TOPICAL POLICY BRIEF

Large Language Models

Yoon Kim
Jacob Andreas
Dylan Hadfield-Menell

November 28, 2023

MIT Schwarzman College of Computing

MIT Washington Office

aipolicy.mit.edu

Authors

Yoon Kim
Assistant Professor
Department of Electrical Engineering and Computer Science

Jacob Andreas
Associate Professor
Department of Electrical Engineering and Computer Science

Dylan Hadfield-Menell
Assistant Professor
Department of Electrical Engineering and Computer Science

Executive Summary

- Large language models (LLMs) are a transformative technology. But their broad applicability, evolving capability, unpredictable behavior, and widespread availability present challenges that should be addressed through regulation – and that need to be taken into account when developing regulations.

- The regulation of LLMs requires consideration of how general the uses of the model are and how broadly available models are.
    - General-purpose models, with broad applications, need incentives for developers to disclose associated risks and ensure responsible use. Task-specific models, which are deployed for a particular purpose, may be regulated through existing domain-specific regulations.
    - LLMs (general-purpose or otherwise) may be released through an application programming interface (API), a hosted service, or as downloadable models. Regardless of the method of release, regulation should require risk assessment by the model provider before release.

- Potential technical innovations that would improve LLM safety include verifiable attribution, hard-to-remove watermarks, guaranteed forgetting, better guardrails, and auditability. These can help mitigate safety concerns, improve user trust, and prevent misuse. Regulation and funding should be structured to encourage work on, and deployment of, these innovations.

- LLM regulation can be accomplished through standards that can be progressively tightened and/or by offering incentives for safe deployment. A mixture of both approaches will likely be useful.

Part 1: Challenges of Large Language Models

Large language models (such as GPT-4) serve as the foundation for some of the most capable and general-purpose AI systems that exist today, and hold the potential to have a transformative impact across multiple industries. However, they also possess several characteristics, described below, that make it important to regulate them and that should be taken into account when doing so.

Broad applicability. Fundamentally, language models are text processing systems that output pieces of text in response to pieces of input text. However, because text is a flexible medium with which to convey and receive information, and because LLMs can be readily specialized to different tasks, they have the potential to become a broad technology with applications across multiple domains. The diversity of their use cases makes it difficult – if not impossible – for providers of general-purpose LLMs to specify the full universe of their intended or possible uses.

Rapidly-evolving capability. While the capabilities of LLMs can be harnessed for positive applications, they can also be exploited for malicious purposes, e.g., to disseminate disinformation at scale, automate cyberattacks, and more. Risks stemming from their misuse are commensurate with their capabilities, which are quickly evolving. For example, OpenAI's GPT-4 was initially released to the public as a text-only model but was recently updated with image and voice processing capabilities, substantially expanding the scope of possible applications and associated risks.

Widespread availability. With many organizations training and releasing models through interfaces or even direct downloads, it has become relatively easy for individuals and organizations to access these models at modest cost. While ready access to such useful technologies can foster innovation and bring the benefits of AI to the public at large, it also makes it difficult to monitor and regulate use.

Unpredictable behavior. LLMs rely on deep learning methods that are hard to interpret and control, making them susceptible to unpredictable and undesirable behavior. For example they may generate meaningfully different responses to inputs that differ from one another in only minor ways; surface sensitive information or biases contained in their training sets; and even "hallucinate" ostensibly plausible outputs that are factually incorrect. They are moreover vulnerable to "jailbreaks" that can allow a user to bypass guardrails set up by the model provider by manipulating the input. Insofar as LLMs often serve as a component of a larger AI stack, such unpredictable behavior may propagate to downstream components in undesirable ways. It is currently not possible to provide strict guarantees against such unpredictable behavior.

Fluency. LLMs can produce fluent and contextually relevant responses to almost any user input. Due to humans' natural inclination to perceive fluent language as genuine and authoritative, these models can engender a presumption of factuality on the part of the user, even when the output may be erroneous or misleading. This may be especially problematic in areas such as health, law, and finance, where the user might not have the expertise to critically evaluate the accuracy of the model's response.

