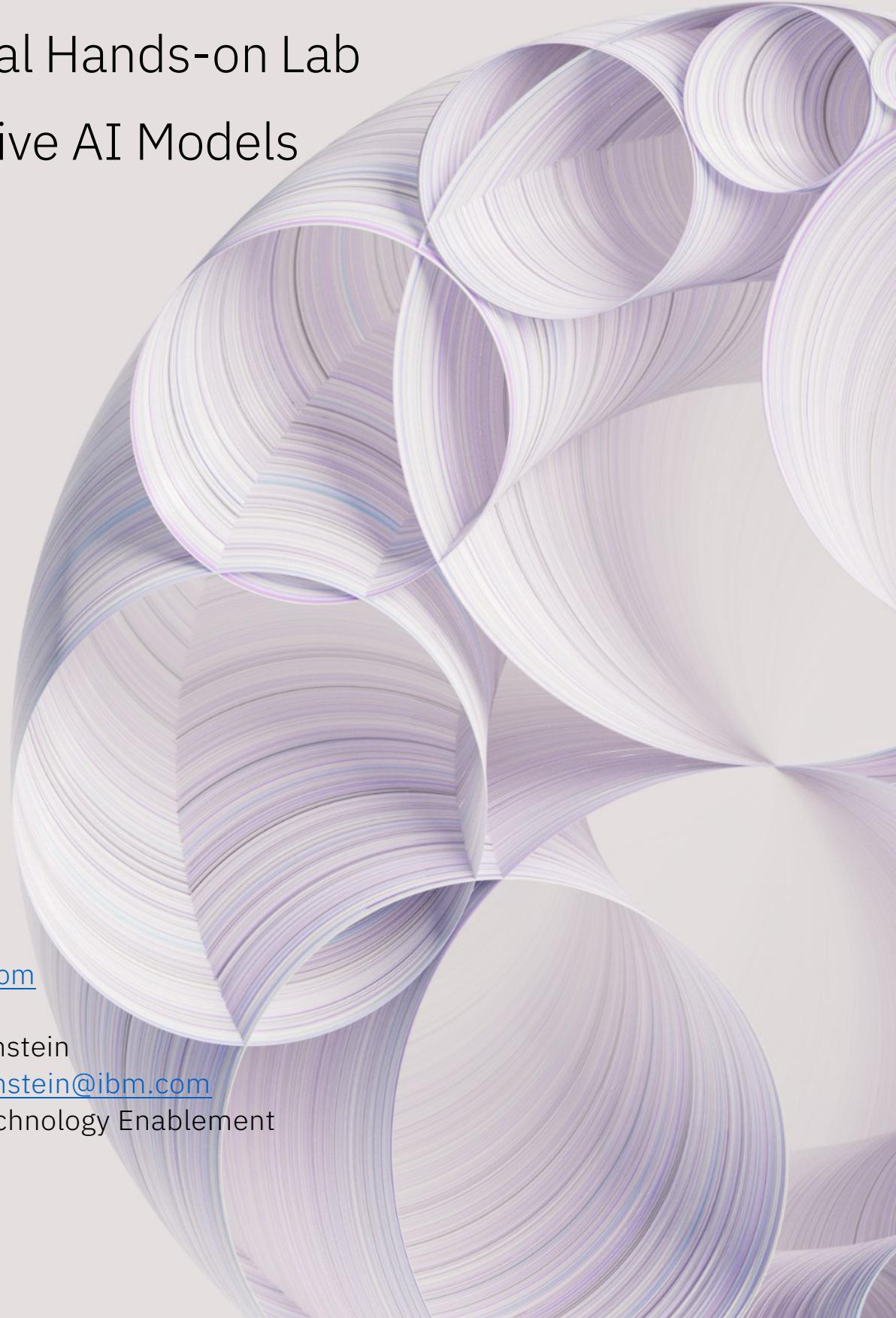


IBM watsonx.ai

Technical Hands-on Lab
Generative AI Models

Felix Lee
felix@ca.ibm.com

Michael Rubenstein
Michael.Rubenstein@ibm.com
Worldwide Technology Enablement



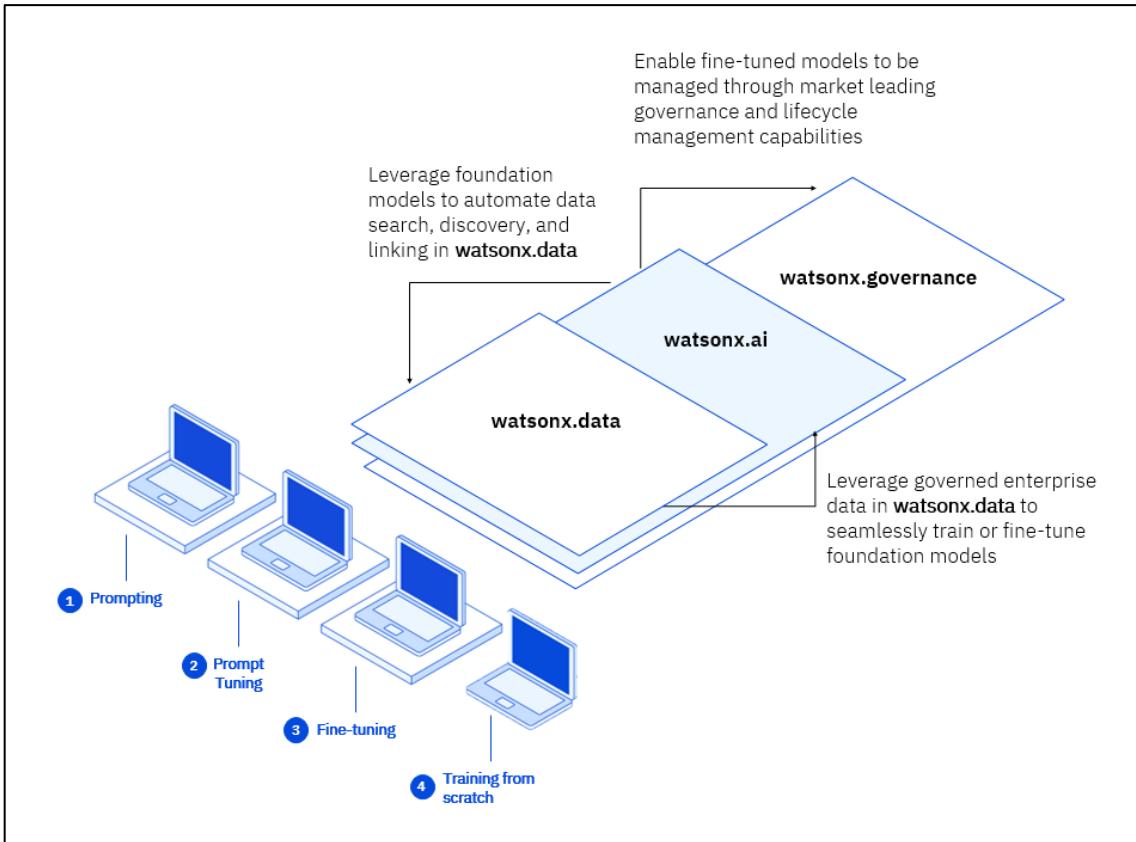
Contents

1 Introduction	3
2 About this lab	4
3 Disclaimer.....	4
3.1 Getting Help.....	4
4 Prerequisites & Getting Started.....	5
4.1 Reserve an environment in IBM Technology Zone.....	5
4.1.1 Verify your TechZone account	12
4.1.2 Logging onto watsonx.ai.....	15
5 Watsonx.ai Foundation Models	19
5.1 Finding Model Details	20
5.1.1 Taxonomy information	27
5.1.2. Filtering by model types.....	29
5.2 Why Model Choice Matters (Model Architecture)	34
5.3 Why Model Choice Matters (Model Training)	43
5.4 Why Model Choice Matters (different instruct models)	53
5.5 Why Model Choice Matters (Multimodal vs Text-only).....	59
6 Custom Foundation Model (CFM)	66
7 Summary	67
Appendix A. Revision History	68

1 Introduction

IBM watsonx.ai is a core component of watsonx, IBM's enterprise-ready AI and data platform designed to multiply the impact of AI across an enterprise's business.

The watsonx platform has three powerful components: watsonx.ai for new foundation models, generative AI, and Machine Learning (traditional AI); the watsonx.data fit-for-purpose data store that provides the flexibility of a data lake with the performance of a data warehouse; and the watsonx.governance toolkit, which enables AI workflows to be built with responsibility, transparency, and explainability.



IBM watsonx.ai (the focus of this lab) makes it possible for enterprises to train, validate, tune, and deploy AI models – both traditional AI and generative AI. With watsonx.ai, enterprises can leverage their existing traditional AI investments as well as exploit the innovations and potential of generative AI building on foundation models to bring advanced automation and AI-infused applications to reduce cost, improve efficiency, scale, and accelerate the impact of AI across their organizations.

