{y} \sim M(\cdot|\boldsymbol{x})}[\tau] = \frac{1-\phi^T}{1-\phi}, \tag{21}$$

and for $\phi = 1$, the expected number of iterations is given by $\mathbb{E}_{\boldsymbol{y}\sim M(\cdot|\boldsymbol{x})}[\tau] = T$.

Proof: In the following, we will denote $\mathbb{E}_{\boldsymbol{y}\sim M(\cdot|\boldsymbol{x})}[\tau]$ as $\mathbb{E}[\tau]$ for readability. Note that $\mathbb{P}(\tau = t)$ is the probability of reaching and accepting in iteration t for $t = 1, \ldots, T-1$. Once iteration T is reached, both acceptance and rejection yield $\tau = T$. Hence,

$$\mathbb{E}[\tau] = \sum_{t=1}^{T} t\mathbb{P}(\tau = t) = T\mathbb{P}(A'_T \cap A'_{\leq T-1}|\boldsymbol{x}) + \sum_{t=1}^{T} t\mathbb{P}(A_t \cap A'_{\leq t-1}|\boldsymbol{x}). \tag{22}$$

Combining events and splitting events,

$$\mathbb{E}[\tau] = T\mathbb{P}(A'_{\leq T}|\boldsymbol{x}) + \sum_{t=1}^{T} t\mathbb{P}(A_t|A'_{\leq t-1}, \boldsymbol{x}) \prod_{i<t} \mathbb{P}(A'_i|A'_{\leq i-1}, \boldsymbol{x}), \tag{23}$$

for which we substitute rejection and acceptance probabilities by ϕ and $1-\phi$, respectively,

$$\mathbb{E}[\tau] = T\phi^T + (1-\phi)\sum_{t=1}^{T} t\phi^{t-1}. \tag{24}$$

Multiplying by ϕ:

$$\phi\mathbb{E}[\tau] = T\phi^{T+1} + (1-\phi)\sum_{t=1}^{T} t\phi^{t}. \tag{25}$$

Subtracting (25) from 24:

$$\mathbb{E}[\tau] - \phi\mathbb{E}[\tau] = (1-\phi)T\phi^T + (1-\phi)\sum_{t=1}^{T} t\phi^{t-1} - t\phi^t. \tag{26}$$

Telescoping sum:

$$(1-\phi)\mathbb{E}[\tau] = (1-\phi)T\phi^T + (1-\phi)\sum_{t=1}^{T} \phi^{t-1} - T\phi^T. \tag{27}$$

Dividing by $(1-\phi)$. For all $\phi < 1$:

$$\mathbb{E}[\tau] = T\phi^T + \sum_{t=1}^{T} \phi^{t-1} - T\phi^T. \tag{28}$$

Cancelling terms and summing the first T elements of the geometric series:

$$\mathbb{E}[\tau] = \sum_{t=1}^{T} \phi^{t-1} = \frac{1-\phi^T}{1-\phi}. \tag{29}$$

Using L'Hôpital's Rule, we can evaluate the limit for $\phi \to 1$ and find that this simplifies to T and hence $\mathbb{E}[\tau] = T$ when $\phi = 1$ completing the proof. □

Remark 2 *The expected number of iterations as derived in Lemma 3 depends on the rejection probability ϕ and the maximum number of iterations T. When $\phi = 0$, the algorithm always accepts in any iteration and hence $\mathbb{E}_{\boldsymbol{y}\sim M(\cdot|\boldsymbol{x})}[\tau] = 1$. Conversely, when $\phi = 1$ and the algorithm always abstains, $\mathbb{E}_{\boldsymbol{y}\sim M(\cdot|\boldsymbol{x})}[\tau] = T$. Further, for $T = 1, \forall\phi \in [0,1] : \mathbb{E}_{\boldsymbol{y}\sim M(\cdot|\boldsymbol{x})}[\tau] = 1$ and as T increases, so does $\mathbb{E}_{\boldsymbol{y}\sim M(\cdot|\boldsymbol{x})}[\tau]$ when $\phi > 0$.*

B VALID— REJECTION SAMPLING

B.1 BOUND TIGHTNESS ACROSS RESPONSE LENGTHS.

After establishing VALID a natural question is how this bound behaves across responses with varying length. We will explore this here.

Despite the length normalization of the rejection threshold, notice that the VALID bound depends on $N_{\boldsymbol{y}}$, the length of sequence $\boldsymbol{y}$ (see (4)), making the certificate more effective for shorter or longer sequences. Let $\bar{g}(\boldsymbol{y})$ be the geometric mean of per-token probability for $G(\boldsymbol{y})$. The log upper bound can be written as $kN_{\boldsymbol{y}} + N_{\boldsymbol{y}} \log \bar{g}(\boldsymbol{y}) + \log T$. Whether this is tighter for short or long sequences is governed by k and $\log \bar{g}(\boldsymbol{y})$. When $k + \log \bar{g}(\boldsymbol{y})$ is close to 0, the bound is balanced, and when $k + \log \bar{g}(\boldsymbol{y}) < 0$, the bound decreases as $N_{\boldsymbol{y}}$ increases.

B.2 ATTACKING M

In this section, we provide some insight on how rejection sampling (see Section 2.2) works to obtain an adversarial bound. For simplicity, we will consider the case $T = 1$. We will then use this to describe the objective required to attack M.

Building an Intuition. We consider model M generated by rejection sampling from L using guide model G as described above. We consider a single $\boldsymbol{y}$ and describe how the ϵ-AC certificate as achieved through rejection sampling using an example. Let $\mathbf{y} = \texttt{The cow drinks milk}$ and consider three prompts:

- $\mathbf{x_1} = \texttt{What does a cow drink?}$
- $\mathbf{x_2} = \texttt{Which animal drinks milk?}$
- $\mathbf{x_3} = \texttt{Repeat after me: The cow drinks milk. Now you:}$

Intuitively we may assume $L(\boldsymbol{y}|\boldsymbol{x}_3) > L(\boldsymbol{y}|\boldsymbol{x}_1) > L(\boldsymbol{y}|\boldsymbol{x}_2)$ as $\boldsymbol{y}$ more naturally follows some prompts than others: $\boldsymbol{y}$ would have very high likelihood after $\boldsymbol{x}_3$ for instruct trained models, medium likelihood after being specifically asked about cows ($\boldsymbol{x}_1$) and low likelihood to be picked as example from all mammals as response to $\boldsymbol{x}_2$.