Part 2: A Framework for Large Language Model Regulation

Language models are not monolithic. They are developed by a wide range of actors and have different purposes, risks, and potential benefits. We identify two key distinctions that should be taken into account when shaping regulation: the degree of model scoping and the model release type. These categories and their implications are described below, and the categories and the appropriate types of regulation for each are outlined in this table:

	In-House Models	Hosted Models	Downloadable Models
General Purpose Models	• Disclosure	• Use-Case Monitoring and Disclosure • Guardrails to Prevent Misuse • Auditing	• Pre-deployment Risk Assessments • Documentation of Training Data and Evaluation • Documentation and Disclosure of Inappropriate Uses and Risks
Specialized Tools	• Compliance with Relevant Domain-Specific Rules and Regulations • Support for External Audits and Red Teaming	• Compliance with Relevant Domain-Specific Rules and Regulations • Support for External Audits and Red Teaming • Guardrails to Prevent Misuse	• Compliance with Relevant Domain-Specific Rules and Regulations • Pre-deployment Risk Assessment • Documentation of Training Data and Intended Use Cases

Types of Models. A key differentiating characteristic among LLMs is generality. At one end of the spectrum are general models that are released without a specific purpose in mind. For example, consider the use of an Application Programming Interface (API) service such as the GPT-4 API from OpenAI. The model is marketed for a wide range of uses, such as document drafting, programming, translation, and tutoring, among others. At the other end of the spectrum are specialized, task-specific models. These models are created or specialized for specific purposes. For example, Duolingo, a service that helps subscribers learn new languages, uses a task-specific language model to explain mistakes a learner makes or to have conversations in the new language.

A key difference between these two categories is the implied contract between the developer and the downstream user. In the case of general models, there are few or no promises made by the developer. Without regulation, model developers may share as much or as little detail about the system as they choose. As a result, it may be important for regulation to create incentives or requirements for model developers to understand and disclose the patterns of use that are likely to occur; clearly communicate what constitutes responsible use of the model; and identify use cases that are inappropriate or out-of-scope for the model. Regulators should require that these inappropriate use cases be disclosed and that guardrails and standards be established to prevent such uses.

Furthermore, regulation should require or encourage as much transparency about such models as possible. For example, regulation could require the creation of industry standards for model documentation and evaluation, as well as for identifying and disclosing the range of appropriate model uses. To promote transparency, regulation should create a notification process for the release of general models. This will allow regulators to understand the market and build capacity for future regulation as best practices and industry standards emerge.

On the other hand, in deploying a task-specific model, the scoping of the deployment and user expectations can and should be clearer. This type of deployment is more amenable to existing domain-specific regulation. For example, when a developer trains or produces a model for a regulated domain, such as resume filtering for job applicants, it is easier to identify and comply with relevant regulations, such as laws that prevent discrimination against protected groups. Regulation (or, where necessary, statute) should clarify that companies or individuals who develop or deploy task-specific models are responsible for complying with existing domain-specific regulations.

Crucially, if a general model is used with minimal modification for a regulated domain, the company or individual that *deploys* the model should be considered responsible for ensuring that the use is a responsible and legal use of the general model. (Law and regulation should also determine the extent to which the deployer should have recourse if they believe the provider of one or more of the underlying AI models is ultimately responsible for the harm.)

Types of Releases. A second key variation among LLMs is the way a model is released. Solaiman (2023) describes the different ways that a model developer can choose to release models. This ranges from models that are developed and used in-house at a company, to models that are hosted for access through a web interface or API, to models that are downloadable and can be run locally. Each of these different categories has different benefits and risks.

There are three factors that regulators should use to assess the tradeoffs. First, different release types enable different levels of transparency. Different types of model deployments make it harder or easier for independent auditors, researchers, or regulators to understand how a model can be used and to identify vulnerabilities and bias. Second, different release types give model developers different levels of control over how a model is used. Third, different release types change a model developer's ability to update models to fix bugs or security vulnerabilities.

In-house models. In-house development and deployment of models give model developers many options to control model use and update models. However, they may provide little external transparency to regulators or independent researchers. It might be hard to understand how the model was developed, what purposes it is being used for, or that a model is being used at all. On the other hand, the deploying entity is in a strong position to understand and decide how the model is used. Because development and deployment are performed by a single entity, it is most natural to regulate these models through existing domain-specific regulations.

For example, an in-house model used for resume filtering would need to comply with existing non-discrimination laws. Because it may be hard to determine when LLMs are being used, regulation might require that companies disclose when language models are used in a regulated domain and clarify that systems need to comply with existing rules and regulations. To support investigations into model uses, regulation should require that copies of the weights and parameters of models used in regulated domains be stored for a set period of time.

Hosted models. In the case of hosted models like ChatGPT that facilitate use by customers (either through specific contracts or individual use by members of the public), there are more opportunities for transparency to regulators and the general public. However, the extent to which this is possible depends on the type of access allowed and the degree to which details of the training data and procedure are disclosed. In this case, model developers have less control over how the model is used: they can communicate how to use the model responsibly but need to rely on guardrails around the model to prevent intentional or unintentional misuse. In this situation, regulation should require that model developers deploy guardrails that prevent reasonably predictable misuse and hold developers liable when such safeguards are not deployed. Because model developers can monitor how the system is used in practice, regulation should require that developers audit and disclose prevalent use cases. This is especially important for general models that have a broad and evolving set of use cases.