2 About this lab

This IBM watsonx.ai hands-on lab introduces you to the various foundation models available in watsonx.ai, and the information that clients use to gain an understanding of the strengths, weaknesses, and capabilities of various models to make appropriate business decisions to maximize their Return on Investment (ROI) with generative AI technology and watsonx.ai. By completing this lab, you will gain the experience needed to demo these capabilities to clients.

Specifically, you will get hands-on experience in the following areas:

- The watsonx.ai web-based user interface (UI) to select models
- Information available in watsonx.ai for models (data taxonomy, fact sheets, and more)

3 Disclaimer

IBM watsonx.ai is being developed and released in an agile manner. In addition to adding new capabilities, the web interface is likely to change over time. Therefore, the screenshots used in this lab may not always look exactly like what you see. You can expect to encounter some of the following:

- A different list of foundation models in the library list. Some models will be deprecated, and new ones will be added frequently to provide the best combinations for clients.
- Changes in the user interface (location of buttons, text for various fields)
- Additional tabs/buttons (especially when the Tuning Studio is rolled out).

If a model mentioned in this lab is deprecated, you can skip the steps (or apply them to another model – although the results will not be the same). Otherwise, there should not be much of a difference in how the labs work, but have patience and explore.

Important notes:

1. New models are constantly being added, and some have been deprecated. Users should refer to the watsonx.ai documentation to find the [latest list](#) (or from the watsonx.ai [View all foundation models](#) page). Some models may be available in specific Geo locations initially (for example, the elyza-japanese-llama-2-7b-instruct is initially only available in the Tokyo data center).

3.1 Getting Help

Lab guide help: If you require assistance in interpreting any of the steps in this lab, please post your questions to the [#data-ai-demo-feedback](#) Slack channel (IBMer only). IBM partners can request help at the [Partner Plus Support](#) website.

Watsonx.ai: Assistance with the watsonx.ai product itself is available in the [#watsonx- ai- feedback](#) (IBMer) and the [#watsonx-ai-enablement](#) Slack channels (IBMer only). Additionally, please refer to the [watsonx.ai documentation](#) as needed.

4 Prerequisites & Getting Started

Important: You must use a TechZone account to perform this lab, or you will run into missing services or other issues. When you use TechZone, you will not (**and should not**) need to subscribe to any additional services (such as the free Lite Watson Machine Learning service).

The detailed setup instructions are as follows.

4.1 Reserve an environment in IBM Technology Zone

1. Go to the [IBM TechZone](#) website.
2. If you already have an IBMid, skip to Step 3. If you do not have one, click **Create an IBMid**.

The screenshot shows the 'Log in to IBM' page. At the top, there is a field labeled 'IBMid' with a placeholder 'Enter your IBMid'. Below it is a blue 'Continue' button with a right-pointing arrow. Underneath the 'IBMid' field is a checkbox labeled 'Remember me' with a small 'i' icon. In the center, there is a red rectangular box around a link that says 'Don't have an account?'. Inside this box is another blue 'Create an IBMid' button with a right-pointing arrow. To the right of the 'Create an IBMid' button is a red circle containing the number '2'. At the bottom of the page, there is a link 'Forgot IBMid? Contact the IBMid help desk'.

3. Provide your IBMid and click **Continue**.

The screenshot shows the 'Log in to IBM' page again. The 'IBMid' field now contains the value 'Michael.Rubenstein@ibm.com', which is enclosed in a red rectangular box. Below the field is a blue 'Continue' button with a right-pointing arrow, also enclosed in a red rectangular box. To the right of the 'Continue' button is a red circle containing the number '3'. At the bottom of the page, there is a checkbox labeled 'Remember me' with a small 'i' icon.

4. Select your Single-Sign-On method. The **w3id Credentials** option is selected in this example:



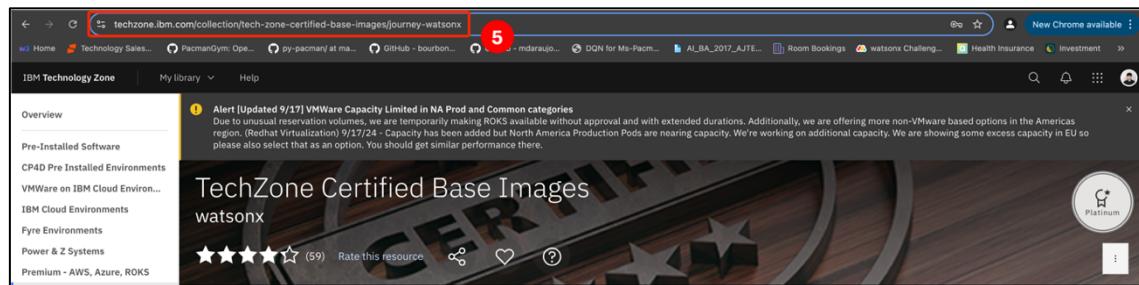
- Provide the necessary information (w3id and password, QR code, etc.) and sign in.

