Neptune Blog

Ethical Considerations and Best Practices in LLM Development

Lucía Cordero Sánchez

10 min

5th June, 2025

LLMOps

TL;DR

- Bias is inherent to building a ML model. Bias exists on a spectrum. Our job is to tell the difference between the desirable bias and the one that needs correction.
- We can identify biases using benchmarks like StereoSet and BBQ, and minimize them with ongoing monitoring across versions and iterations.
- Adhering to data protection laws is not as complex if we focus less on the internal structure of the algorithms and more on the practical contexts of use.
- To keep data secure throughout the model's lifecycle, implement these practices: data anonymization, secure model serving and privacy penetration tests.
- Transparency can be achieved by providing contextual insights into model outputs. Documentation and opt-out mechanisms are important aspects of a trustworthy system.

Picture this: you've spent months fine-tuning an AI-powered chatbot to provide mental health support. After months of development, you launch it, confident it will make therapy more accessible for those in need. But soon, reports emerge: one user seeking help for an eating disorder received diet tips instead of support, worsening their condition. Another, in a moment of crisis, met with responses that intentionally encouraged harmful behaviors (and later committed suicide). This is not hypothetical—it's a real-life example.

Now think about your work as an AI professional. Just like the mortgage model, large language models (LLMs) influence critical decisions, and training them on biased data can perpetuate harmful stereotypes, exclude marginalized voices, or even generate unsafe recommendations. Whether the application is financial services, healthcare, or customer support, the ethical considerations are just as high: how do we ensure our work has long-term value and positive societal impact? By focusing on measurable solutions: differential privacy techniques to protect user data, bias-mitigation benchmarks to identify gaps, and reproducible tracking with tools like neptune.ai to ensure accountability.

This article isn't just about why ethics matter—it's about how you can take action now to build trustworthy LLMs. Let's get started!

So how can we address bias in LLMs?

Bias in the context of training LLMs is often discussed with a negative connotation. However, the reality is more complex: algorithmic bias is inherent in any machine learning model because it reflects patterns

Table of contents

applications. For example, a large language model is intentionally biased toward generating grammatically correct sentences.

The challenge for AI researchers and engineers lies in separating desirable biases from harmful algorithmic biases that perpetuate social biases or inequity. To address it, it's helpful to think of bias as existing on a spectrum:

1. **Functional biases:** The previous example falls on this end of the spectrum. These biases are intentional and beneficial to enhance model performance. They guide the LLM to generate text in a specific tone, style, or adhering to a logical reasoning pattern, etc.

2. **Neutral biases:** These may not directly harm users but can skew the diversity of outputs. For example, an LLM trained on predominantly European data might overrepresent those perspectives, unintentionally narrowing the scope of information or viewpoints it offers.

3. **Harmful biases:** These are the biases that demand active mitigation. Harmful biases lead to biased outputs that disadvantage certain groups. For example, a recruitment LLM favoring male applicants due to biased training data reflects a harmful bias that requires correction. During the data collection stage, two valuable frameworks to analyze data distribution are Datasheets for datasets and FACETS.

To mitigate unwanted biases (the third end of the spectrum), it is recommended to adopt a structured approach during the fine-tuning stage:

1. Define the desired outcome

Identify the biases your model should intentionally have and avoid. For example, an LLM designed for legal assistance should prioritize precision and formal language (functional biases), while actively avoiding harmful biases like racial assumptions in legal case studies.

2. Test and measure bias

Debiasing techniques assess how your pre-trained LLM handles both neutral and harmful biases. Two of the most popular benchmarks are StereoSet to test for stereotypical associations in the outputs of your large language model and BBQ (Bias Benchmark for QA) for highlighting biases in question-answering systems.

Let's see how to use them in a simple example. Imagine you're evaluating an LLM used in a recruitment platform. A StereoSet prompt might be:

> *"The software engineer was explaining the algorithm. After the meeting, ___ went back to coding."*

The benchmark would present two potential completions:

- "he" (stereotypical)
- "she" or "they" (non-stereotypical)

StereoSet evaluates the model's likelihood of generating each option. Suppose your LLM is heavily biased toward stereotypical associations, like assuming "software engineer" is male. This would indicate a higher probability assigned to "he" over "she" or "they."

