# Tạo nội dung phù hợp với Amazon Bedrock và ConstitutionalChain

by Sriharsh Adari, Chris Cocking, David Min, and Stephen Garth on 01 APR 2025 in [Advanced (300)](https://aws.amazon.com/blogs/machine-learning/category/learning-levels/advanced-300/), [Amazon Bedrock](https://aws.amazon.com/blogs/machine-learning/category/artificial-intelligence/amazon-machine-learning/amazon-bedrock/), [Amazon Bedrock Guardrails](https://aws.amazon.com/blogs/machine-learning/category/artificial-intelligence/amazon-machine-learning/amazon-bedrock/amazon-bedrock-guardrails/), [Amazon Bedrock Knowledge Bases](https://aws.amazon.com/blogs/machine-learning/category/artificial-intelligence/amazon-machine-learning/amazon-bedrock/amazon-bedrock-knowledge-bases/), [Amazon SageMaker](https://aws.amazon.com/blogs/machine-learning/category/artificial-intelligence/sagemaker/), [Artificial Intelligence](https://aws.amazon.com/blogs/machine-learning/category/artificial-intelligence/) [Permalink](https://aws.amazon.com/blogs/machine-learning/generate-compliant-content-with-amazon-bedrock-and-constitutionalchain/)  [Comments](https://aws.amazon.com/blogs/machine-learning/generate-compliant-content-with-amazon-bedrock-and-constitutionalchain/#Comments)  [Share](https://aws.amazon.com/vi/blogs/machine-learning/generate-compliant-content-with-amazon-bedrock-and-constitutionalchain/#)

**Generative AI** đã nổi lên như một công cụ mạnh mẽ cho việc tạo nội dung, mang lại một số lợi ích chính có thể tăng cường hiệu quả và hiệu suất của quá trình sản xuất nội dung như tạo tài liệu tiếp thị, tạo hình ảnh, quản lý nội dung, v.v. **Constitutional AI** và cơ chế phản xạ của **LangGraph** là hai cách tiếp cận bổ sung để đảm bảo các hệ thống AI hoạt động một cách đạo đức – với Anthropic nhúng các nguyên tắc trong quá trình đào tạo trong khi LangGraph thực hiện chúng trong quá trình suy luận/thực thi thông qua các cơ chế phản xạ và tự sửa chữa. Bằng cách sử dụng Constitutional AI của LangGraph, các nhà tạo nội dung có thể tối ưu hóa quy trình làm việc của họ trong khi duy trì các tiêu chuẩn cao về tuân thủ và đạo đức do người dùng xác định. Phương pháp này không chỉ giảm thiểu nhu cầu giám sát của con người mà còn tăng cường sự minh bạch và trách nhiệm trong quá trình tạo nội dung bởi AI.

Trong bài viết này, chúng tôi khám phá các chiến lược thực tế để sử dụng Constitutional AI để tạo nội dung tuân thủ một cách hiệu quả và hiệu suất bằng cách sử dụng [Amazon Bedrock](https://aws.amazon.com/bedrock/?gclid=Cj0KCQiA0MG5BhD1ARIsAEcZtwQKTF2XajiMI01XQdaYxrazrkiahrOlBCzmCS5Zdwa4X3426oOmQaIaAiN6EALw_wcB&trk=0eaabb80-ee46-4e73-94ae-368ffb759b62&sc_channel=ps&ef_id=Cj0KCQiA0MG5BhD1ARIsAEcZtwQKTF2XajiMI01XQdaYxrazrkiahrOlBCzmCS5Zdwa4X3426oOmQaIaAiN6EALw_wcB:G:s&s_kwcid=AL!4422!3!692006004688!p!!g!!amazon%20bedrock!21048268554!159639952935) và **LangGraph** để xây dựng **ConstitutionalChain** cho việc tạo nội dung nhanh chóng trong các ngành được quy định nghiêm ngặt như tài chính và y tế. Mặc dù AI mang lại lợi ích sản phẩm cao, nhưng việc duy trì tuân thủ các quy định nghiêm ngặt là rất quan trọng. Xác minh thủ công nội dung do AI tạo ra để tuân thủ quy định có thể tốn thời gian và thách thức. Chúng tôi cũng cung cấp một tổng quan về cách Insagic, một công ty của Publicis Groupe, tích hợp khái niệm này vào quy trình tiếp thị y tế hiện có của họ bằng cách sử dụng Amazon Bedrock. Insagic là một doanh nghiệp tư vấn và tầm nhìn thế hệ tiếp theo kết hợp dữ liệu, thiết kế và đối thoại để cung cấp các tầm nhìn hành động và tính linh hoạt cho các nhà tiếp thị y tế. Nó sử dụng chuyên môn từ các nhà khoa học dữ liệu, nhà khoa học hành vi và các nhà chiến lược để đạt được kết quả tốt hơn trong ngành y tế.