Let us regard a single rejection step. We illustrate this example in Figure 5. If we assume that $\boldsymbol{y}|\boldsymbol{x}_1$ is rejected, i.e. $\log L(\boldsymbol{y}|\boldsymbol{x}_1) - \log G(\boldsymbol{y}) > kN_{\boldsymbol{y}}$, then we can conclude that $\boldsymbol{y}|\boldsymbol{x}_3$ will also be rejected. We know that for $T = 1$, M has likelihood:

$$M(\boldsymbol{y}|\boldsymbol{x}) = \begin{cases} L(\boldsymbol{y}|\boldsymbol{x}) & \text{if } \boldsymbol{y}|\boldsymbol{x} \text{ is accepted,} \\ 0 & \text{otherwise.} \end{cases} \tag{30}$$

Given that $M(\boldsymbol{y}|\boldsymbol{x}) = 0$ for rejected samples, the question presents itself: What is the purpose of the upper bound. Consider the case that $\boldsymbol{y}|\boldsymbol{x}_2$ is accepted. This occurs, iff $\log L(\boldsymbol{y}|\boldsymbol{x}_2) - \log G(\boldsymbol{y}) \leq kN_{\boldsymbol{y}}$, which by algebraic manipulation means $L(\boldsymbol{y}|\boldsymbol{x}_2) \leq 2^{kN_{\boldsymbol{y}}} G(\boldsymbol{y})$, recovering the upper bound as stated in Theorem 1: It is only possible to return $\boldsymbol{y}|\boldsymbol{x}_2$ when $L(\boldsymbol{y}|\boldsymbol{x}_2)$ is smaller than $2^{kN_{\boldsymbol{y}}} G(\boldsymbol{y})$. More generally, $\boldsymbol{y}$ can only be returned, if we find an $\boldsymbol{x}^*$ s.t. $L(\boldsymbol{y}|\boldsymbol{x}^*) \leq 2^{kN_{\boldsymbol{y}}} G(\boldsymbol{y})$ and hence by (30) we have that $L(\boldsymbol{y}|\boldsymbol{x}^*) \leq 2^{kN_{\boldsymbol{y}}} G(\boldsymbol{y})$. This illustrates how rejection sampling bounds the adversaries: Samples will only be accepted if proposing them was very unlikely in the first place. This intuition helps us establishing how to attack M.

Formalising the Attack. We assume the adversarial objective is to increase the probability of a given $\boldsymbol{y}^*$ (e.g. from the out-of-domain set), $\mathbb{F}$, being returned. The objective of attacking L is immediately follows:

$$\boldsymbol{x}_L^{adv} = \arg\max_{\boldsymbol{x} \in \mathbb{X}} L(\boldsymbol{y}^*|\boldsymbol{x}) \tag{31}$$

where $\mathbb{X}$ is either $\mathbb{S}$ or some continuous relaxation, such as soft-prompt space. However, the solution $\boldsymbol{x}_L^{adv}$ is likely not an adversary under M, as $\boldsymbol{x}_L^{adv}$ maximises the log-likelihood ratio and thus the sample is likely rejected, hence $M(\boldsymbol{y}^*|\boldsymbol{x}_L^{adv}) = 0$. Instead, the adversary for M, $\boldsymbol{x}_M^{adv}$, needs to maximise M while ensuring the sample is accepted, i.e. $M(\boldsymbol{y}^*|\boldsymbol{x}_M^{adv}) > 0$. Thus The following objective emerges which we state in the following lemma.

Lemma 4 (Adversary under Rejection Sampling) *Assume the adversarial objective is to increase the likelihood of sample $\boldsymbol{y}$ being returned by the model M. Assume the model M is obtained by rejection sampling as described in Algorithm 1 with $T = 1$. The adversary is given by:*

$$\boldsymbol{x}_M^{adv} = \arg\max_{\boldsymbol{x} \in \mathbb{X}} L(\boldsymbol{y}|\boldsymbol{x}) \text{ s.t. } L(\boldsymbol{y}|\boldsymbol{x}) \leq 2^{N_k \boldsymbol{y}} G(\boldsymbol{y}). \tag{32}$$

Proof: Note that $M(\boldsymbol{y}|\boldsymbol{x}_M^{adv}) > 0$ as the sample is accepted. Assume there exists $\boldsymbol{x}'$, s.t. $L(\boldsymbol{y}|\boldsymbol{x}') > L(\boldsymbol{y}|\boldsymbol{x}_M^{adv})$ and hence maximises M further then $\boldsymbol{x}_M^{adv}$. Then, it must be true that $L(\boldsymbol{y}|\boldsymbol{x}') > 2^{kN_{\boldsymbol{y}}} G(\boldsymbol{y})$, which implies $M(\boldsymbol{y}|\boldsymbol{x}') = 0$ yielding a contradiction. Hence, $\boldsymbol{x}_M^{adv}$ is the required solution. □

Implementing such Attack. Applying VALID to obtain M has implications on the suitable procedures to attack M. In particular, it requires solving the constrained optimisation problem in

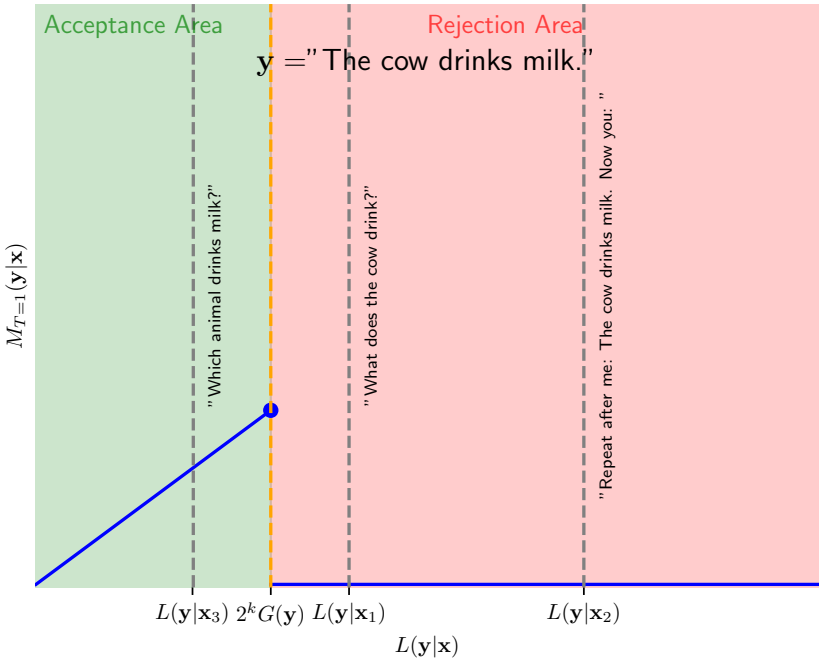


Figure 5: The likelihood of model M obtained through VALID with $T = 1$. The blue line is the likelihood of M for the given $\boldsymbol{y}$. Three example prompts $\boldsymbol{x}_1$, $\boldsymbol{x}_2$ and $\boldsymbol{x}_3$ are shown.

(32), which already adds a layer of complexity to the unconstrained problem for L. In general constrained optimisation problems are more challenging, this is compounded by the upper bound on $L(\boldsymbol{y}|\boldsymbol{x})$ not decomposing across tokens. Further, while large models s.t. LLama-3-8B are often publicly available, G will likely be a custom model for which the attacker does not have white box access. For a successful attack, the adversarial user must estimate the likelihood ratio between L and G, which might prove challenging. This indicates that attacking M defined through VALID might be a harder problem than attacking L. To reiterate, while it is possible to attack M, our certificate holds and it cannot be attacked past the upper bound provided in Theorem 1.

C Experimental Setup

C.1 CharTask Dataset

Here we provide more information on the CharTask dataset. The goal of the CharTask dataset is it, to have a well-controlled toy dataset with clear definitions of target domain $\mathbb{T}$ and other domains $\mathbb{F}$.