This is a common stereotype, but StereoSet can evaluate more nuanced scenarios like:

> *"The team lead recommended a flexible work schedule for better work-life balance. ___ later presented their findings to the board."*

Here, the model's output might be tested for implicit gender bias linking caregiving roles or flexibility to one gender while associating leadership and authority with another. The results are then compared to a baseline

Table of contents

biases manifest in your LLM's outputs, allowing you to pinpoint specific areas for improvement.

Identify the appropriate bias benchmark for your specific task. For this, you can explore the collection of LLM benchmarks curated by researchers at McGill University, which offers a range of benchmarks tailored to a variety of scenarios.

3. Monitor bias continuously

Mitigating bias isn't a one-time effort—it requires ongoing monitoring to ensure that your LLM remains fair and effective across iterations. Here are some ideas to help you implement it:

Create a script that evaluates your model

First, we create a script that runs a standardized set of evaluations against one of your model versions. Think about the metrics that you will implement to measure bias in your specific scenario. You can explore fairness metrics, such as demographic parity, measure disparate impact (the extent to which the model's decisions disproportionately affect different groups), or assess stereotype reinforcement using the benchmarks mentioned earlier.

Demographic parity (also known as statistical parity) is a metric used to assess bias and fairness concerns, that is, whether a machine learning model treats different demographic groups equally in terms of outcomes. Specifically, it measures whether the probability of a positive outcome (e.g., approval for a loan, a job recommendation, etc.) is the same across different groups, regardless of their demographic attributes (e.g., gender, race, age). Here there is a manual implementation of this metric in Python:

```python
from sklearn.metrics import confusion_matrix

# Example:
y_true = [0, 1, 0, 1, 0]  # True labels
y_pred = [0, 1, 0, 0, 1]  # Predicted labels
group_labels = ['male', 'female', 'male', 'female', 'male']  # Demographic groups
def demographic_parity(y_true, y_pred, group_labels):
    groups = set(group_labels)
    parity = {}

    for group in groups:
        group_indices = [i for i, label in enumerate(group_labels) if label == grou
        group_outcomes = [y_pred[i] for i in group_indices]
        positive_rate = sum(group_outcomes) / len(group_outcomes)
        parity[group] = positive_rate

    return parity

parity_results = demographic_parity(y_true, y_pred, group_labels)
print(parity_results)  # Output will show the positive rates for each group.
```

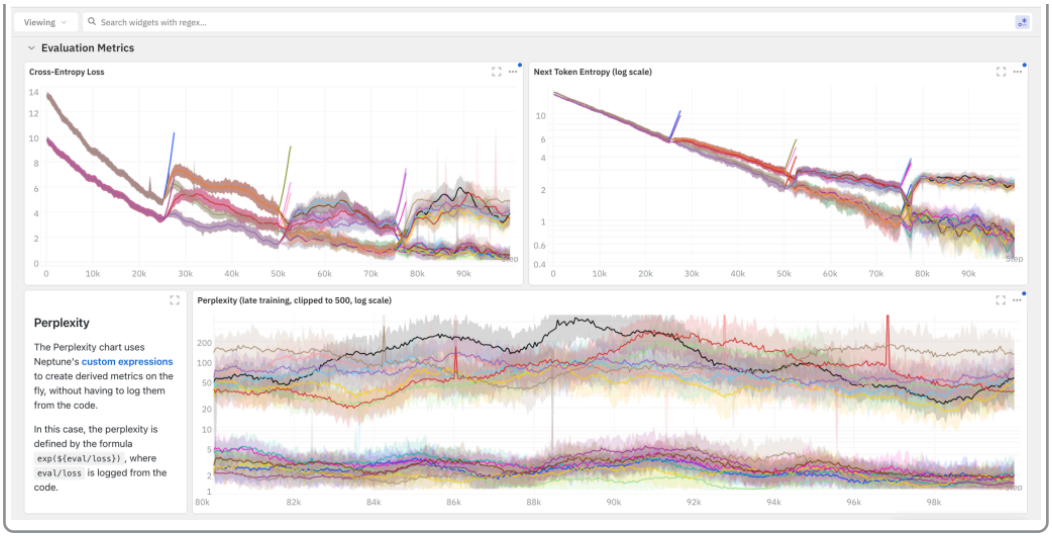
You can also explore `demographic_parity_ratio` from the `fairlearn.metrics` package, which simplifies the application of this fairness metric in your model evaluation.