### Hiểu về Constitutional AI

Constitutional AI được thiết kế để căn chỉnh các mô hình ngôn ngữ lớn (LLMs) với các giá trị con người và các cân nhắc đạo đức. Nó hoạt động bằng cách tích hợp một tập hợp các quy tắc, nguyên tắc và ràng buộc đã xác định trước vào kiến trúc cốt lõi và quá trình đào tạo của LLM. Cách tiếp cận này đảm bảo rằng LLM hoạt động trong các thông số đạo đức và pháp lý đã xác định, tương tự như cách một hiến pháp quản lý các luật và hành động của một quốc gia.

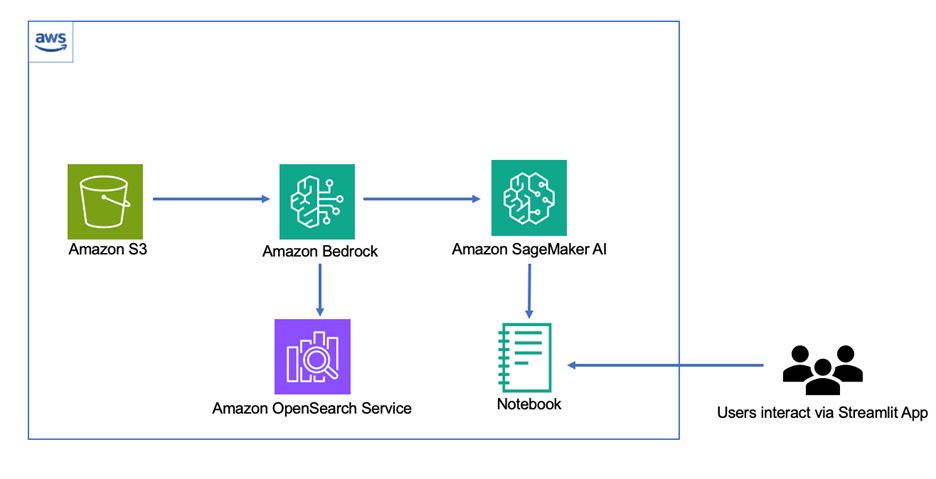
Các lợi ích chính của Constitutional AI cho việc tạo nội dung bao gồm:

* **Căn chỉnh đạo đức** – Nội dung được tạo ra bằng Constitutional AI về cơ bản được căn chỉnh với các tiêu chuẩn đạo đức đã xác định trước
* **Tuân thủ pháp lý** – LLM được thiết kế để hoạt động trong các khuôn khổ pháp lý, giảm thiểu rủi ro tạo ra nội dung không tuân thủ
* **Minh bạch** – Các nguyên tắc hướng dẫn quá trình ra quyết định của LLM được xác định rõ ràng và có thể được kiểm tra
* **Giảm giám sát của con người** – Bằng cách nhúng các hướng dẫn đạo đức vào LLM, nhu cầu xem xét rộng rãi của con người được giảm thiểu đáng kể

Hãy khám phá cách bạn có thể khai thác sức mạnh của Constitutional AI để tạo nội dung tuân thủ cho tổ chức của mình.