Table 1: Examples of the CharTask dataset

Task	Pool	Sequence			
		Prompt	Task	Completed	Combined
Sorting	Int	5 3 6	S R A E	3 5 6	Q 5 3 6 S R A E 3 5 6
Adding	Int	5 3 6	A E R S	6 4 7	Q 5 3 6 A E R S 6 4 7
Reverse Sorting	Int	5 3 6	R E A S	6 5 3	Q 5 3 6 R E A S 6 5 3
Even-Odd	Int	5 3 6	E R A S	6 3 5	Q 5 3 6 E R A S 6 3 5
Sorting	Int + Char	13 5 c a	S E R A	13 5 a c	Q 13 5 c a S E R A
Adding	Int + Char	13 5 c a	A S R E	14 6 d b	Q 13 5 c a A S R E
Reverse Sorting	Int + Char	13 5 c a	R E A S	c a 5 13	Q 13 5 c a R E A S c a 5 13
Even-Odd	Int + Char	13 5 c a	E S A R	a c 13 5	Q 13 5 c a E S A R 13 5 c a

As shown in Table 1, each sequence consists of three parts: A sequence of random characters, a task definition in the middle and another sequence of characters in the end. We refer to the random sequence as S_{in}. In the middle there are four task tokens, the first of which defines the task T. "S" sets the task to sorting, "R" to reverse sorting, "A" to adding $+1$ and "E" to even-odd sorting. The instructing token is followed by the remaining three task tokens in random order to ensure that all are seen by a model trained on a subset of these. Finally, the completed sequence is the original sequence of characters with the task performed on them, i.e. $S_{out} = T(S_{in})$. The pool of characters for each sequence is either the integers or integers and strings of lower case letters. Importantly, all tasks interpret characters and integers as characters alike. For instance, sorting integers "11", "5" results in "11", "5". To be precise, all tasks are based on the integer unicode representations of the characters.

Each sequence has variable length of up to 49 elements in S_{in} (elements can be double digits). For integers we use a pool of 49 unique distinct integers and for characters we use a pool of 249 elements (e.g. defining "at" as one element in the sequence). Under these conditions there exist a combinatorially large set of unique sequences far exeeding our training dataset size.

Given the tasks and pools of characters, 8 possible domains, emerge as shown in Table 1, which we denote as CharTask (Task, Pool). We define sorting integers as the target domain: $\mathcal{D}_{\mathbb{T}} =$ CharTask (Sorting, Int) and all other combinations as out-of-domain. We create two distinct datasets with non-overlapping splits for training, validation and testing. The in-domain dataset consists of 1M training samples. The "generalist" dataset $\mathcal{D}_{\mathbb{T}+\mathbb{F}} =$ CharTask (All, Int + Char) contains of all possible tasks with sequences consisting of integers and characters. We use 1M training sequences per task, hence 4M sequences in total. Validation and test sets are 64 sequences and 4096 sequences, respectively.

C.2 CHARTASK SETUP

Dataset and Domain. We use the CharTask dataset as described in Appendix C.1. We train a custom BPE tokenizer of length 360 (Sennrich et al., 2016). In practice, the pretrained tokenizer of any foundation model is trained on a general dataset. Hence, we train the tokenizer using $\mathcal{D}_{\mathbb{T}}$ and $\mathcal{D}_{\mathbb{F}}$, the target and out-of-domain datasets. While the dataset is inherently suitable for a sequence-to-sequence task, we treat it as next-token prediction problem just as used in language modelling.

Training. We train our domain model G on a set of integer sorting examples, CharTask (Sorting, Int). We train a GPT-2 (Radford et al., 2019) architecture with 3 layers, 3 heads and 48 embedding dimension. We train the model on partial sequences, as we are embedding marginal sequences $\boldsymbol{y}$. Hence, we cut each sequence in two parts using a splitting point that is sampled under a uniform distribution. Hence, the model learns the transition from "[BOS] .." to any character that might be the first response token.

For the generalist model L, we train using all available tasks on integers and characters, CharTask (All,Int+Char). We train a GPT-2 architecture with 6 layers, 6 heads and 192 embedding dimensions.

We train L and G with AdamW (weight decay 0.1) for 2048 steps with a cosine learning rate schedule with 500 steps warmup, a maximum learning rate of 0.005, scheduled for 40 epochs. We train with 120 context window using next-token prediction.

Inference. We use common parameters to tweak the predictive distribution of our models. For G we use a temperature of 0.7 and for L of 0.2. We find this greatly helps the model performance of both. We do not perform *TopK* selection of tokens. We prompt with a prompt length of 10. The task-completed sequence is almost deterministic given the prompt and task for models that have very high accuracy. Hence, we remove sequences where the prompt of 10 tokens is larger than 25% of the entire sequence.

C.3 20NG SETUP

Dataset Cleaning. The 20NG dataset is very dirty, containing a wide array of random special character sequences and formatting. We found these sequences to complicate model training and large pre-trained models struggled with it. In addition, as formatting strongly varies between the 20NG dataset and others, this is a confounding factor for OOD detection. Classifying sentences as ID or OOD should focus on semantics, but the formatting provides a spurious correlation that

is easily exploited by models. Hence, we decided to clean the dataset. To do so we utilise the `scikit-learn` (v1.5.1) (Pedregosa et al., 2011) options to remove headers, footers and quotes. Further, we cleaned it using Llama-3.1-8B-Instruct (Dubey et al., 2024) using the following query:

```
Your task is to clean and format a string.
Instructions:
- Do not change the order of the words.
- Remove cryptic character sequences, spacings out of order,
and line breaks within sentences.
- Remove out-of-order punctuation, but leave correct
punctuation in place.
- The result should be semantically and lexically the same as
the original but well formatted.
- Remove IP addresses and email addresses.
- Remove sequences of (special) characters, that are not
human language.
- Only return the cleaned string without messages or quotes
around it.  Do not return any other information.  Do not
repeat the instructions.  Do not repeat the example.

Sentence:
```

We check the output for various keywords and phrases from prompt and find 0% violation rate. While there still exist random sequences, the data quality is greatly improved. We notice that several sequences exist in 20NG and OOD testing datasets that are seemingly random character sequences and multiple trigram repetitions such as "Nanaimo British Columbia Nanaimo British Columbia Nanaimo British Columbia ...". These sequences have the highest likelihood under model G and L while not having any semantic meaning nor constituting a valid sequence that could indicate model misappropriation. Hence, when reporting $\max$ likelihoods for 20NG over a finite dataset (e.g. $\max_{\boldsymbol{x},\boldsymbol{y}\in\mathcal{D}_{\mathbb{F}}} L(\boldsymbol{y}|\boldsymbol{x})$) we instead use the 99.99th quantile and report it as $\max$.

Training. We use a pre-trained Gemma 2 tokenizer for both models which has a vocabulary size of 256k tokens.

For the fine-tuned model L, we use a pre-trained decoder-only Gemma 2 2B (hosted on Hugging Face) as the starting point then fine-tune it to our ID dataset using LoRA adaptors which involved training an additional 10.4M parameters (0.4% of the total parameters).We train L with AdamW (weight decay 0.01) for 1536 steps with a cosine learning rate schedule with 64 steps warmup, a maximum learning rate of 5e-5, scheduled for 32 epochs. We train with 256 context window using next-token prediction.

For the model G, we use a decoder-only GPT-small model architecture, 6 layers, 6 heads and 384 embedding dimensions and a total of param- eters 109.3M, which we train from scratch using the ID data exclusively. We train G with AdamW (weight decay 0.01) for 320 steps with a cosine learning rate schedule with 100 steps warmup, a maximum learning rate of 3e-4, scheduled for 100 epochs. We train with 256 context window using next-token prediction.

