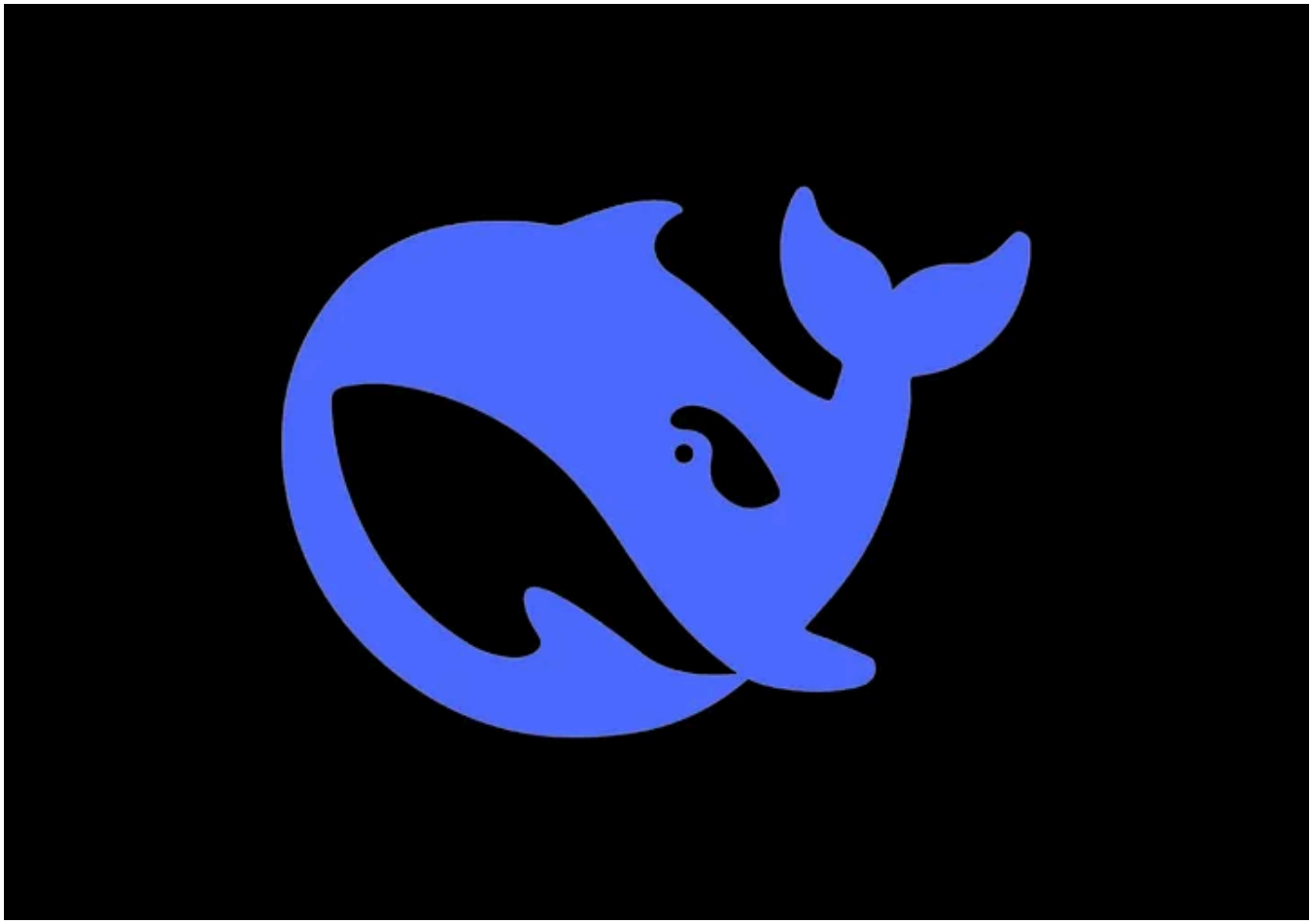
DeepSeek R1

🚀 Unlocking the Power of DeepSeek R1 Models: A Comprehensive Guide for Beginners and Experts 🚀

Naveen Krishnan
Published in Generative AI
8 min read · Feb 3, 2025



Artificial Intelligence (AI) is revolutionizing industries, and the **DeepSeek R1 models** are at the forefront of this transformation. Whether you're a beginner

looking to dive into the world of AI or an expert seeking to optimize your workflows, this guide will walk you through everything you need to know about DeepSeek R1 models. We'll cover hosting the model both locally and on Azure using the Model Catalog, connecting via API keys, and even how these models can accelerate AI agent implementations. Let's get started! 🎉

📖 Table of Contents

1. Introduction to DeepSeek R1 Models 🤖

DeepSeek R1 models are state-of-the-art AI models designed for a wide range of applications, from natural language processing (NLP) to computer vision. These models are pre-trained on vast datasets, making them highly versatile and capable of handling complex tasks with minimal fine-tuning.