Downloadable models. The final relevant category is models that can be downloaded and run directly by end users. These are sometimes referred to as open-source or open-use models. Downloadable models are easier for independent researchers or auditors to investigate. This can help identify vulnerabilities in models or problematic biases. Clear documentation of the model training data and processes helps to enable these benefits. However, this type of model release also presents several challenges. These include an increased potential for misuse, difficulty monitoring use, and difficulty pushing updates to models. A key challenge for downloadable releases is that most guardrails to prevent misuse can be removed by bypassing input/output filtering or with relatively minimal retraining of the model. Regulation should require developers to assess potential risks prior to deployment and establish liability for developers that distribute models that are used to cause foreseeable harm.

Part 3: Innovations that Could Improve Large Language Models

LLM technology is still in a state of flux. There are a number of potential technical innovations that could help mitigate the safety concerns articulated above but that do not yet exist or are insufficiently robust to be deployed today. Beyond ensuring the safe use of current models, LLM governance should promote investment in the development of these innovations. The following capabilities seem to be particularly important goals:

Verifiable attribution: methods for training models such that, when they output a factual claim, they also produce a (genuine) citation to an existing, human-authored document containing evidence for that claim. Such methods would be useful for improving user trust and making it easier for users to recognize incorrect or hallucinated output. (Guaranteeing that a system does not hallucinate is much more challenging.)

Hard-to-remove watermarks: methods for inserting digital signatures in model-generated text so that it can later be recognized. Such watermarks must be robust to significant alterations to text, including alterations produced by other models. Still, they would make it possible for search engine providers and public discussion boards to identify machine-generated content, among other benefits.

Guaranteed forgetting: algorithms that, given a model and some targeted piece of information, comprehensively remove that information from the model (so that it cannot be accessed by users via any prompt or recovered from the model's parameters). Safety and privacy could be addressed by such a forgetting mechanism: it could be used to prevent models from outputting dangerous information (e.g., how to engineer a bioweapon), sensitive information (e.g., biographical details about specific private individuals), or copyrighted content.

Guardrails: procedures for preventing models from responding to particular user requests. Such procedures are less stringent than the "forgetting" mechanism described above, as models might still "know" certain harmful pieces of information, but decline to provide them in the service of malicious user requests. There are instances where guardrails are more appropriate than "forgetting." As a concrete example, a model for content moderation must be able to recognize racial slurs in input text, but should possess guardrails that prevent it from generating slurs as output. Guardrails could be designed to prevent undesirable behavior resulting from user-provided input text (through what is sometimes known as "prompt injection") or, more ambitiously, to prevent undesirable behavior resulting from training models on user-provided data (through what is known as "fine-tuning").

Auditability: frameworks that make it easy (1) to discover whether released models exhibit previously unidentified failure modes, and (2) to verify that protections against known vulnerabilities work as intended. Unlike the other advances above, this includes both a technical component (models that are easy to inspect and computational tools that assist with inspection) as well as a social component (creating incentives for "white-hat" users to discover failures modes before "black-hat" users do, as in other computer security applications). Also, standards need to be created that describe what constitutes a responsible and effective audit. Otherwise, audits can be easily manipulated.

All of the topics above are active areas of research in the academic community. There is also significant industry movement toward attribution tools (as many of the largest LLM developers are also search engine providers). However, we believe that the other four innovations described above receive insufficient investment (especially relative to research on directly improving model capabilities). These are, therefore, research topics on which regulation and increased public investment might have an outsized impact.

Part 4: Modes of Regulation

There are at least two models, not mutually exclusive, for how regulation of LLMs might be carried out. One model involves a *progressive tightening of standards*: a regulatory regime might permit "unsafe" (or insufficiently safe) models to be deployed today, while requiring providers of such models to notify users of potential safety issues. Such a regulatory regime would also formally put providers on notice that models a set number of years in the future will be regulated more strictly (e.g., required to attribute all generated factual assertions). This model of establishing standards that gradually increase in strictness with advance notice has been employed in other industries. For example, the 2007 Energy Independence and Security Act established a progressive ratcheting-up of fuel economy standards for cars, and the National Highway Traffic Safety Administration announced its backup camera mandate four years before it went into effect.

A second model is to offer *regulatory incentives for safe deployment.* In a "positive incentives" model, LLM providers might be subject to reduced liability or oversight (e.g. reduced training data disclosure obligations or less frequent audits) in exchange for voluntary implementation of any of the above capabilities. This model has similarities to existing "safe harbor" mechanisms – for example, in which private landowners agree to take specific actions to protect endangered species, and regulatory agencies in turn agree not to impose certain land use restrictions. In addition, LLMs should be covered by existing liability laws, which create "negative incentives" – enhancing products to reduce the chances of being sued. We anticipate that a mixture of both models will be useful for ensuring safe and effective deployment of LLM-based applications.