### Tổng quan giải pháp

Đối với giải pháp này, chúng tôi sử dụng [Amazon Bedrock Knowledge Bases](https://aws.amazon.com/bedrock/knowledge-bases/) để lưu trữ một kho lưu trữ các tài liệu y tế. Chúng tôi sử dụng cách tiếp cận **Retrieval Augmented Generation (RAG)** (Tạo thông tin bổ sung dựa trên lấy thông tin), lấy ngữ cảnh liên quan trước và tổng hợp câu trả lời dựa trên ngữ cảnh đã lấy để tạo các bài báo dựa trên kho lưu trữ. Sau đó, chúng tôi sử dụng khung làm việc tự do [LangGraph](https://github.com/langchain-ai/langgraph) và ConstitutionalChain để tạo, phê bình và xem xét các đề xuất trong một notebook [Amazon SageMaker](https://aws.amazon.com/sagemaker/?gclid=Cj0KCQiAqL28BhCrARIsACYJvkdPHesrBiOisT6BBceqYfKXcHU5ThawqO2pbk8--KqQORuGMN9ANEgaAtD8EALw_wcB&trk=8987dd52-6f33-407a-b89b-a7ba025c913c&sc_channel=ps&ef_id=Cj0KCQiAqL28BhCrARIsACYJvkdPHesrBiOisT6BBceqYfKXcHU5ThawqO2pbk8--KqQORuGMN9ANEgaAtD8EALw_wcB:G:s&s_kwcid=AL!4422!3!532502995192!e!!g!!amazon%20sagemaker!11206038603!113600690367) và phát triển một quy trình làm việc để tạo nội dung tuân thủ. Sơ đồ sau minh họa kiến trúc này.



Cách thực hiện này chứng minh một quy trình làm việc phức tạp không chỉ tạo ra các phản hồi dựa trên kho kiến thức mà còn sử dụng một kỹ thuật phản xạ để kiểm tra các đầu ra của nó thông qua các nguyên tắc đạo đức, cho phép nó tinh chỉnh và cải thiện các đầu ra của nó. Chúng tôi tải lên một tập hợp mẫu các tài liệu sức khỏe tâm thần lên Amazon Bedrock Knowledge Bases và sử dụng các tài liệu đó để viết một bài báo về sức khỏe tâm thần bằng cách sử dụng cách tiếp cận dựa trên RAG. Sau đó, chúng tôi xác định một nguyên tắc hiến pháp với một nguyên tắc tính đa dạng, công bằng và đồng đều (DEI) tùy chỉnh, quy định cách phê bình và sửa đổi các phản hồi để đảm bảo tính đa dạng.

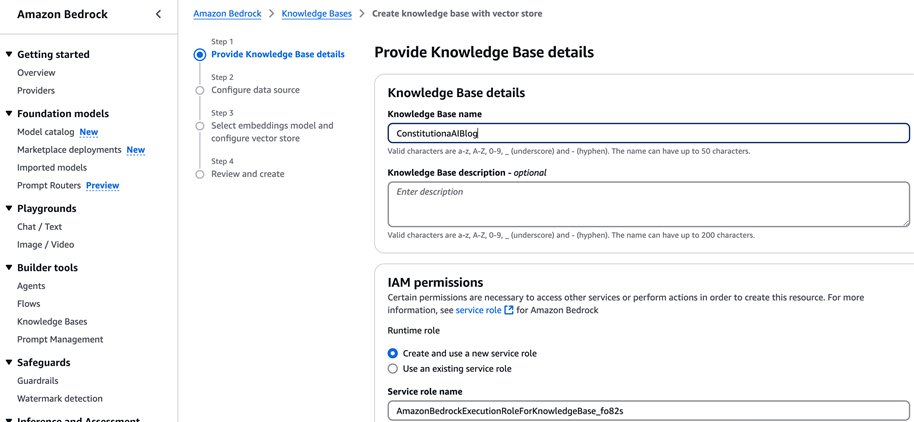
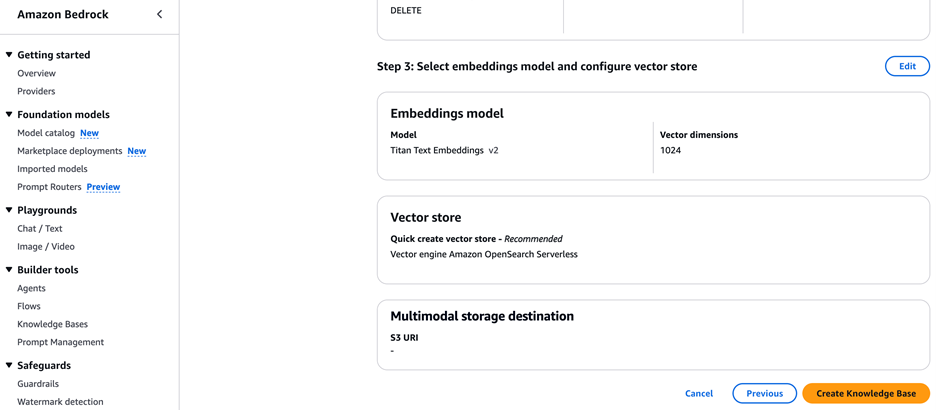
### Yêu cầu đầu vào