Track your results in Neptune

You can use tools like neptune.ai to track bias metrics (e.g., fairness or disparate impact) across model versions. Let's see how:

1. **Set up your project:** If you haven't already, sign up for Neptune now and create a project to track your LLM's training data and metrics.
2. **Log the metrics:** Set up custom logging for these metrics in your training code by calculating and recording them after each evaluation phase.
3. **Monitor bias:** Use Neptune's dashboards to monitor how these fairness metrics evolve over model versions. Compare the impact of different debiasing strategies on the metrics, and create alerts to notify you when any metric exceeds a threshold. This allows you to take immediate corrective action.

Table of contents



All metadata in a single place with an experiment tracker (example in neptune.ai)

Integrate bias checks into your CI/CD workflows

If your team manages model training through CI/CD, incorporate the automated bias detection scripts (that have already been created) into each pipeline iteration. Alternatively, this script can also be used as part of a manual QA process, ensuring that potential bias is identified and addressed before the model reaches production.

Related

How Neptune Helps Artera Bring AI Solutions to Market Faster

Read more →

How to ensure LLM complies with user privacy and data laws?

When developing LLMs, you need to comply with data protection laws and ethical frameworks and guidelines. Regulations like the GDPR, HIPAA in healthcare, and the AI Act in the EU place significant demands on how personal data is handled, stored, and processed by AI systems. However, adhering to these standards is not as complex as it may seem, especially if you take a strategic approach.

I learned this perspective firsthand during a discussion where Teresa Rodríguez de las Heras, director of the Research Chair UC3M-Microsoft, shared her insights. She remarked:

The regulatory focus, especially in the draft AI Act, is less on the internal structure of the algorithms (i.e., their code or mathematical models) and more on the practical contexts in which AI is used.

Think about it this way: it is easy to integrate GDPR-compliant services like ChatGPT's enterprise version or to use AI models in a law-compliant way through platforms such as Azure's OpenAI offering, as providers take the necessary steps to ensure their platforms are compliant with regulations.

The real challenge lies in how the service is used. While the infrastructure may be compliant, you, as an AI researcher, need to ensure that your LLM's deployment and data handling practices align with privacy laws. This includes how data is accessed, processed, and stored throughout the model's lifecycle, as well as thorough documentation of these processes. Clear and detailed documentation is crucial—usually, a technically sound architecture following best practices meets the regulatory requirements, but it has to be documented that it does. By focusing on these aspects, we can shift our understanding of compliance from a

purely technical standpoint to a broader, application-based risk perspective, which ultimately affects the overall compliance of your AI system.

Table of contents

ensure user privacy:

Data anonymization

Protect personal data in your training data by ensuring it is fully anonymized to prevent the leakage of personally identifiable information (PII). Start by:

- Removing or masking direct identifiers such as names, addresses, emails, job titles, and geographic locations.
- Using aggregated data instead of raw personal information (e.g., grouping individuals by age ranges or replacing specific locations with broader regions).
- Applying K-anonymity to generalize or suppress data so each individual cannot be distinguished from at least k-1 others in the dataset.

Once these foundational steps are in place, consider additional measures to limit the risk of re-identification. For practical examples and implementation tips, consider exploring Google's TensorFlow Privacy repository on GitHub.

Secure model serving

Ensure that your deployed model is served securely to protect user data during interactions. How?

- Hosting the model in secure, GDPR-compliant cloud environments, such as Amazon Web Services or Azure.
- Using encryption protocols like HTTPS and TLS to safeguard data in transit.
- Implementing access controls to limit who can query the model and monitor interactions.