Key Features:

- **High Accuracy**: Trained on diverse datasets, ensuring robust performance.
- **Scalability**: Can be deployed on cloud platforms like Azure for seamless scaling.

- **Ease of Integration:** Simple API-based integration for quick deployment.

2. Why DeepSeek R1 Models? 🌟

DeepSeek R1 models stand out due to their:

- **Versatility**: Suitable for various industries, including healthcare, finance, and e-commerce.
- **Efficiency**: Optimized for fast inference, reducing latency.
- **Customizability**: Easily fine-tuned to meet specific business needs.

3. Running DeepSeek Model Locally

Step 1**: Install Ollama**

Visit Ollama's download page and download the installer for your operating system.

Follow the on-screen instructions to complete the installation.

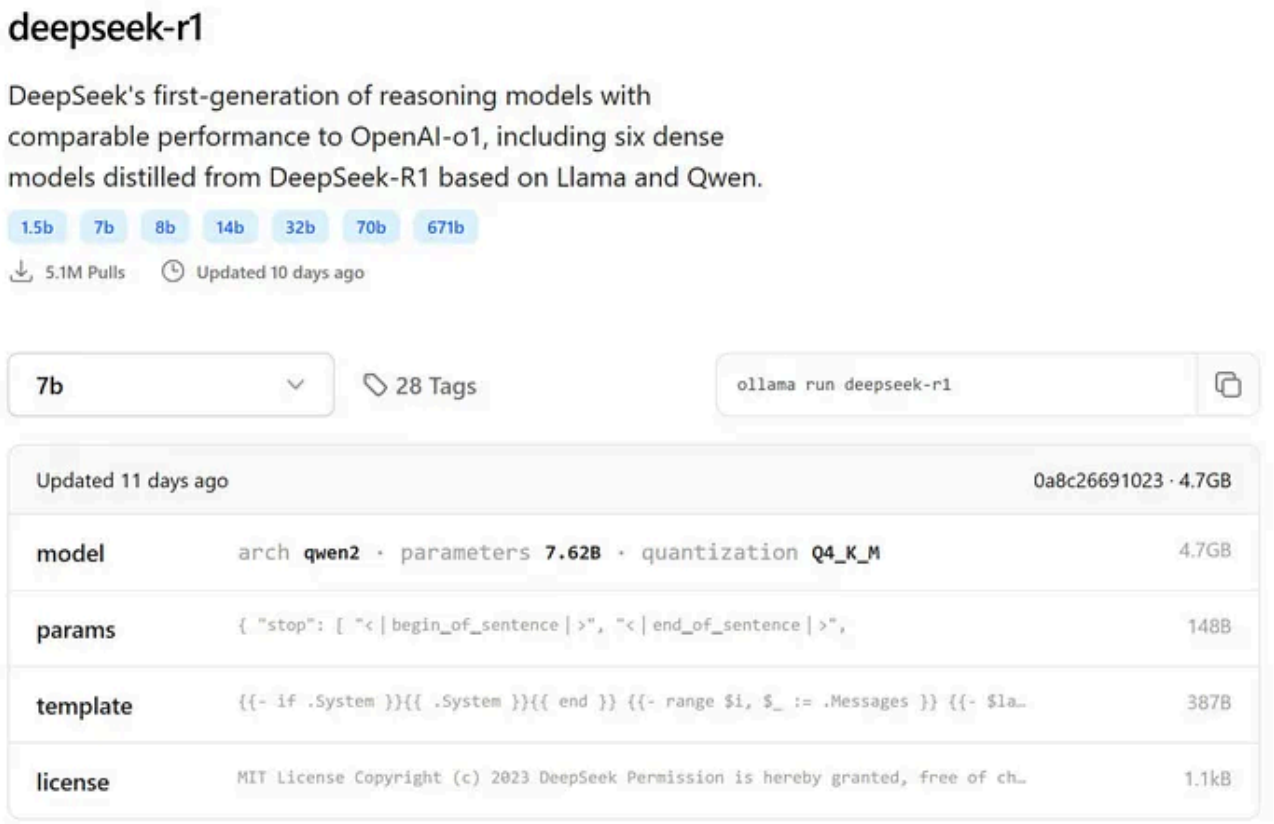
Once installed, open the terminal (or command prompt) and verify the installation by typing:

```
ollama --version
```

You should see the version number if the installation was successful.