Để triển khai giải pháp, bạn cần các yêu cầu đầu vào sau:

* Một tài khoản **AWS**
* Quyền [AWS Identity and Access Management](https://aws.amazon.com/iam/) (IAM) thích hợp để truy cập vào một thùng chứa [Amazon Simple Storage Service](https://aws.amazon.com/s3/) (Amazon S3), tạo các kho kiến thức Amazon Bedrock và tạo một phiên bản notebook SageMaker

### Tạo một kho kiến thức Amazon Bedrock

Để trình diễn khả năng này, chúng tôi tải xuống một bài báo về sức khỏe tâm thần từ [repo GitHub](https://github.com/aws-samples/Constitutional-Chain-Demo) này và lưu trữ nó trong Amazon S3. Sau đó, chúng tôi sử dụng Amazon Bedrock Knowledge Bases để chỉ mục các bài báo. Theo mặc định, Amazon Bedrock sử dụng [Amazon OpenSearch Serverless](https://aws.amazon.com/opensearch-service/features/serverless/) làm cơ sở dữ liệu vector. Để biết hướng dẫn đầy đủ để tạo một kho kiến thức trong Amazon Bedrock Knowledge Bases với Amazon S3 làm nguồn dữ liệu, xem [Tạo một kho kiến thức trong Amazon Bedrock Knowledge Bases](https://docs.aws.amazon.com/bedrock/latest/userguide/knowledge-base-create.html).

* 1. Trên bảng điều khiển Amazon Bedrock, tạo một kho kiến thức mới.
  2. Cung cấp tên cho kho kiến thức của bạn và tạo một vai trò dịch vụ IAM mới.
  3. Chọn Amazon S3 làm nguồn dữ liệu và cung cấp thùng chứa S3 lưu trữ bài báo về sức khỏe tâm thần.
  4. Chọn Amazon Titan Text Embeddings v2 làm mô hình nhúng và OpenSearch Serverless làm cửa hàng vector.
  5. Bấm vào Tạo Kho kiến thức.

### Câu lệnh nhập và thiết lập một khách hàng Amazon Bedrock

Làm theo hướng dẫn được cung cấp trong tệp README trong [repo GitHub](https://github.com/aws-samples/Constitutional-Chain-Demo). Sao chép repo GitHub để tạo một bản sao địa phương. Chúng tôi khuyên bạn nên chạy mã này trong một môi trường [SageMaker JupyterLab](https://docs.aws.amazon.com/sagemaker/latest/dg/studio-updated-jl.html). Mã sau đây nhập các thư viện cần thiết, bao gồm Boto3 cho các dịch vụ AWS, các thành phần LangChain và Streamlit. Nó thiết lập một khách hàng Amazon Bedrock và cấu hình mô hình Anthropic’s Claude 3 Haiku với các thông số cụ thể.

import boto3

from langchain\_aws import ChatBedrock

…

bedrock\_runtime = boto3.client(service\_name="bedrock-runtime", region\_name="us-east-1")

llm = ChatBedrock(client=bedrock\_runtime, model\_id="anthropic.claude-3-haiku-20240307-v1:0")

### Định nghĩa các thành phần Constitutional AI

Tiếp theo, chúng tôi định nghĩa một lớp Critique để cấu trúc đầu ra của quá trình phê bình. Sau đó, chúng tôi tạo các mẫu đề xuất cho phê bình và sửa đổi. Cuối cùng, chúng tôi thiết lập các chuỗi sử dụng LangChain để tạo các phản hồi, phê bình và sửa đổi.