If this is the very first time you are logging in to TechZone, you will be brought to the TechZone dashboard. Copy and paste the following HTTP address in your browser's address field (do not use the magnifying glass icon) and press **Enter**:

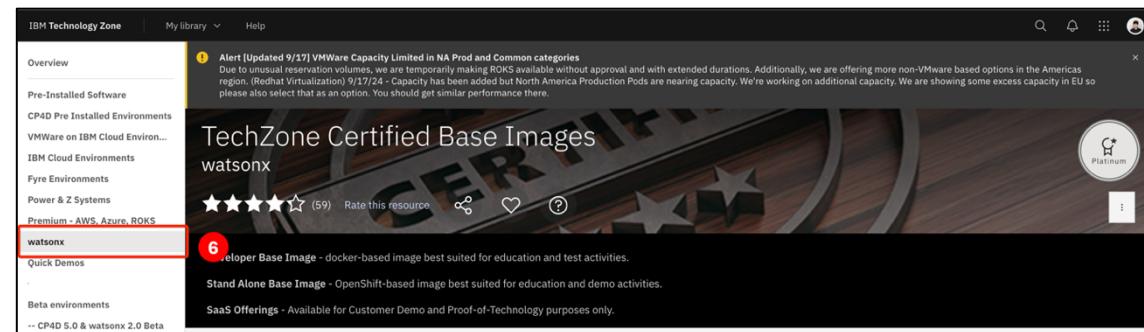
<https://techzone.ibm.com/collection/tech-zone-certified-base-images/journey-watsonx>

You will be taken to the **TechZone Certified Base Images** page.

If you have been to TechZone before, you might be taken directly to the TechZone Certified Base Images page. In that case proceed to Step 6.



- Your browser will open to the TechZone Certified Base Images page. On the left-hand side, the **watsonx** section should be highlighted. If not, click on it



- Scroll down until you find the **watsonx.ai/.governance SaaS** tile (highlighted in red below).

8. Hover your cursor over **IBM Cloud environment** and it changes to **Reserve it**. Click **Reserve it**.

9. The **Create a reservation** page opens. Click **Reserve now**. If you are not ready to start, you can click the **Schedule for later** radio button – the steps are the same.

Now perform the following:

10. Change **Name** of the reservation to <your id> watsonx.ai L3 SaaS. For example, Michael watsonx.ai L3 SaaS.
11. Click the **Practice/Self-Education** tile.
12. For the **Purpose description**, enter Completing L3 Lab.
13. Select your **preferred Geography**. In this example here **AMERICAS** was chosen.

The screenshot shows the 'Name' field (10) containing 'Michael watsonx.ai L3 SaaS'. Below it is a note: 'Name this reservation. This will help identify it in your reservation list.' To the right, there's a 'Purpose' section with three options: 'Demo' (selected), 'Education' (11), and 'Test'. The 'Demo' option has a description: 'Deliver a client specific demonstration based on discovery with the client and aligns to the identified architecture. Automatically captures a Technical Sales Activity in IBM Sales Cloud on the Opportunity code provided.' The 'Education' option is checked and has a description: 'Gaining experience with specific technology, product, or solution.' The 'Test' option has a description: 'Need to test a specific function, configuration, or customization.'

The 'Sales Opportunity number' field (12) contains 'Competing L3 Lab'. Below it is a note: 'What are you doing? Why do you need this? What are you trying to accomplish?' The 'Preferred Geography' field (13) contains 'any - AMERICAS - us-south region - any datacenter'.

14. Specify the **End date and time**. By default, the end day is 14 days from the current date when you make this reservation. You can extend it up to a maximum of 4 days. Note that both the duration and extension capability are subject to TechZone policies that could change at any time.