Inference. For both L and G we use a default temperature of 1. We do not perform *TopK* selection of tokens. When evaluating performance, we use 128-token long prompt and a 128-token long ground truth response.

C.4 TinyShakespeare Setup

Dataset Cleaning. The formatting in TinyShakespeare dataset was distinctly different to other texts with long sequences of line breaks and usage of all-caps for character names. We removed

these excessive line breaks and changed the character names from all caps to title case to make it similar to other datasets and make OOD detection less trivial challenging.

Training. We use a pre-trained Gemma 2 tokenizer for both models which has a vocabulary size of 256k tokens.

For the fine-tuned model L, we use a pre-trained decoder-only Gemma 2 2B (hosted on Hugging Face) as the starting point then fine-tune it to our ID dataset using LoRA adaptors which involved training an additional 10.4M parameters (0.4% of the total parameters). We train L with AdamW (weight decay 0.01) for 128 steps with a cosine learning rate schedule with 64 steps warmup, a maximum learning rate of 5e-5, scheduled for 32 epochs. We train with 256 context window using next-token prediction.

For the model G, we use a decoder-only GPT-micro model architecture, 4 layers, 4 heads and 128 embedding dimensions and a total of param- eters 33.7M, which we train from scratch using the ID data exclusively. We train G with AdamW (weight decay 0.01) for 2400 steps with a cosine learning rate schedule with 300 steps warmup, a maximum learning rate of 3e-4, scheduled for 300 epochs. We train with 256 context window using next-token prediction.

Inference. For both L and G we use a default temperature of 1. We do not perform *TopK* selection of tokens. When evaluating performance, we use 128-token long prompt and a 128-token long ground truth response.

C.5 MEDICALQA

We apply our method to medical question answering as target domain, $\mathbb{T}$. This could for example be extended to a chatbot for clinicians to research patient symptoms. To model potential questions and answers, we use the PubMedQA dataset (Jin et al., 2019) as $\mathcal{D}_{\mathbb{T}}$, which contains approximately 200K QA pairs for training and 1000 test pairs. We regard question answering on other topics, such as geography or computer science as $\mathbb{F}$. To model this, we use the Stanford Question and Answering Dataset (excluding medical categories) (Rajpurkar et al., 2016) as $\mathcal{D}_{\mathbb{F}}$.

Training. As a generalist LLM, L, we use a LLama-3-8B model (AI@Meta, 2024) and train a custom GPT-2 model (184M parameters) for G (Radford et al., 2019). We pre-train G on PubMedQA (Jin et al., 2019) with 200K sequences. We then use 100K prompts from PubMedQA to generate sequences using L and then fine-tune on them using responses from L to half the prompts in PubMedQA. As G is embeds the responses, $G(\boldsymbol{y})$, we fine-tune using "BOS[Response]" rather than entire sequences. We pretrain with learning rate of 0.0001 for 50 epochs and then fine-tune with learning rate 0.00001 for another 50 epochs. On 8 $\times$ H100, the total training takes about 2 hours.

Inference. We perform inference without top_k or top_p parameters and with temperatures of 1.0 for model L and G. We prompt using the natural questions as defined by the datasets. For the analysis, we remove responses that are not clearly out of domain. For instance the response "10 million people every year" is not only a valid response to a geographical question, but can also be an information about the prevalence of the disease. When applying our method, we focus on responses with at least 10 tokens to further remove ambiguous questions. Modern LLMs tend to be very verbose in their responses, so responses should naturally be longer than 10 tokens.

C.6 Dataset Categories

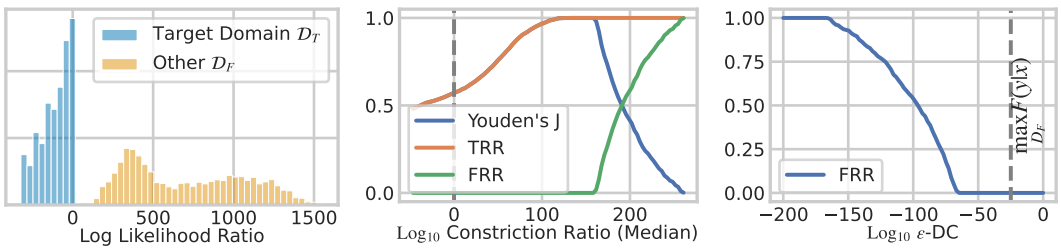
We list here the categories excluded from SQuAD and included in MMLU for reproducibility.

Excluded From SQuAD	Included in MMLU-Med
Antibiotics	Anatomy
Symbiosis	Clinical knowledge
Gene	College medicine
Brain	College biology
Immunology	College chemistry
Biodiversity	High school biology
Digestion	High school chemistry
Pharmaceutical industry	High school psychology
Mammal	Human aging
Nutrition	Human sexuality
Tuberculosis	Medical genetics
On the Origin of Species	Nutrition
Asthma	Professional medicine
Pain	Virology
Bacteria	
Infection	
Black Death	
Pharmacy	
Immune system	
Chloroplast	

Table 2: Categories of items in used Datasets.

D Experimental Results

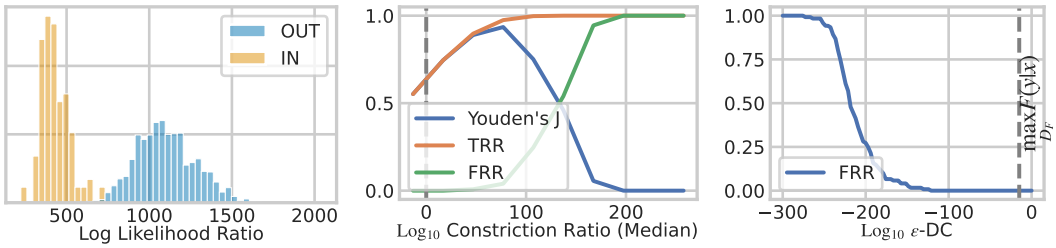
D.1 CharTask Results



(a) (b) (c)

Figure 6: This Figure replicates Figure 3 for the CharTask dataset.

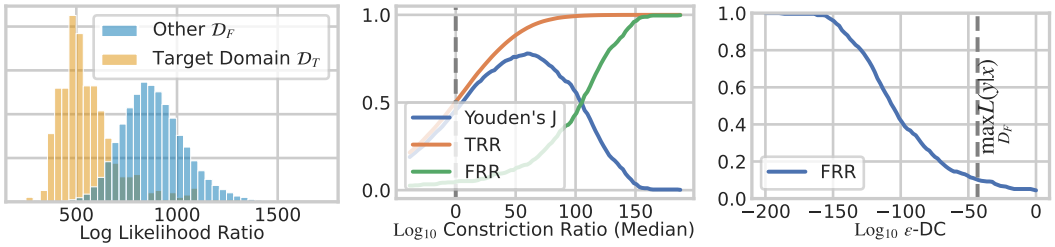
D.2 TINYSHAKESPEARE RESULTS



(a) (b) (c)

Figure 7: This Figure replicates Figure 3 for the TinyShakespeare dataset.