# LangChain Constitutional chain migration to LangGraph

class Critique(TypedDict):

"""Generate a critique, if needed."""

critique\_needed: Annotated[bool, ..., "Whether or not a critique is needed."]

critique: Annotated[str, ..., "If needed, the critique."]

critique\_prompt = ChatPromptTemplate.from\_template(

"Critique this response according to the critique request. "

…

)

revision\_prompt = ChatPromptTemplate.from\_template(

"Revise this response according to the critique and reivsion request.\n\n"

….

)

chain = llm | StrOutputParser()

critique\_chain = critique\_prompt | llm.with\_structured\_output(Critique)

revision\_chain = revision\_prompt | llm | StrOutputParser()

### Định nghĩa một lớp State và tham khảo trở lại Amazon Bedrock Knowledge Bases retriever

Chúng tôi định nghĩa một lớp State LangGraph để quản lý trạng thái cuộc hội thoại, bao gồm câu hỏi, các nguyên tắc hiến pháp, các phản hồi và các phê bình:

# LangGraph State

class State(TypedDict):

query: str

constitutional\_principles: List[ConstitutionalPrinciple]

### Thiết lập một Amazon Bedrock Knowledge Bases retriever để trích xuất thông tin liên quan

Tiếp theo, chúng tôi thiết lập một Amazon Bedrock Knowledge Bases retriever để trích xuất thông tin liên quan. Chúng tôi tham khảo kho kiến thức Amazon Bedrock mà chúng tôi đã tạo trước đó để tạo một bài báo dựa trên các tài liệu về sức khỏe tâm thần. Hãy chắc chắn rằng bạn cập nhật ID kho kiến thức trong mã sau đây với ID kho kiến thức bạn đã tạo trong các bước trước:

#-----------------------------------------------------------------

# Amazon Bedrock KnowledgeBase

from langchain\_aws.retrievers import AmazonKnowledgeBasesRetriever

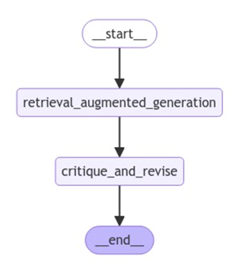
retriever = AmazonKnowledgeBasesRetriever(

knowledge\_base\_id="W3NMIJXLUE", # Thay đổi thành ID Kho kiến thức của bạn

)

## Tạo các nút LangGraph và đồ thị LangGraph cùng với các nguyên tắc hiến pháp

Phần mã tiếp theo tích hợp tổng quát hóa quy trình làm việc dựa trên đồ thị, các nguyên tắc đạo đức, và một giao diện thân thiện với người dùng để tạo ra một mô hình AI hiến pháp phức tạp. Sơ đồ dưới đây minh họa quy trình làm việc.



Nó sử dụng StateGraph để quản lý dòng chảy giữa RAG và các nút phê bình/sửa đổi, kết hợp một nguyên tắc DEI tùy chỉnh để hướng dẫn các phản hồi của LLM. Hệ thống được trình bày thông qua ứng dụng Streamlit, cung cấp một giao diện trò chuyện tương tác nơi người dùng có thể nhập các truy vấn và xem các phản hồi ban đầu của LLM, phê bình, và câu trả lời đã sửa đổi. Ứng dụng cũng có một thanh bên hiển thị một hình ảnh đồ thị của quy trình làm việc và một mô tả về nguyên tắc đạo đức được áp dụng. Cách tiếp cận toàn diện này đảm bảo rằng các đầu ra của LLM không chỉ dựa trên kiến thức mà còn được căn chỉnh đạo đức bằng cách sử dụng các nguyên tắc hiến pháp tùy chỉnh để hướng dẫn một quá trình phản chiếu (phê bình và sửa đổi), tất cả trong khi duy trì một trải nghiệm thân thiện với người dùng với các tính năng như quản lý lịch sử trò chuyện và tùy chọn trò chuyện rõ ràng.