Step 2**: Download and Set Up DeepSeek-R1**

Open the Ollama app or use the command line to search for the DeepSeek-R1 model. There are different versions of DeepSeek-R1. Please see this page for versions. In this example, I am installing the 32b version.




```
ollama install deepseek-r1:32b
```

Wait for the model to download and install. This process may take some time depending on your internet speed and system performance.

Verify the installation by running:

```
ollama list
```

DeepSeek-R1 should appear in the list of installed models.

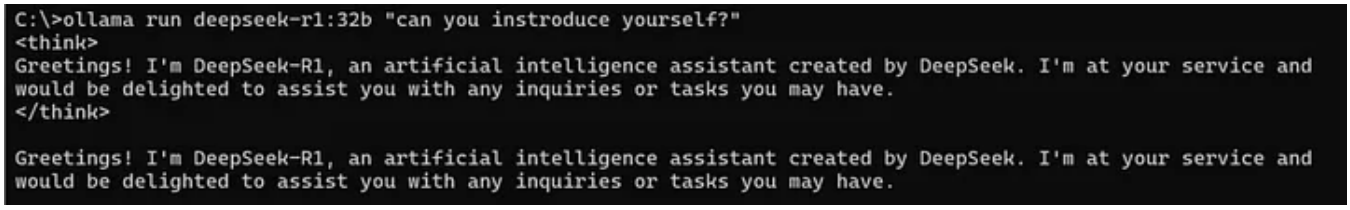
Step 3**: Run DeepSeek-R1**

Start the Ollama runtime by opening a terminal and typing:

```
ollama start
```

Once the runtime is active, run DeepSeek-R1 by entering:

```
ollama run deepseek-r1:32b
```



The model will process the input and return results directly in the terminal or your connected application.

4. Hosting DeepSeek R1 Models on Azure Using Model Catalog ☁️

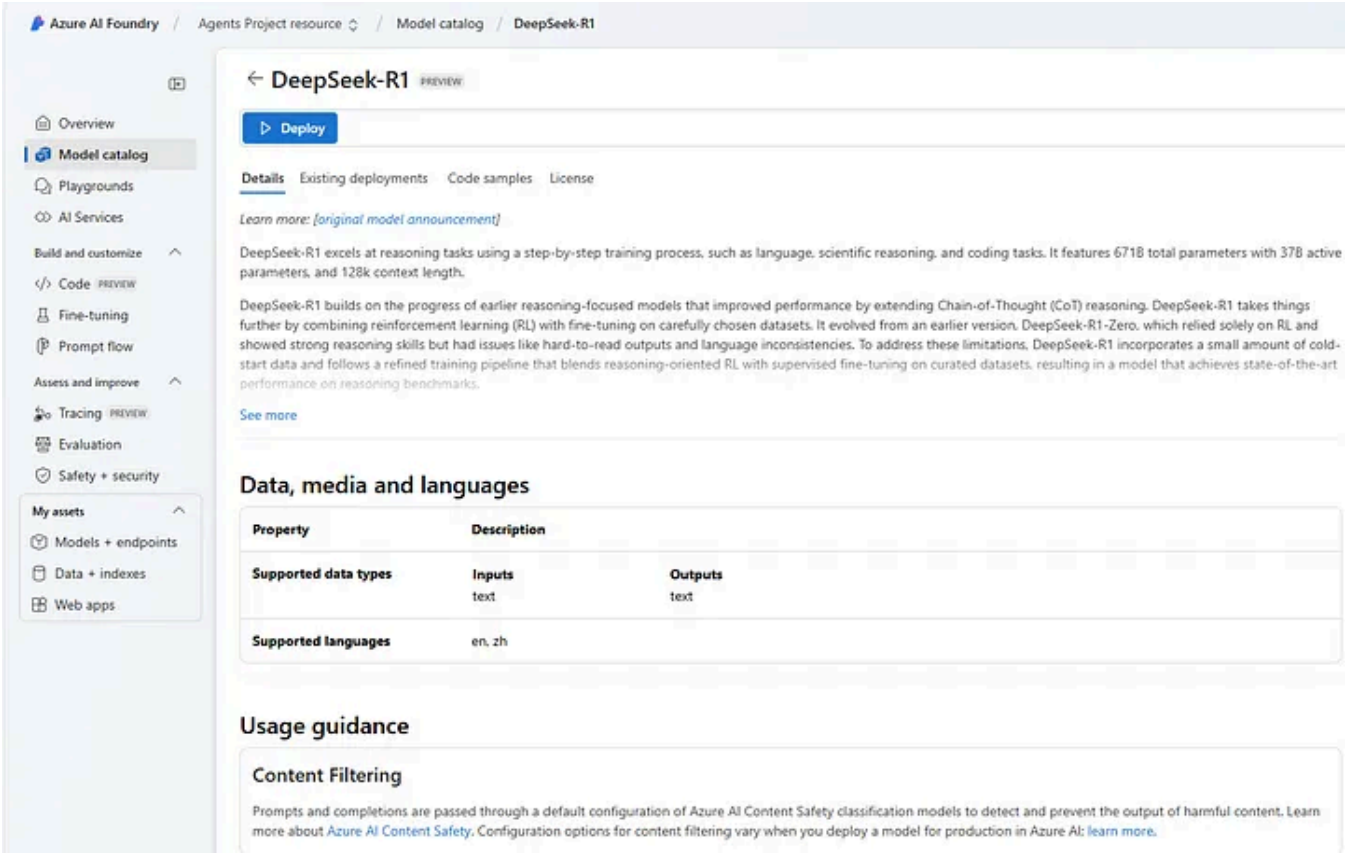
The **Azure Model Catalog** is a centralized repository for pre-trained models, including DeepSeek R1. Hosting your DeepSeek R1 model on Azure using the Model Catalog is straightforward. Here's a step-by-step guide:

Step 1: **Set Up Azure Account**

- Create an Azure account if you don't have one.
- Navigate to the Azure portal and set up a new resource group.

Step 2: **Access the Model Catalog**

- Go to the **Azure Foundry** -> **Model Catalog** in the Azure portal.
- Search for **DeepSeek R1** in the catalog.



Screenshot taken by Author

Step 3: **Deploy the Model**

- Select the DeepSeek R1 model and click on **Deploy.**
- - Configure the deployment settings, including compute resources and scaling options.

Step 4: **Create an Endpoint**

- Once the model is deployed, create a new endpoint.
- - Note down the endpoint URL and API key for future use.

5. Connecting to the Model Using API Keys 🔑

Once your model is hosted on Azure, you can connect to it using API keys. Here's how:

Authentication using API Key: For Serverless API Endpoints, deploy the Model to generate the endpoint URL and an API key to authenticate against the service. In

this sample endpoint and key are strings holding the endpoint URL and the API Key.

The API endpoint URL and API key can be found on the Deployments + Endpoint page once the model is deployed.

Azure inferencing package includes a client for Chat Completion. A client can be authenticated using the API key. The code sample creates and authenticates a synchronous ChatCompletionsClient:

```
from azure.ai.inference import ChatCompletionsClient
from azure.core.credentials import AzureKeyCredential

# For Serverless API or Managed Compute endpoints
client = ChatCompletionsClient(
    endpoint="<ENDPOINT_URL>",
    credential=AzureKeyCredential(key)
)
```

Install dependencies: Install the Azure AI Inference SDK using pip (Requires: Python >=3.8):

```
pip install azure-ai-inference
```

Run a basic code sample: This sample demonstrates a basic call to the chat completion API. The call is synchronous.

```
import os
from azure.ai.inference import ChatCompletionsClient
from azure.ai.inference.models import SystemMessage, UserMessage
from azure.core.credentials import AzureKeyCredential

endpoint = "<ENDPOINT_URL>"
model_name = "DeepSeek-R1"
client = ChatCompletionsClient(
    endpoint=endpoint,
    credential=AzureKeyCredential(key),
)
response = client.complete(
```

```
    messages=[
        SystemMessage(content="You are a helpful assistant."),
        UserMessage(content="I am going to Paris, what should I see?")
    ],
    max_tokens=2048,
    model=model_name
)
print(response.choices[0].message.content)
```

6. Explore More Samples

Run a multi-turn conversation:

This sample demonstrates a multi-turn conversation with the chat completion API. When using the model for a chat application, you'll need to manage the history of that conversation and send the latest messages to the model.

```
import os
from azure.ai.inference import ChatCompletionsClient
from azure.ai.inference.models import AssistantMessage, SystemMessage, UserMess
from azure.core.credentials import AzureKeyCredential

endpoint = "<ENDPOINT_URL>"
model_name = "DeepSeek-R1"

client = ChatCompletionsClient(
    endpoint=endpoint,
    credential=AzureKeyCredential(key),
)

response = client.complete(
    messages=[
        SystemMessage(content="You are a helpful assistant."),
        UserMessage(content="I am going to Paris, what should I see?"),
        AssistantMessage(content="Paris, the capital of France, is known for it
        UserMessage(content="What is so great about #1?")
    ],
    max_tokens=2048,
    model=model_namemodel=model_name
)

print(response.choices[0].message.content)
```