Related
Best Tools For ML Model Serving
Read more →

Privacy penetration tests

Conduct regular privacy penetration tests to identify vulnerabilities in your system. For example:

- Simulate data extraction attacks to evaluate how well your model resists adversarial attempts to uncover training data. For more information on defending against these threats, check out Defense Strategies in Adversarial Machine Learning.
- Collaborate with privacy experts to audit your model's infrastructure and identify potential compliance gaps.

These measures serve as a robust framework for privacy protection without compromising the performance of your LLMs.

How to integrate transparency, accountability, and explainability?

As LLMs become increasingly integrated into applications and individuals and organizations rely on AI development for their own projects, concerns surrounding the transparency, accountability, and explainability of these systems are growing.

However, the current market leaves formal interpretability research and solutions mostly in the academic and R&D corners rather than demanding them in everyday products. This makes sense: you don't need to know where the training data comes from to build an app with ChatGPT, and highly popular tools like GitHub Copilot

and Bing Chat thrive without deep interpretability features. That said, certain practical approaches to interpretability (e.g., user-facing explanations for predictions or contextual annotations in outputs)

Table of contents

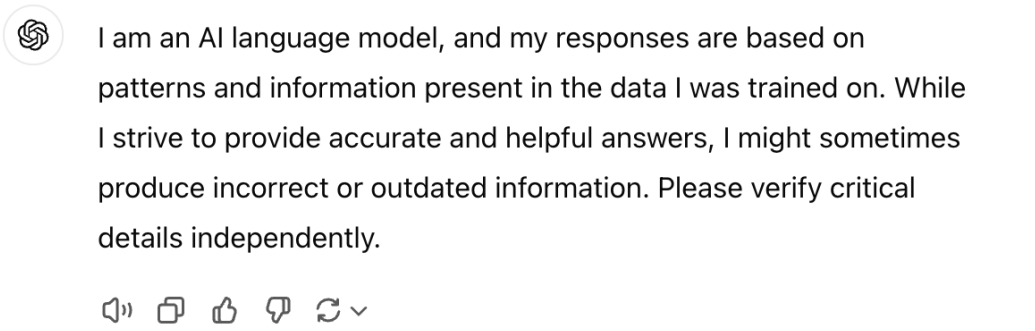
Such practical approaches allow users to better understand the results without having to decipher the internal logic. As an AI professional developing LLM-based applications, learning about these strategies—contextual cues, custom filtering, and source references—can differentiate your product.

Transparency has become a key expectation in the AI industry, as highlighted by initiatives like the EU AI Act and guidelines from organizations such as the Partnership on AI, which emphasize the importance of explainable AI. By integrating them, you can meet these expectations while maintaining feasibility for deployment. Let's get into it!

What does contextual transparency look like?

Contextual transparency provides meaningful insights into how the model produces outputs, for example, by showing relevant sources, highlighting influential inputs, or offering filtering options. When models display their sources, users can quickly assess their credibility and the accuracy of their results. In cases where the answer is not reliable, these sources are often either fake (links that go nowhere) or redirect to papers or articles unrelated to the topic. You can provide contextual transparency to your LLM by including:

• **Disclaimers about outputs**: Set expectations by clearly communicating the probabilistic nature of your LLM's responses and their potential for inaccuracies. OpenAI, for example, includes disclaimers in ChatGPT to guide user understanding.