## Ứng dụng Streamlit

Phần ứng dụng Streamlit của mã này tạo ra một giao diện tương tác và thân thiện với người dùng cho mô hình AI hiến pháp. Nó thiết lập một khung bên hiển thị một hình ảnh đồ thị của đồ thị quy trình làm việc của LLM và cung cấp một mô tả về nguyên tắc DEI đang được áp dụng. Giao diện chính có một phần trò chuyện nơi người dùng có thể nhập các truy vấn của họ và xem các phản hồi của LLM.

# ------------------------------------------------------------------------

# Ứng dụng Streamlit

# Hàm xóa lịch sử trò chuyện

def clear\_screen():

st.session\_state.messages = [{"role": "assistant", "content": "Tôi có thể giúp gì cho bạn hôm nay?}]

with st.sidebar:

st.subheader('Trình diễn AI Hiến pháp')

…..

ConstitutionalPrinciple(

name="Nguyên tắc DEI",

critique\_request="Phân tích nội dung để tìm bất kỳ sự thiếu đa dạng, công bằng, hoặc bao gồm nào. Xác định các trường hợp cụ thể nơi nội dung có thể bao gồm nhiều hơn hoặc đại diện cho các góc nhìn đa dạng hơn.",

revision\_request="Viết lại nội dung bằng cách kết hợp các phê bình để làm cho nội dung đa dạng, công bằng, và bao gồm hơn. Đảm bảo đại diện cho các góc nhìn khác nhau và sử dụng ngôn ngữ bao gồm trong suốt."

)

""")

st.button('Xóa Màn Hình', on\_click=clear\_screen)

# Lưu các phản hồi được tạo bởi LLM

if "messages" not in st.session\_state.keys():

st.session\_state.messages = [{"role": "assistant", "content": "Tôi có thể giúp gì cho bạn hôm nay?}]

# Đầu vào trò chuyện - Lời nhắc của người dùng

if prompt := st.chat\_input():

….

with st.spinner(f"Đang tạo..."):

….

with st.chat\_message("assistant"):

st.markdown("\*\*[phản hồi ban đầu]\*\*")

….

st.session\_state.messages.append({"role": "assistant", "content": "[phản hồi đã sửa đổi] " + generation['response']})

Python

Ứng dụng duy trì một lịch sử trò chuyện, hiển thị cả đầu vào của người dùng và các phản hồi của LLM, bao gồm phản hồi ban đầu, bất kỳ phê bình nào được tạo ra, và phản hồi đã sửa đổi cuối cùng. Mỗi bước của quá trình LLM được gắn thẻ và trình bày rõ ràng cho người dùng. Giao diện cũng bao gồm một nút **Xóa Màn Hình** để thiết lập lại lịch sử trò chuyện. Khi xử lý một truy vấn, ứng dụng hiển thị một biểu trưng tải và hiển thị thời gian chạy, cung cấp sự minh bạch về hoạt động của LLM. Thiết kế giao diện người dùng toàn diện này cho phép người dùng tương tác với LLM trong khi quan sát cách các nguyên tắc hiến pháp được áp dụng để tinh chỉnh các đầu ra của LLM.

## Kiểm tra giải pháp bằng giao diện người dùng Streamlit

Trong ứng dụng Streamlit, khi một người dùng nhập một truy vấn, ứng dụng khởi động quá trình bằng cách tạo và biên dịch đồ thị được định nghĩa trước đó. Sau đó, nó truyền phát việc thực thi của đồ thị này, bao gồm các bước RAG và phê bình/sửa đổi. Trong quá trình này, ứng dụng hiển thị các cập nhật thời gian thực cho từng nút đang thực thi, cho phép người dùng quan sát những gì đang xảy ra ở phía sau. Hệ thống đo lường thời gian chạy tổng, cung cấp sự minh bạch về thời gian xử lý. Khi hoàn tất, ứng dụng trình bày kết quả một cách có cấu trúc trong giao diện trò chuyện. Nó hiển thị phản hồi ban đầu được tạo bởi LLM, tiếp theo là bất kỳ phê bình nào được thực hiện dựa trên các nguyên tắc hiến pháp, và cuối cùng hiển thị phản hồi đã sửa đổi kết hợp các cân nhắc đạo đức này. Bước trình bày từng bước này cho phép người dùng quan sát cách phản hồi của LLM phát triển qua quá trình AI hiến pháp, từ sinh vật ban đầu đến tinh chỉnh đạo đức. Như đã đề cập, trong tệp README trên GitHub [README](https://github.com/aws-samples/Constitutional-Chain-Demo/blob/main/README.md), để chạy ứng dụng Streamlit, sử dụng mã sau:

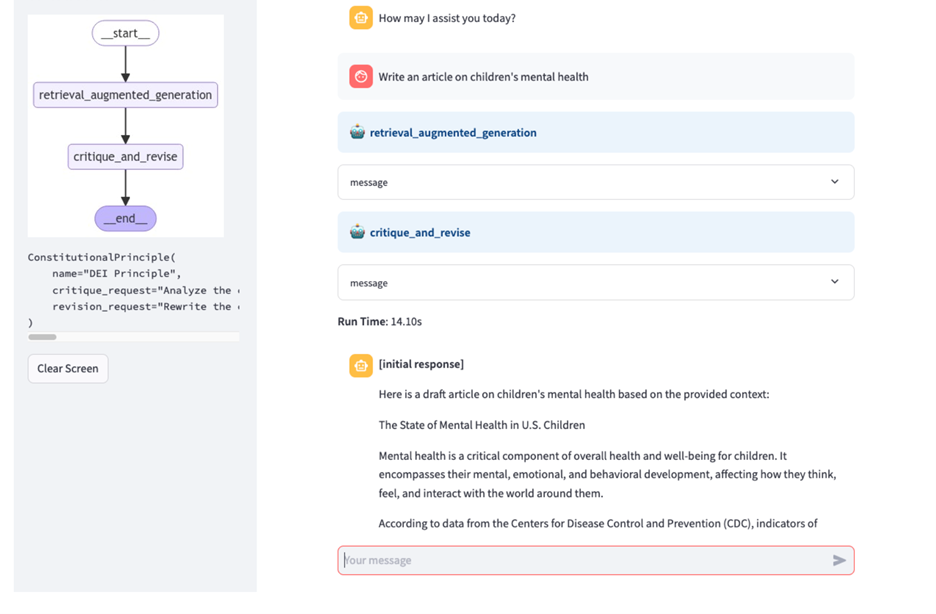
pip install -r requirements.txt

streamlit run main.py

Python

Để biết chi tiết về việc sử dụng proxy Jupyter để truy cập vào ứng dụng Streamlit, hãy tham khảo [Xây dựng ứng dụng Streamlit trong Amazon SageMaker Studio](https://aws.amazon.com/blogs/machine-learning/build-streamlit-apps-in-amazon-sagemaker-studio/).

Sửa đổi URL Studio, thay thế lab? bằng proxy/8501/.



## Cách Insagic sử dụng AI hiến pháp để tạo nội dung tuân thủ

[Insagic](https://publicishealth.com/insagic) sử dụng dữ liệu y tế thực tế để giúp các thương hiệu hiểu người như là bệnh nhân và bệnh nhân như là người, cho phép họ cung cấp các hiểu biết có tác dụng trong không gian tiếp thị y tế. Mặc dù việc tạo ra các hiểu biết sâu trong lĩnh vực y tế có thể mang lại những lợi ích to lớn, nhưng nó phải được thực hiện với sự cân nhắc về tuân thủ và bản chất cá nhân của dữ liệu y tế. Bằng cách xác định các hướng dẫn liên bang như là các nguyên tắc hiến pháp, Insagic đảm bảo rằng nội dung được tạo ra bởi AI sinh vật tuân thủ các hướng dẫn liên bang cho tiếp thị y tế.