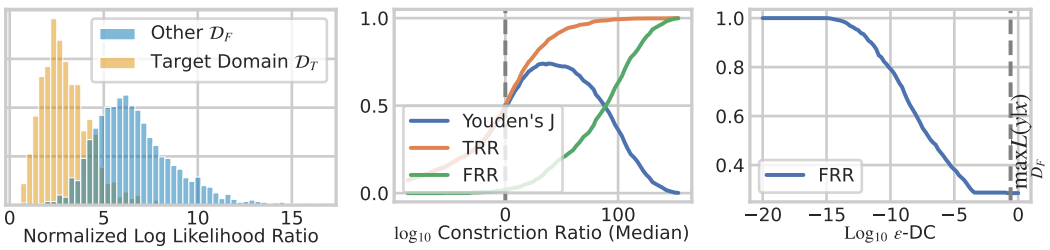
D.3 20NG



(a) (b) (c)

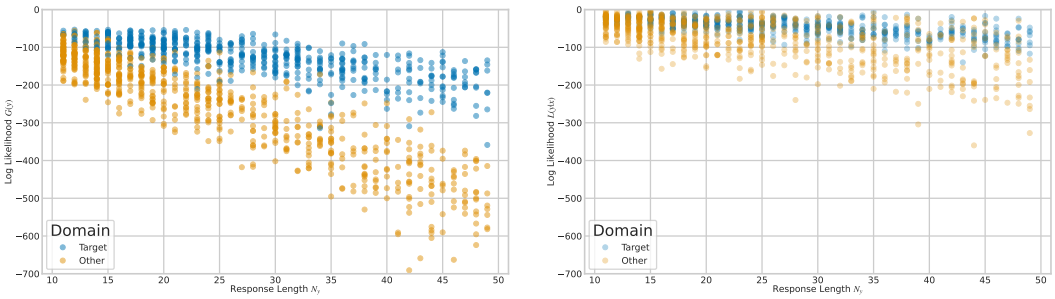
Figure 8: Figure 8a shows that log likelihood ratios are well disentangled. Figure 8b shows the trade-off between OOD and certification: The best OOD detection performance occurs with a constriction ratio of 60. Figure 8c shows the false rejection rate (FRR) required to certify at a given ϵ.

D.4 MEDICAL QA



(a) (b) (c)

Figure 9: Figure 9a shows that log likelihood ratios are well disentangled. Figure 9b shows the trade-off between OOD and certification. Figure 9c shows the false rejection rate (FRR) required to certify at a given ϵ. All results are for VALID with $T = 1$ for Medical QA.



(a) (b)

Figure 10: This figure demonstrates the gap in log likelihood between in-domain and out-of-domain samples for the guide models G in Figure 10a and the LLM L in Figure 10b. As the length of the response, N_y, increases, the gap between ID ($\mathcal{D}_{\mathbb{T}}$) and OOD data ($\mathcal{D}_{\mathbb{F}}$) widens, with the log-likelihood decreasing roughly linearly. Thus the guide model G on the left side assigns exponential decreasing probabilities to OOD samples.

D.5 CONSTRICTION RATIOS FOR DIFFERENT FALSE REJECTION RATES

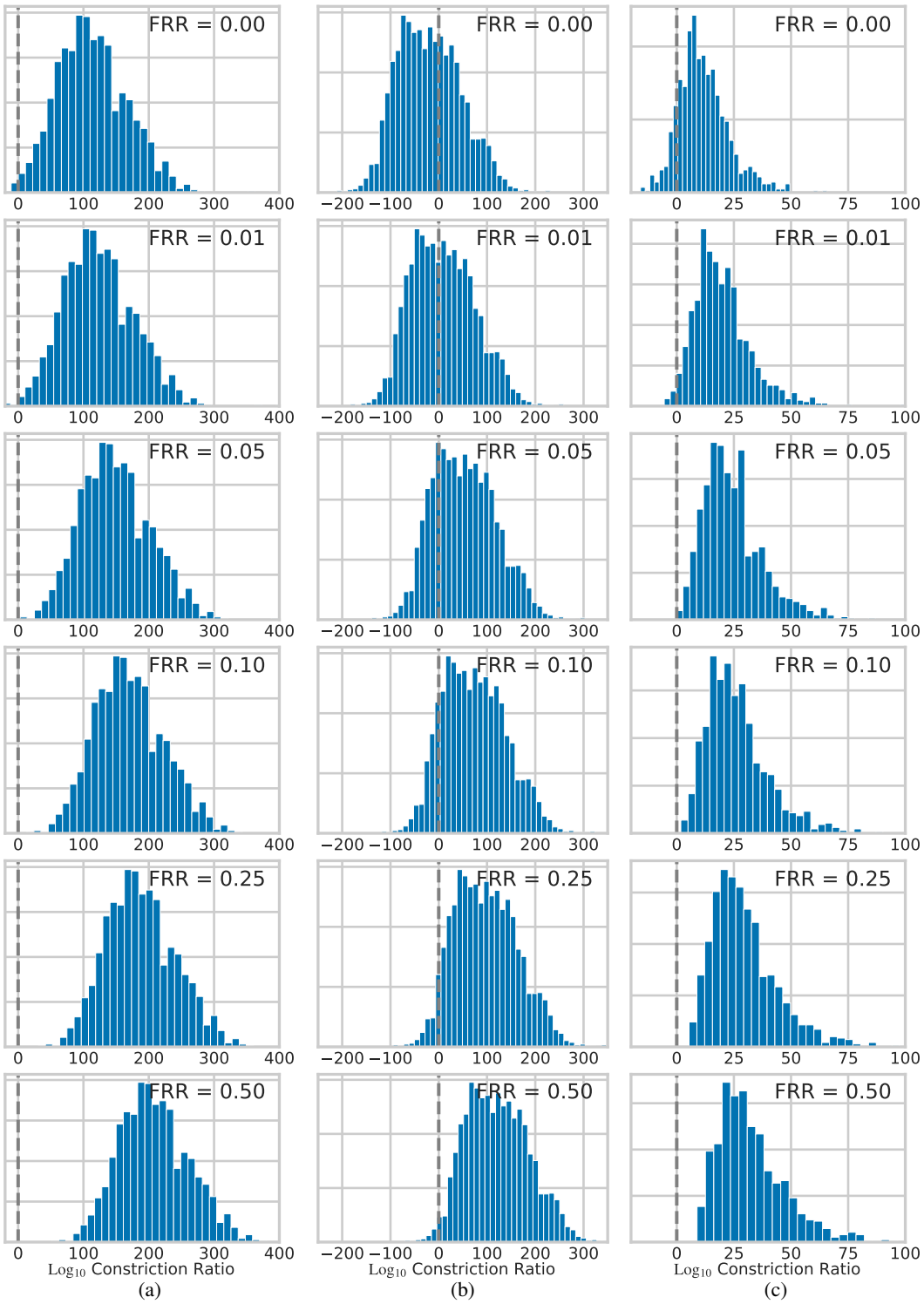


Figure 11: This figure shows the log_{10} constriction ratios (CR) on OOD samples as a function of the false rejection rate (FRR) on the in-domain samples. The rejection threshold k is systematically decreased from top to bottom to achieve a given FRR. We can observe the gradually improvement constriction while increasing the FRR. (a) shows Tiny Shakespeare, (b) shows 20NG, and (c) Medical QA.