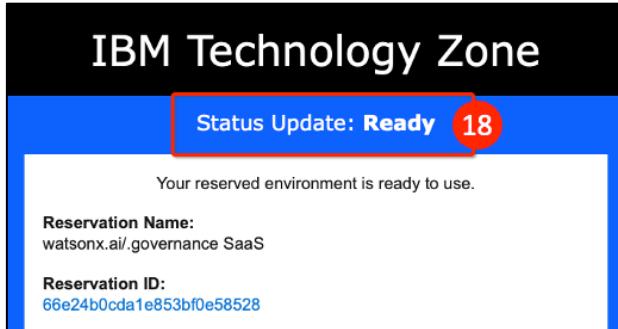
15. Ensure the **Install DB2** field is set to No.

16. You can optionally provide some additional notes in the **Notes** field if you choose

17. On the right-hand side at the bottom, accept the **Terms and Conditions**, then click **Submit**.

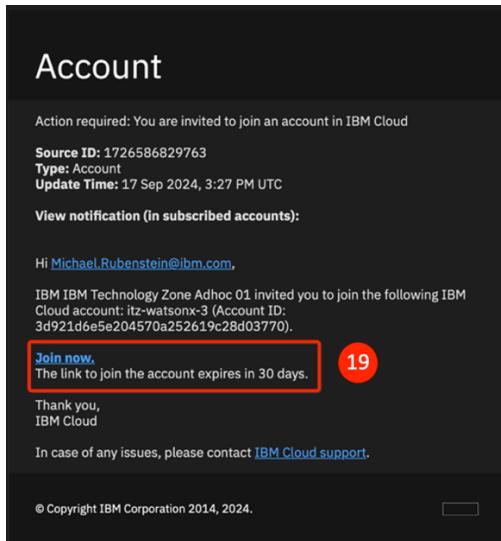
The screenshot shows the 'End date and time' section (14) with the date set to '10/01/2024' and the time to '11:10 AM' in 'America/Toronto'. A note below says: 'Reservation policy: Recommended 2 weeks, but can be reserved up to 2 weeks on this reservation form. Extend later for 4 days increments up to 4 days total. Max time 3 weeks total.' The 'Install DB2?' field (15) is set to 'No'. The 'Notes' field (16) contains 'Enter any notes you would like to attach to this reservation'. At the bottom right, there's a checkbox labeled 'I agree to IBM Technology Zone's Terms & Conditions and End User Security Policies' (17) and a blue 'Submit' button.

18. You will get an email sent to your IBMid email address letting you know that your reservation is being provisioned (this typically happens instantly, but can take 10-15 minutes). Once the watsonx.ai provisioning is completed, you will receive a second email telling you that it is ready for use.



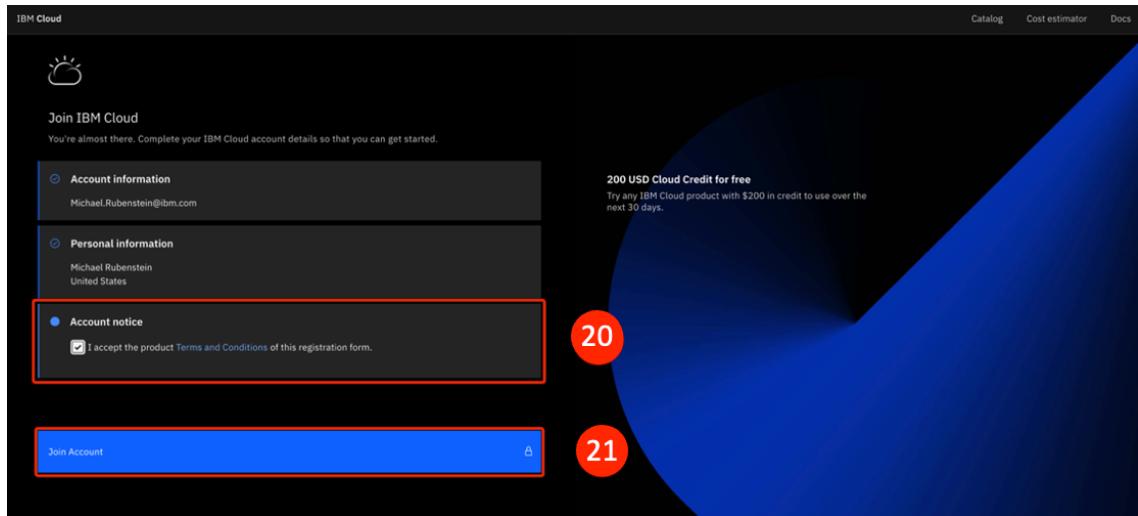
19. If TechZone had previously assigned this account to you and you have already accepted the invitation, there is nothing more to do. You can skip to Section 4.1.1.