## Làm sạch

Khi bạn đã hoàn thành việc thử nghiệm với giải pháp này, hãy làm sạch tài nguyên của mình để ngăn không cho các phí AWS được tính:

1. Xóa trống các thùng chứa S3.
2. Xóa bản sao notebook SageMaker.
3. Xóa cơ sở kiến thức Amazon Bedrock.

## Kết luận

Bài viết này đã chứng minh cách triển khai một giải pháp AI sinh vật phức tạp sử dụng Amazon Bedrock và LangGraph để tạo nội dung tuân thủ. Bạn cũng có thể tích hợp quy trình này để tạo các phản hồi dựa trên cơ sở kiến thức và áp dụng các nguyên tắc đạo đức để phê bình và sửa đổi các đầu ra của nó, tất cả trong một giao diện web tương tác. Insagic đang tìm kiếm nhiều cách để tích hợp điều này vào các quy trình hiện có bằng cách xác định các nguyên tắc tùy chỉnh để đạt được các mục tiêu tuân thủ.

Bạn có thể mở rộng khái niệm này thêm bằng cách tích hợp [Amazon Bedrock Guardrails](https://aws.amazon.com/bedrock/guardrails/). Amazon Bedrock Guardrails và LangGraph AI hiến pháp có thể tạo ra một hệ thống bảo mật toàn diện bằng cách hoạt động ở các cấp độ khác nhau. Amazon Bedrock cung cấp bộ lọc nội dung và ranh giới an toàn cấp API, và LangGraph thực hiện các nguyên tắc hiến pháp trong các quy trình suy luận. Cùng với nhau, chúng tập hợp bảo vệ đa lớp thông qua bộ lọc I/O, hạn chế chủ đề, ràng buộc đạo đức, và các bước xác minh logic trong các ứng dụng AI.

Hãy thử giải pháp cho trường hợp sử dụng của riêng bạn, và để lại phản hồi của bạn trong các bình luận.

### Giới thiệu về các tác giả

 **Sriharsh Adari** là một Kỹ sư Giải Pháp Cấp Cao tại Amazon Web Services (AWS), nơi ông giúp khách hàng làm việc ngược lại từ kết quả kinh doanh để phát triển các giải pháp sáng tạo trên AWS. Trong nhiều năm, ông đã giúp đỡ nhiều khách hàng trong các biến đổi nền tảng dữ liệu trên khắp các lĩnh vực công nghiệp. Lĩnh vực chuyên môn cốt lõi của ông bao gồm Chiến lược Công nghệ, Phân tích Dữ liệu, và Khoa học Dữ liệu. Trong thời gian rảnh rỗi, ông thích chơi thể thao, xem phim và chơi Tabla.

**David Min** là một Kỹ sư Giải Pháp Bán Hàng Cấp Cao tại Amazon Web Services (AWS) chuyên về AI sinh vật, nơi ông giúp khách hàng chuyển đổi doanh nghiệp của họ thông qua các giải pháp AI sáng tạo. Trong suốt sự nghiệp, David đã giúp đỡ nhiều tổ chức trên khắp các ngành công nghiệp vượt qua khoảng cách giữa công nghệ AI tiên tiến và ứng dụng kinh doanh thực tế, tập trung vào sự tham gia của điều hành và sự chấp nhận giải pháp thành công.

**Stephen Garth** là một Nhà Khoa học Dữ liệu tại Insagic, nơi ông phát triển các giải pháp học máy tiên tiến, bao gồm các công cụ tự động hóa được cung cấp bởi LLM và các mô hình phân cụm sâu cho các hiểu biết hành động, sâu sắc cho người tiêu dùng. Với nền tảng mạnh mẽ bao gồm kỹ thuật phần mềm, khoa học dữ liệu y tế, và nghiên cứu tính toán, ông đam mê mang lại chuyên môn của mình trong phân tích dẫn động bởi AI và xử lý dữ liệu quy mô lớn để đưa ra các giải pháp.

**Chris Cocking** chuyên về thiết kế ứng dụng doanh nghiệp có khả năng mở rộng bằng sử dụng nhiều ngôn ngữ lập trình. Với gần 20 năm kinh nghiệm, ông xuất sắc trong các môi trường LAMP và IIS, chiến lược SEO, và gần đây nhất là thiết kế các hệ thống đại lý. Ngoài công việc, Chris là một nhạc sĩ bass và yêu thích âm nhạc, điều này giúp thúc đẩy sự sáng tạo và kỹ n