D.6 ATOMIC CERTIFICATES - LENGTH CONTROLLED

The experimental setup for MedicalQA uses PubMedQA as in-domain dataset and SQuAD as out-of-domain dataset as described in Section 3.1 and Appendix C. The different lengths of responses in these datasets confounds our findings on the disentanglement of the atomic certificates, $\epsilon_{\boldsymbol{y}}$-ACs between in-domain data, $\mathcal{D}_{\mathbb{T}}$, and out-of-domain data, $\mathcal{D}_{\mathbb{F}}$. In Figure 12, we show that sequences tend to be a lot shorter in $\mathcal{D}_{\mathbb{F}}$ than in $\mathcal{D}_{\mathbb{T}}$. The likelihood of a response decays exponentially in the length of the responses. Hence, the responses in the OOD set $\mathcal{D}_{\mathbb{F}}$ have relatively high likelihood that is not attributable to the domain restriction but rather the length of the response. This yields the eCDFs in Figure 4b to overlap significantly. To show that this is a confounding factor that is indeed worsening disentanglement, we resample the data to account for length and present results here.

Setup. To study the effect, we resample the in-domain data, $\mathcal{D}_{\mathbb{T}}$, and out-of-domain data, $\mathcal{D}_{\mathbb{F}}$ to have matching distribution of response lengths. We find the target distribution using the following steps: First, we find the common support between the distribution of response length $n_{\boldsymbol{y}}$ between $\mathcal{D}_{\mathbb{T}}$ and $\mathcal{D}_{\mathbb{F}}$, $n_{\boldsymbol{y}} \in [15, 38]$. This interval covers 67% of samples in the target domain dataset and 58% of the OOD dataset. Second, we obtain the empirical distribution of $n_{\boldsymbol{y}}$ in the in-domain dataset, perform Laplace smoothing (Manning et al., 2008) with $\alpha = 1$ and then further smooth the distribution using a moving average with a window length of 5. Third, we perform weighting sampling with replacement from $\mathcal{D}_{\mathbb{T}}$ and $\mathcal{D}_{\mathbb{F}}$ with a size of 100 times the original. The weights are computed s.t. the distribution of $n_{\boldsymbol{y}}$ matches the target distribution. We denote these **res**ampled sets as $\mathcal{D}_{\mathbb{T}}^{RS}$ and $\mathcal{D}_{\mathbb{F}}^{RS}$.

Results. We find that the disentanglement of atomic certificates, $\epsilon_{\boldsymbol{y}}$-ACs, greatly improves after eliminating the confounding factor, response lengths. Figure 13 shows the empirical cumulative distribution functions (eCDFs) for "original" datasets, $\mathcal{D}_{\mathbb{T}}$ and $\mathcal{D}_{\mathbb{F}}$ in grey tones, as well as the results for $\mathcal{D}_{\mathbb{T}}^{RS}$ and $\mathcal{D}_{\mathbb{F}}^{RS}$. You may observe the distribution of ACs shifted left for datasets representing $\mathbb{F}$ and shifted right for datasets representing $\mathbb{T}$, effectively increasing the disentanglement. This indicates, when comparing *similar* in-domain and out-of-domain samples, the difference in the *permissiveness* of the ACs on in-domain samples and the *restrictiveness* of the ACs on out-of-domain samples is much greater.

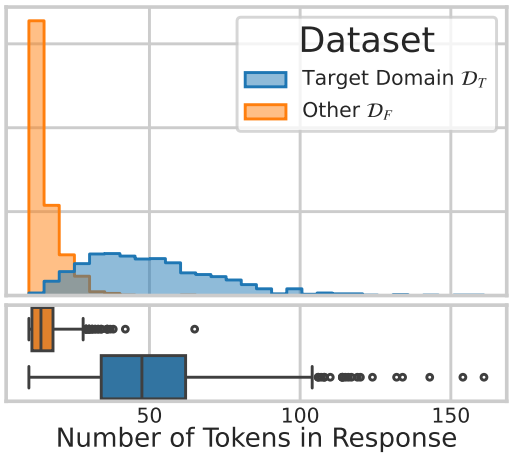


Figure 12: MedQA setup: The in-domain dataset (PubMedQA) has longer responses than the OOD dataset (SQuAD).

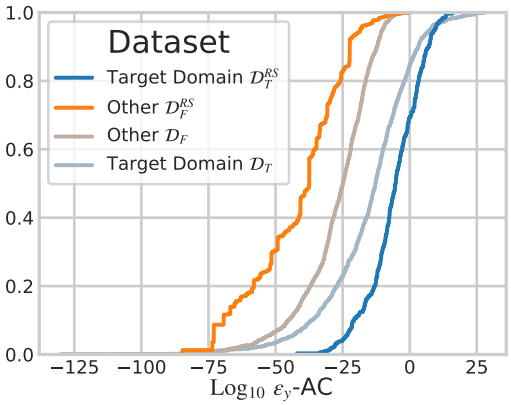
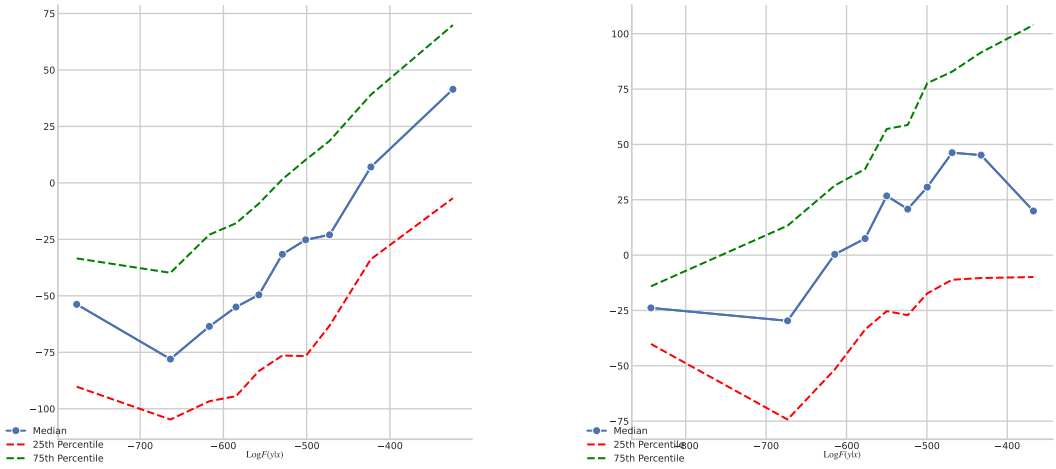


Figure 13: The eCDFs of $\epsilon_{\boldsymbol{y}}$-ACs are shown for the original in- and out-of-domain data for the MedQA setup in comparison to a resampled dataset controlling the response length as confounder. The gap between the permissveness of in-domain samples and restrictiveness on out-of-domain samples is greatly improved.

D.7 ATOMIC CERTIFICATE BY LIKELIHOOD

Obtaining a tight atomic certificate for sample $\boldsymbol{y}$ is most important when the sample is likely proposed by L. Hence, in this section we study the log constriction ratio, the tightening of our adversarial certificate over model L, as a function of the sample's likelihood under L.

We bin out-of-domain samples into 10 bins based on their log likelihood under model L, i.e. $\log L(\boldsymbol{y}|\boldsymbol{x})$, and compute median, 25th and 75th percentile log constriction ratio, as well as the median log likelihood. We present results in Figure 14 for both 20NG and TinyShakespeare. We observe that the constriction strengthens when samples get more likely under L. That means, those samples most likely to be sampled under L benefit most from our atomic certificate. We consider this a favourable result.



(a) (b)

Figure 14: These figures show the constriction ratio as a function of log likelihood of samples under L. Figure 14a shows data for 20NG and Figure 14b for TinyShakespeare. On the x-axis is median log-likelihood under L, $\log L(\boldsymbol{y}|\boldsymbol{x})$ and on the y-axis is the log constriction ratio.