Otherwise, you will see an invitation email to join a TechZone Account like the following. Click **Join now**.

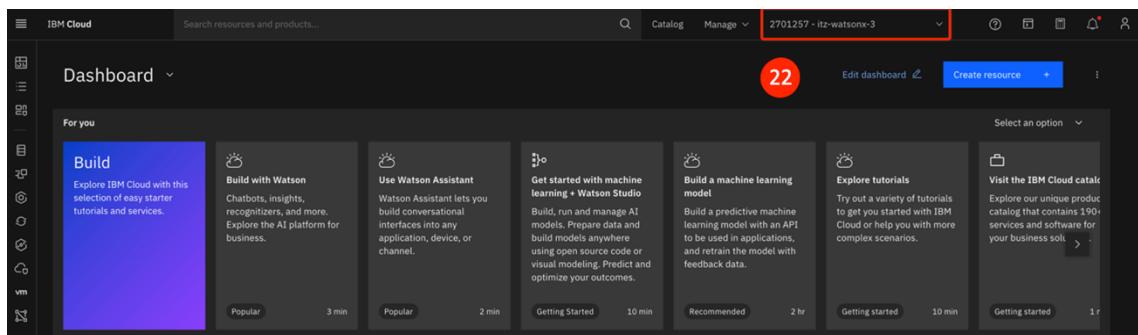


20. Click the **I accept the product Terms and Conditions of this registration form** checkbox.

21. Click **Join Account**.



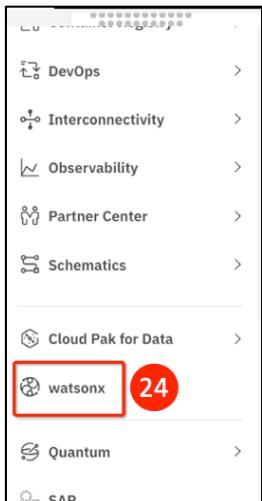
22. The Dashboard opens in an account that should be the same one that TechZone assigned to you. If you have had previous TechZone reservations you may have multiple **itz-watsonx-n** accounts listed in the drop-down. If for some reason this is not showing the currently assigned TechZone account, click the dropdown and select the appropriate account.



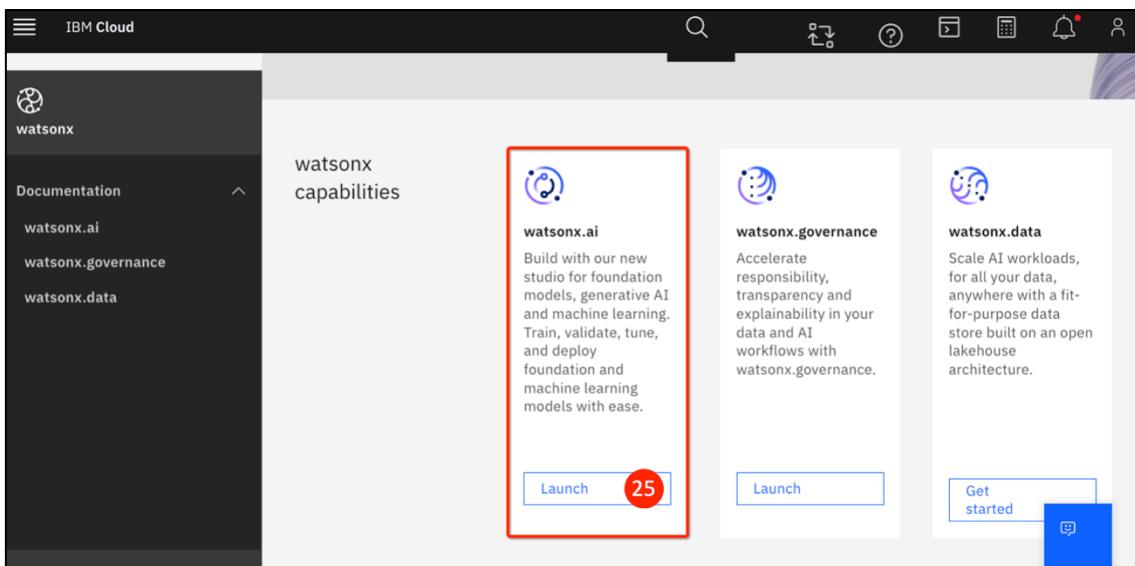
23. Click on the main menu in the upper left corner.