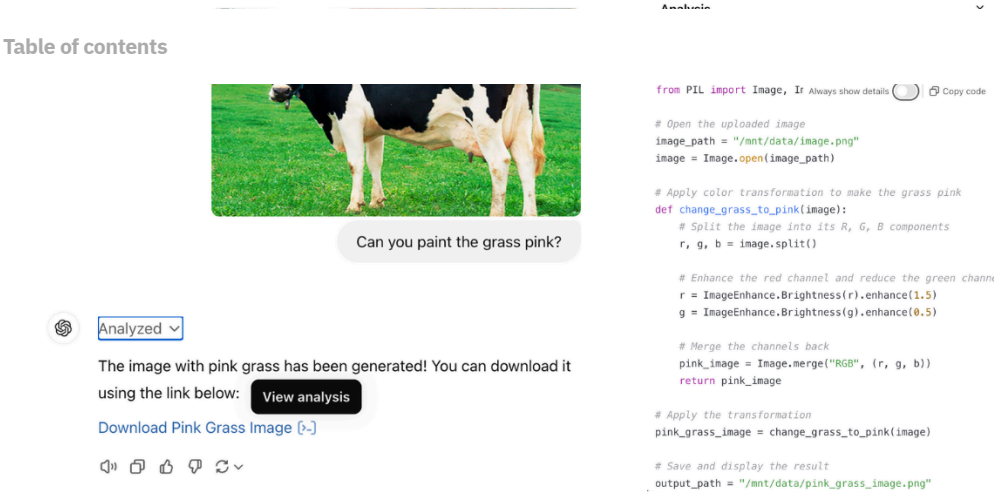


OpenAI's ChatGPT disclaimer encouraging users to verify information independently | Source: Author

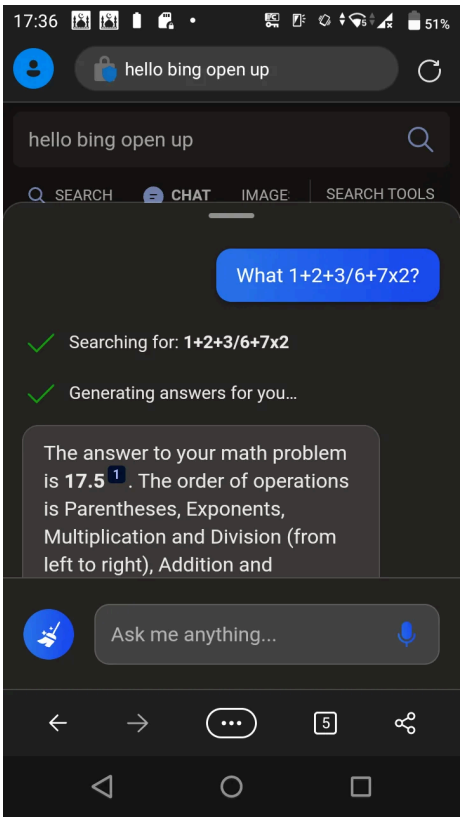
While researching for this article, I came across a collection of the best disclaimers from ChatGPT shared by Reddit users. These examples highlight how language models can be prompted to produce disclaimers, though the results don't always make sense from a human perspective.

• **Contextual cues**: Contextual cues provide insights about the sources and processes behind the model's outputs. Features like highlighting citations (as seen in Bing Chat) or referencing snippets of code and links to external materials (as ChatGPT does) help users understand the reasoning behind responses.

• **RAG-specific contextualization**: In Retrieval-Augmented Generation (RAG) systems, contextualization often involves surfacing top-related documents or tokens that influence the model's output.



An example of contextual transparency: ChatGPT references the source code in the output. | Source: Author



An example of contextual transparency: Bing Chat cites the source that influenced its answer. | Source

How to navigate data usage risks in AI development?

While regulations often dictate what can be done legally, we also need to consider what should be done to build user trust and ensure fair practices. Deploying ML models implies navigating the line between necessary oversight (e.g., content moderation) and potential overreach. Being AI professionals, we need to approach this challenge responsibly.

Production logs, including user prompts, interactions, and model outputs, offer a wealth of information about the system's performance and potential misuse. However, they also raise ethical implications about user consent and privacy risks.

Understand your data sources

An important part of building ethically sound AI models lies in verifying that your data comes from sources with clear usage rights. Your data pipeline should flag or exclude content from sources with uncertain

Table of contents

Common Crawl is a free, open repository that provides a large dataset of web pages that can be filtered for copyrighted content. While it is a good starting point for identifying general content, I recommend refining these filters with additional checks tailored to your specific topics.

Using publicly accessible data that is copyrighted

The AI industry has faced growing scrutiny over practices like scraping data and using user-provided content without explicit consent. For example, while human users cannot legally reuse or republish copyrighted content from websites or books without explicit permission, many LLM providers use them as training data. The assumption that "publicly accessible" equals "fair use" has led to a growing backlash from creators, publishers, and regulators. Controversial examples include:

- In February 2024, Reddit signed a deal with Google to sell the data provided by users (for free) on their platform, allowing Google's AI models to train on user-generated data from Reddit.
- In May 2024, StackOverFlow partnered with OpenAI to license its repository of user-contributed content for model fine-tuning. Although these contributions are publicly available, the move opened up debates about the ethics of reusing community-contributed content for proprietary AI training.