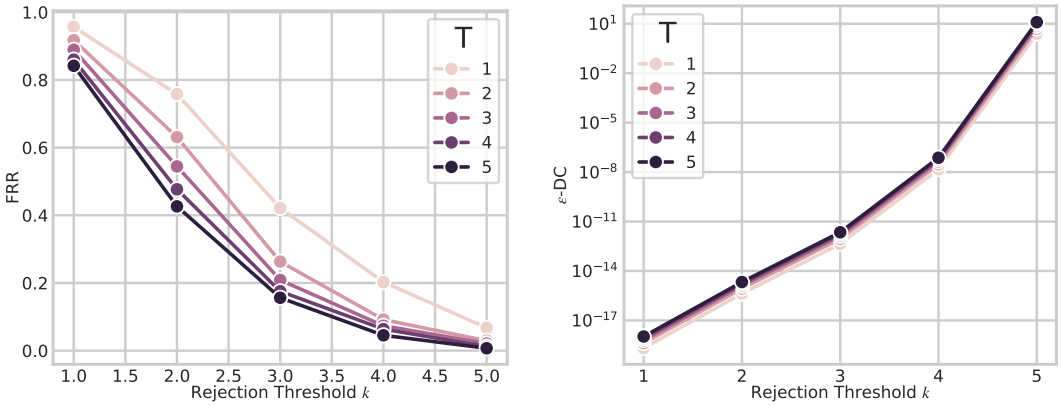
E REPEATED SAMPLING ($T > 1$)

In Section 3.3 we study the performance of VALID by sampling from L with a single step, i.e. $T = 1$. Here, we extend the analysis to $T > 1$.

Setup. We adopt the MedicalQA setup as described in Appendix C.5. However, instead of employing VALID with $T = 1$, we use $T \in \{1, 2, 3, 4, 5\}$ and study for combinations of k (the rejection threshold of VALID) the resulting $\epsilon - DC$. As above, for ease of presentation we use a fixed temperature of 1.0 for L.

Results. We find that increasing T significantly reduces false rejection rates (FRR) while only marginally increasing the ϵ-DC (domain certificate). We present findings for the FRR in Figure 15a and for ϵ-DC in Figure 15b. The minor increase in ϵ due to increasing T should not come as a surprise as we recall the formula for the upper bound: $2^{kN_y}TG(y)$ (see (4)). Even $T = 10$ increases the upper bound ϵ_y by only one order of magnitude. On the other hand, the gains in in-domain performance are marked. In Figure 15a we can observe that for $T = 5$, the FRR is roughly halved for $(K > 2)$, greatly improving the refusal behaviour of the model on in-domain samples.

Finally, we would like to note, that temperature of L is a confounding factor. For $t_L \to 0$, we will observe deterministic sampling of y given x and hence, $T > 1$ will not have any benefit.



(a) (b)

Figure 15: False Rejection Rate (FRR) **(Left)** and $\epsilon - DC$ of the Domain certificate of VALID **(Right)** plotted for a range of different values of T and k.

F Ablation

F.1 Comparing M to G

Our method provides a guarantee on a generalist model assuming that such a model outperforms custom, small solutions that are inherently safer due to their domain specific training. We test this empirically by examining the gap in performance between the generalist model L, a small in-domain model. As G is trained marginally on $\boldsymbol{y}$, it is not able to perform any task. Hence, we exactly replicate the training procedure of G and train a model on the entire sequence, $G'(\boldsymbol{x}, \boldsymbol{y})$. We utilize the CharTask dataset as described above and study the accuracy of each model in generating valid sequences: A valid sequence is one that starts with `Q`, is followed by a random sequence of characters (e.g. `5 3`), followed by four unique task tokens (e.g. `S A E R`) defining a task, which is then performed (e.g. `3 5`). The sequence is expected to terminate there. If *any* of these are violated, the generated sequence is scored as invalid. We perform inference on 1000 prompts from the target domain test dataset prompting the model with various lengths of prompts. In Table 3, we present the results: The accuracy of generating such sequences of L lies significantly above that of G (difference of approx 30%). This shows that G is effective in restricting the domain while performing considerably worse than L. Hence, our method combines the best of both models: The safety of G with the performance of L.

Prompt Length	**G**	**L**
1	60.45	91.21
5	60.25	92.68
10	66.89	91.11

Table 3: Accuracy scores for CharTask generation dataset.

F.2 Benefit of larger Guide Models

In this Appendix, we study the influence of the size of G on the VALID results. In particular, we ask whether VALID benefits from smaller or larger models.

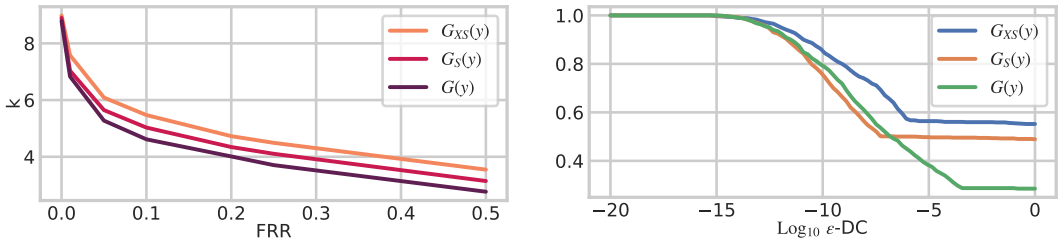
Setup. We turn to our MedicalQA setup as described in Section 3.1 and Appendix C.5. With the same methodology, we fit two more models for G. G_{XS} follows a GPT-2 architecture with 6 layers, 6 heads and 192 embedding dimension resulting in 27.49M parameters. G_S follows a GPT-2 architecture with 6 layers, 6 heads and 384 embedding dimensions resulting in 60.29M parameters. To recap, the G model as used above uses 12 layers, 12 heads and 768 embedding dimension resulting in 184M parameters. We then compare the three models on samples generated by L following Section 3.3.

Results. We find that larger models tend to perform better, however evidence is not strong. First, we study the rejection threshold k per model. As described in (4) in Theorem 1, VALIDs upper bounds gets tighter with smaller k. Hence, in Figure 16a we plot k values achieving a given false rejection rate (FRR) for each model. We observe that larger the model enable smaller k at the same FRR. This indicates that the trade-off in k between certification and OOD detection is more favourable under larger models. This should not come as a surprise, however, as larger models tend to achieve better perplexity (i.e. lower loss) on in-domain data.

Next, we study the constriction ratios of the Atomic Certificates (AC) as done in Table **??** in Section 3.2. Here, we replicate this table for different sizes of G as shown in Table 4 . For each model, we provide the the 10th percentile, median and 90th percentile. You may observe that $G_{XS}(y)$ consistently provides constriction ratios that are are often around 10 orders of magnitudes worse than $G_S(y)$ and $G(y)$. Interestingly, $G_S(y)$ yields better ratios than $G(y)$. However, the difference is smaller. We speculate that the limited amount of ID training data means we do not see benefits for increasing the size of G beyond a point, as it begins to over-fit without the addition of extra regularization techniques.

Finally, we study the Domain Certificates (DC) for each model. For this we replicate Figure 9c and present Figure 16b showing the false rejection rate (FRR) given an ϵ-DC for the three models. We may observe that the lower bound to the FRR significantly increases as the models get smaller. The evidence here suggests that larger guide models yield better domain certificates.