24. On the pull-out menu, scroll down and click **watsonx**.

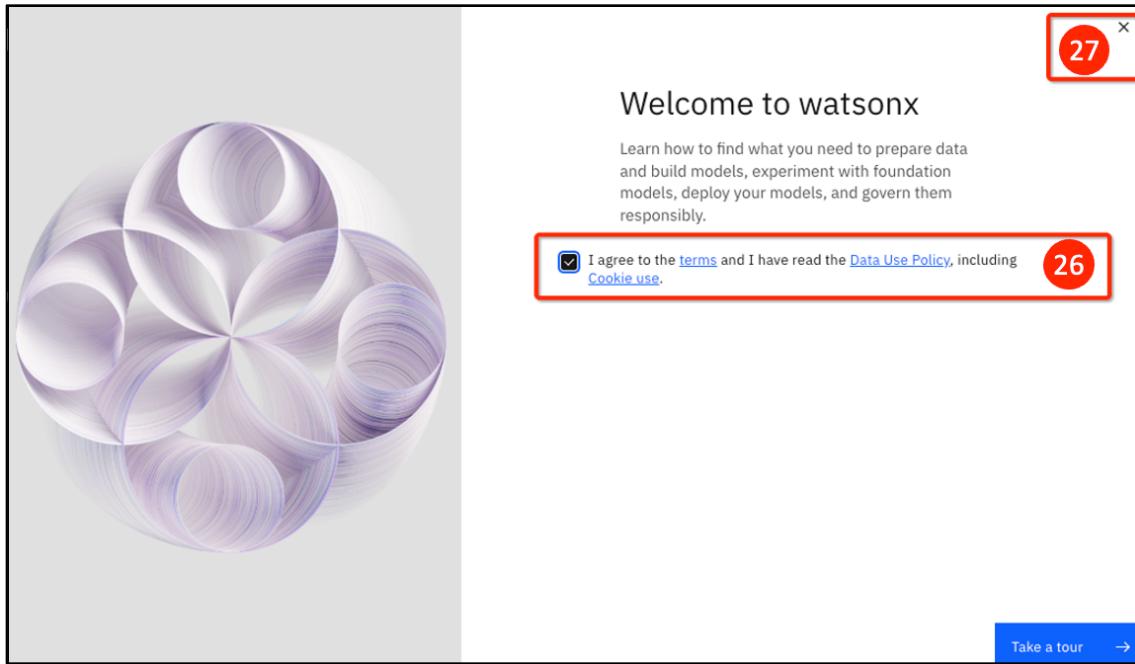


25. The **watsonx capabilities** page opens. Click the **watsonx.ai** tile's **Launch** button.



26. You may see a **Welcome to watsonx** panel. Agree to the terms. If not, skip this step.

27. Click **X** to close this window if you get the pop-up.



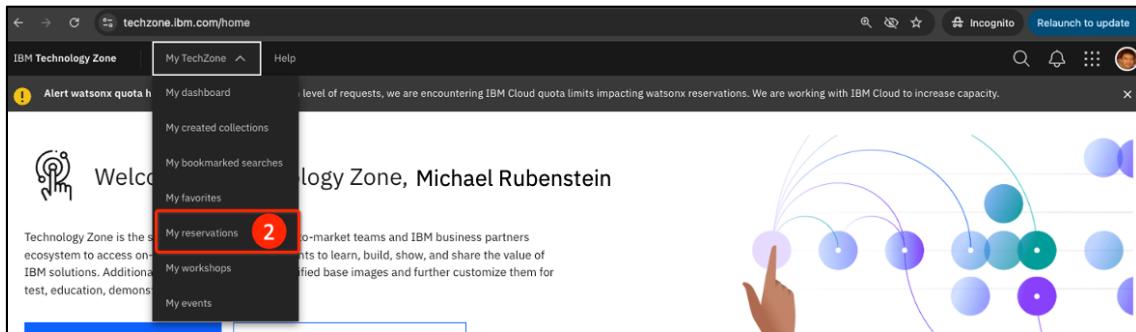
You are now on the watsonx.ai console and ready to continue to explore the various generative AI model detailed in this lab.

4.1.1 Verify your TechZone account

You must always **ensure you are using the right account** when logging into watsonx.ai. If you have completed a wastonx.ai L3 lab already, you have already been added to a **itz-watsonx-n** account. Note that once your reservation has expired, you cannot use that account anymore.