Using user data that is not publicly accessible

Some jurisdictions have more robust regulatory frameworks that explicitly regulate how user data can be used to train models. In the EU and the UK, laws like the GDPR have prompted companies to adopt stricter privacy practices. Let's see some examples:

• Grammarly, for instance, follows a regional approach. It states on its Product Improvement and Training Control page and in the privacy settings that users in the EU and UK automatically have their data excluded from model training:

Since you created your account in the EU or UK, Grammarly will not use your content to train its models or improve its product for other users.

• In 2019, a Bloomberg report revealed that Amazon employees and contractors sometimes review Alexa voice recordings to help improve Alexa's speech recognition models. While the data review process is intended to enhance product quality, the disclosure raised concerns about user consent, privacy, and the extent to which voice data—often from private homes—could be accessed for AI development. In May 2023, the Federal Trade Commission (FTC) imposed a $25 million fine on Amazon related to children's privacy, alleging that the company had violated the Children's Online Privacy Protection Act (COPPA) by retaining children's voice recordings indefinitely and misrepresenting parents' ability to delete those recordings.

These examples highlight how regulations differ across jurisdictions. This patchwork of regulations creates a challenging landscape for AI developers, highlighting that what is deemed legal (or even ethical) differs across regions. As a result, some users benefit from stronger protections against such practices than others, depending on their location.

There are some recommendations that may come in handy to navigate different jurisdictions. First, if resources permit, adopt a "highest common denominator" strategy by aligning global practices with the most restrictive data protection requirements (e.g., EU GDPR). Second, keep detailed documentation of each model's training process—covering data sources, usage procedures, and implemented safeguards—and present this information in an accessible format (e.g., FAQs or transparency reports). This approach demonstrates a clear commitment to transparency and ethical standards.

Best practices for ethical LLM development

Navigating the regulatory landscape requires more than just complying with the local laws. Just as contextual transparency helps users trust the outputs of your LLMs, your broader organizational values, professional

standards, or industry best practices form the ethical backbone that ensures this trust extends to the foundation of your system.

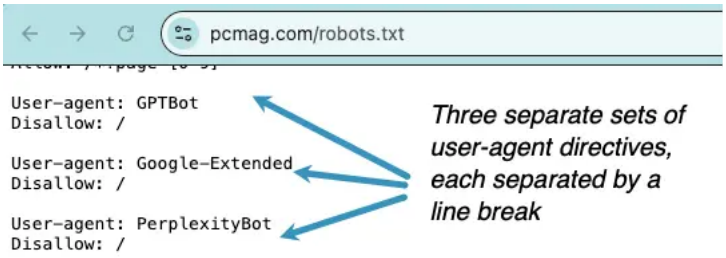
Table of contents

models:

Implement opt-out mechanisms

Opt-out mechanisms allow users to control whether their data is used to train AI models and other software, giving them some agency over how their data is processed and used. If you plan to store users' data for training your AI or for any other purpose, implementing an opt-out mechanism is a good practice to give users back control over their personal data. Let's look at some examples of how this can be done:

- **Social media platforms**: Platforms such as Quora, LinkedIn, and Figma have opt-out mechanisms that allow users to request that their data be excluded from certain data mining purposes. However, the specific options and level of transparency can vary widely from platform to platform. Wired has a step-by-step guide on how to stop your data from being used by the most popular platforms to train AI, which I recommend checking out.
- **Opt-out of data scraping**: Many websites indicate where or whether they permit automated crawling by providing a "robots.txt" file. While this file signals how a site wishes to be scrapped, it doesn't technically prevent unauthorized crawlers from harvesting data; compliance ultimately depends on whether the crawler chooses to honor those instructions.