Stream the output:

For a better user experience, you will want to stream the response of the model so that the first token shows up early and you avoid waiting for long responses.

```
import os
from azure.ai.inference import ChatCompletionsClient
from azure.ai.inference.models import SystemMessage, UserMessage
from azure.core.credentials import AzureKeyCredential

endpoint = "<ENDPOINT_URL>"
model_name = "DeepSeek-R1"
client = ChatCompletionsClient(
    endpoint=endpoint,
    credential=AzureKeyCredential(key),
)
response = client.complete(
    stream=True,
    messages=[
        SystemMessage(content="You are a helpful assistant."),
        UserMessage(content="I am going to Paris, what should I see?")
    ],
    max_tokens=2048,
    model=model_name
)
for update in response:
    if update.choices:
        print(update.choices[0].delta.content or "", end="")
client.close()
```

7. DeepSeek R1 Models and AI Agents 🤖

AI agents are autonomous systems that can perform tasks without human intervention. DeepSeek R1 models can significantly enhance the capabilities of AI agents by providing:

- **Faster Inference**: Reduced latency for real-time decision-making.
- **Improved Accuracy**: Higher accuracy in predictions and classifications.
- **Scalability**: Easily scalable to handle large volumes of data.

Example: AI Agent for Customer Support

- Task: Automatically classify customer queries and route them to the appropriate department.
- Implementation: Use DeepSeek R1 models to classify queries in real-time, ensuring quick and accurate routing.

8. Advanced Use Cases 🛠️

Use Case 1: Sentiment Analysis

- **Description**: Analyze customer reviews to determine sentiment.
- **Implementation**: Use DeepSeek R1 models to classify reviews as positive, negative, or neutral.

Use Case 2: Image Recognition

- **Description:** Identify objects in images for inventory management.
- **Implementation**: Deploy DeepSeek R1 models to recognize and classify objects in real-time.

Use Case 3: Fraud Detection

- **Description**: Detect fraudulent transactions in financial data.
- **Implementation**: Use DeepSeek R1 models to analyze transaction patterns and flag anomalies.

9. Conclusion 🎯

DeepSeek R1 models are powerful tools that can transform your AI workflows. Whether you're a beginner or an expert, this guide provides a comprehensive overview of how to host, connect, and utilize these models effectively. By leveraging **Azure Model Catalog** for hosting and API keys for connectivity, you can seamlessly integrate DeepSeek R1 models into your applications. Additionally, the integration with AI agents opens up new possibilities for automation and efficiency.

So, what are you waiting for? Dive into the world of DeepSeek R1 models and unlock the full potential of AI! 🚀

📚 Further Reading

- [Azure Model Catalog Documentation](https://learn.microsoft.com/en-us/azure/machine-learning/concept-model-catalog)
- [DeepSeek R1 on Azure AI Foundry](https://azure.microsoft.com/en-us/blog/deepseek-r1-is-now-available-on-azure-ai-foundry-and-github/)

By following this guide, you'll be well-equipped to harness the power of DeepSeek R1 models and take your AI projects to the next level.

Happy coding! 💻 ✨

Thank You!

> Thanks for taking the time to read my story! If you enjoyed it and found it valuable, please consider giving it a clap (or 50!) to show your support. Your claps help others discover this content and motivate me to keep creating more.
>
> Also, don't forget to follow me for more insights and updates on AI. Your support means a lot and helps me continue sharing valuable content with you. Thank you!

Generative AI

This story is published on Generative AI. Connect with us on LinkedIn and follow Zeniteq to stay in the loop with the latest AI stories.

Subscribe to our newsletter and YouTube channel to stay updated with the latest news and updates on generative AI. Let's shape the future of AI together!

Deepseek | Llm | AI | Technology | Data Science

Following

Published in Generative AI

42K Followers · Last published 9 hours ago

All the latest news and updates on the rapidly evolving field of Generative AI space. From cutting-edge research and developments in LLMs, text-to-image generators, to real-world applications, and the impact of generative AI on various industries.

Edit profile

Written by Naveen Krishnan

158 Followers · 140 Following

AI Architect @ Microsoft. Passionate about leveraging artificial intelligence to solve real-world problems.

No responses yet

Naveen Krishnan

What are your thoughts?

More from Naveen Krishnan and Generative AI