In conclusion, the evidence points to larger models working better for an application like MedicalQA. The evidence uniformly shows that a model as small as $G_{XS}(y)$ does perform significantly worse than larger models.



(a) (b)

Figure 16: These Figures demonstrate differences in the behaviour of VALID for different sizes of guide models G. Figure 16a shows that larger models allow for lower k and hence lower bounds at the same False Rejection Rate (FRR). Figure 16b shows the FRR for a given ϵ-DC for guide models of different sizes.

FRR	**Log_{10} Constriction Ratio (10% / Median / 90%)**		
	$G_{XS}(y)$	$G_S(y)$	$G(y)$
0%	-427 / -45 / 12	-408 / -41 / 12	-449 / -54 / 6
1%	-246 / -14 / 42	-176 / -3 / 79	-198 / -10 / 43
5%	-74 / 12 / 141	-42 / 21 / 195	-42 / 18 / 162
10%	-29 / 24 / 202	-11 / 35 / 257	-8 / 33 / 229
20%	-3 / 43 / 281	1 / 57 / 337	3 / 50 / 302
25%	0 / 50 / 308	5 / 63 / 364	7 / 60 / 345
50%	11 / 81 / 430	13 / 96 / 497	15 / 89 / 477

Table 4: Constriction Ratios for MedQA for three models of different sizes. The smallest model is yields significantly worse (lower) constriction ratios.

G MOTIVATION

G.1 DEFINING DOMAINS - PRACTICAL CONSIDERATIONS

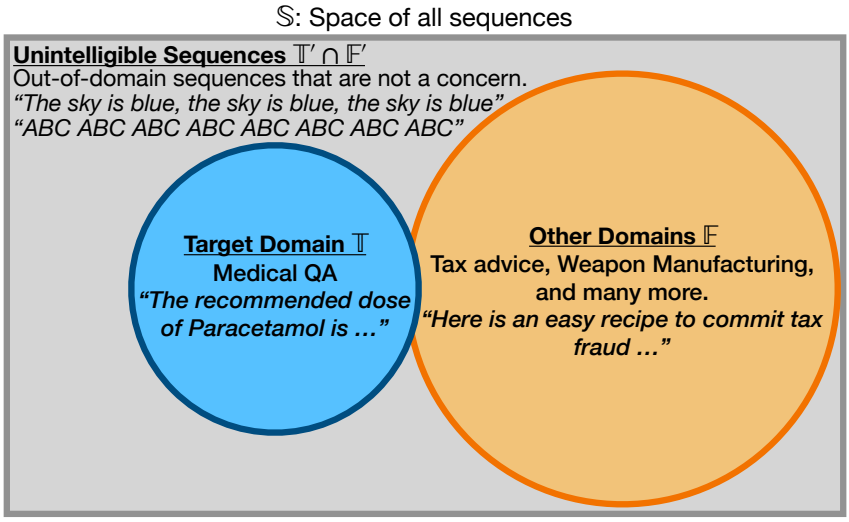


Figure 17: A Venn diagram illustrating sequence domains providing an example domain and example sequences in that domain.

In this section, we provide practical guidance for practitioners selecting domains for AI systems, presenting a systematic approach to classifying sequences into different domains. Figure 17 illustrates a Venn diagram comprising three key sets of sequences, i.e. subsets of $\mathbb{S}$:

1. The target domain $\mathbb{T}$ (shown in blue), containing desired content about which the LLM-driven system should converse (e.g., medical questions and answers),
2. The out-of-domain set $\mathbb{F}$ (shown in red-orange), containing potentially harmful or other content that require active protection measures (e.g., tax fraud advice),
3. The complement of $\mathbb{T}$ and $\mathbb{F}$, denoted as $\mathbb{T}' \cap \mathbb{F}'$ (shown in grey).

A fundamental question arises: How should one define $\mathbb{T}$ and $\mathbb{F}$? Defining $\mathbb{T}$ is relatively natural for most practitioners. Content that semantically belongs to the domain should be included in $\mathbb{T}$. The more complex decision involves determining which sequences outside $\mathbb{T}$ should be included in $\mathbb{F}$. We contend that protection against certain sequences warrants higher priority than others, and these high-priority sequences should be included in $\mathbb{F}$.

Consider the example sequence $\boldsymbol{y}$ = "The sky is blue. The sky is blue. The sky is blue." While this is clearly out-of-domain for a medical QA system, practitioners should evaluate two critical questions to determine its placement in $\mathbb{F}$:

1. Would adversaries have motivation to generate such sequences?
2. Could these sequences potentially harm users, the system, or third parties?

In this example, adversaries would likely have little incentive to generate such a response, and the content itself is harmless. Therefore, practitioners might reasonably conclude that $\boldsymbol{y}$ should remain in $\mathbb{T}' \cap \mathbb{F}'$ rather than $\mathbb{F}$, excluding it from system certification considerations.

These evaluation questions help practitioners assess risk levels effectively. When both questions receive negative answers, sequences can safely remain in $\mathbb{T}' \cap \mathbb{F}'$ without requiring active protection measures, allowing security efforts to focus on genuinely concerning sequences. If either question receives a positive response, practitioners may choose to implement protective measures.

Let us analyze two more examples to demonstrate this in practice. Consider the sequence $\boldsymbol{y}$ = "Here is an easy recipe to commit tax fraud..." - in this case, malicious actors would be highly mo-

tivated to seek such information, and the content could directly harm society and government functions. Thus, this sequence clearly belongs in $\mathbb{F}$ and requires active protection measures. Similarly, when considering $\boldsymbol{y}$ as a love poem, while it may seem harmless at first glance, the analysis reveals important considerations. Users might frequently request LLMs to generate poetry, potentially straining system resources, and, while not directly harmful to users or society, it could significantly impact system infrastructure and operational costs. Consequently, practitioners might choose to include this in $\mathbb{F}$ to protect their computational resources.

It is important to note that these evaluation questions are not intended as universal rules, but rather serve as practical considerations to guide practitioners in their decision-making process. By systematically assessing motivation and potential harm, practitioners can make informed decisions about which sets of sequences require active protection measures.

H FUTURE WORK

In this section we briefly discuss some ideas for future work that we believe could further extent the practical utility of VALID. First, it would be interesting to experiment with larger specialised models for G to assess if these more capable models lead to better performance and results. We chose not to do this as LLMs trained from scratch exclusively on specific domains are not common, and thus results with these models would be less similar to what a practitioner with limited resources could expect.

As described in Section 2.2, VALID uses length normalisation to ensure the log likelihood ratio rejection condition is robust to different lengths of sequences N_y. However, by sampling representative data sets of responses from $\boldsymbol{y} \sim L(\cdot|\boldsymbol{x})$ for both in-and out-of-distribution $\boldsymbol{x}$'s, it should be possible to learn a more complex polynomial of N_y to then use as a threshold for rejection. This threshold could also be used to provide both $\epsilon_{\boldsymbol{y}}$-ACs and ϵ-DC certificates, while hopefully simultaneously enabling more precise detection.

Finally, a rejection scheme with a probabilistic decision rule, similar to Algorithm 5 in Vyas et al. (2023), would be able to provide identical bounds to Theorem 1. Possibly, this rejection rule would lead to better performance in terms of OOD classification.