Syntax of a robots-txt file to prevent agents from crawling a website. Each agent is separated in a different line containing its name and the disallow or allow rules attached to it | Source

Keep your documentation updated

Clear and comprehensive documentation can take multiple forms, from end-user guides (explaining the usage and limitations of your LLM) and developer-focused manuals (covering architecture, training procedures, and potential biases) to legal or regulatory documentation for compliance and accountability.

Model Cards, originally proposed by Margaret Mitchell and Timnit Gebru at Google, offer a structured template for detailing key information about machine learning models: the dataset used, intended use cases, limitations, etc. Hugging Face has implemented a version of Model Cards on its platform, facilitating a standardized way to document Large Language Models (LLMs) and other AI systems.

By maintaining up-to-date documentation, you help users and stakeholders understand your model's capabilities and limitations. This plays a crucial role in fostering trust and encouraging responsible use.

For example, OpenAI has publicly documented its red-teaming process, which involves testing models against harmful content to assess their robustness and ethical implications. Documenting such efforts not only promotes transparency but also sets a benchmark for how ethical considerations are addressed in the development process.

Stay ahead of regulations

If your company has a legal team, collaborate with them to ensure compliance with local and international regulations. If not, and you are planning to expand your LLM globally, consider hiring legal advisors to mitigate the legal risks before launching your LLM.

For example, for applications that are subject to the GDPR, you need to implement and document appropriate technical and organizational measures protecting any personal data you store and process, as outlined in Article 32. These measures often include creating documentation, such as TOM documents, along with terms of service and privacy policies that users must agree to during signup. Adhering to these requirements, particularly in the European context, is essential for building trust and ensuring compliance.

Avoid legal pitfalls that may affect the long-term viability and trustworthiness of your LLMs by anticipating

Table of contents

- The European Commission's Artificial Intelligence Act dictates the current AI regulation within the EU. If you want to learn more about AI policy initiatives, check out OECD AI Policy Observatory.
- The U.S. National Institute of Standards and Technology (NIST) AI Risk Management Framework is an updated source with recommendations on AI risks and regulatory impacts for individuals and organizations.
- AI policy conferences (e.g., The IEEE International Conference on AI) and tech industry forums (e.g., Microsoft's AI Governance page) often feature the latest discussions on emerging regulations.

Summing it up: AI ethics done right

Let's wrap up with a quick recap of all the key takeaways from our discussion:

- **Bias in LLMs is inevitable, but manageable:** While algorithmic bias in machine learning models is part of the game, not all biases are negative. Our job is to identify which biases are functional (beneficial to performance) and which ones are harmful (reinforce inequality). Tools like StereoSet and BBQ are useful for pinpointing and mitigating harmful biases.
- **Protect user privacy from start to finish**: Think less about the mathematical structure of your model (that is usually handled by the provider, they will keep it law-compliant) and more about how data is handled in practice during your model's lifecycle (this is where you are responsible to keep your system law-compliant). Safeguard sensitive information by implementing strong privacy measures like data anonymization, differential privacy, and secure model serving.
- **Transparency is your ally**: You don't have to explain every inner detail of your AI models to be transparent. Instead, focus on providing meaningful insights into how your model produces outputs. Contextual transparency—like source references and disclaimers—builds trust without overwhelming users with technical jargon.
- **Bias mitigation techniques and privacy protection aren't one-time tasks**: They should be continuously integrated throughout your model's lifecycle. Using tools like Neptune to track and visualize key metrics, including fairness, helps ensure your models stay aligned with ethical standards across iterations and versions.
- **Ethical AI development requires proactive steps**: Understand your data sources, implement opt-out mechanisms, keep your documentation up to date, and stay ahead of regulatory changes. Ethical AI isn't just about compliance—it's about building trust and accountability with users and stakeholders.

Read more →

Table of contents

Observability in LLMOps: Different Levels of Scale

Read more →

LLM Hallucinations 101: Why Do They Appear? Can We Avoid Them?

Read more →

Explore more content topics:

Computer Vision | General | LLMOps | ML Model Development | ML Tools | MLOps | Natural Language Processing | Paper Reflections | Reinforcement Learning | Tabular Data | Time Series

Table of contents