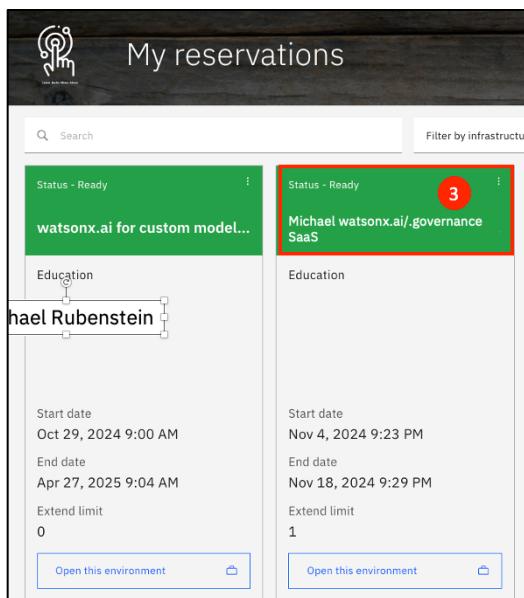
When you make a new reservation, TechZone may assign a different **itz-watsonx-n** account. It is very important to verify that you are using the currently assigned account for this lab to work. **If you encounter messages that ask you to upgrade your Watson Machine Learning (WML) service or to associate a WML instance, it is indicative that you might be using the wrong account.** To check what is currently assigned, do the following.

1. Go to <http://techzone.ibm.com/dashboard>
2. Click **My TechZone**, then click **My reservations**



3. The **My reservations** page opens. Look for the **watsonx.ai/.governance SaaS** tile that represents the instance you provisioned for this lab. You might have changed the Name in Step 10 of Section 4.1. For example: **Michael watsonx.ai/.governance SaaS** as shown in the screen capture below.

Click on the tile.



4. The detailed reservation page appears. Check the value of **Cloud Account**. In this example, it is **ITZ-WATSONX-3**.
5. Check the value of **Region**. In this example, it is **us-south**.

Purpose	
Purpose	Opportunity ID(s)
Education	
Opportunity Product(s)	Opportunity description
Customer(s)	Completing L3 Lab
Environment	
Reservation ID	Type
66e99f128ddc432ac13bf4525	IBM Cloud
Request method	Transaction ID
watsonx-ai-saas	e20845fe-06ed-4783-a8eb-2fcc3d48500b
Cloud Account ITZ-WATSONX-3	Geo americas
Region us-south	Datacenter dal10
Customer data false	Environment watsonxai-v3mufuwi

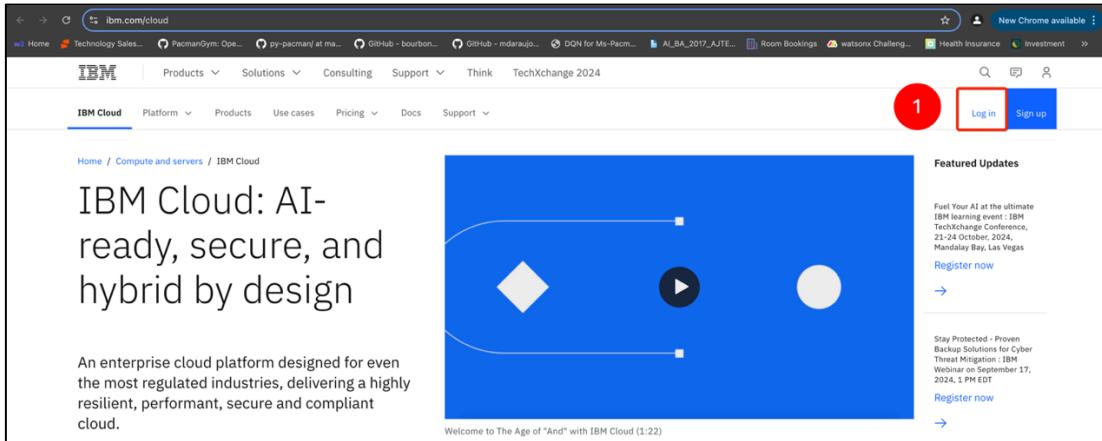
6. When you log into watsonx.ai, ensure you are using the right **Account** and **Region**: in this example, they are **ITZ-WATSONX-3** and **us-south** (or **Dallas**) respectively. **If either of the values is wrong, the lab will not work.**
7. Note that the **Region** value from the **TechZone** reservation page is different from what is shown on watsonx.ai. Use the following table for the mapping value:

Value from TechZone detail page	Value on watsonx.ai User Interface
us-south	Dallas
eu-de	Frankfurt
eu-gb	London
jp-tok	Tokyo

4.1.2 Logging onto watsonx.ai

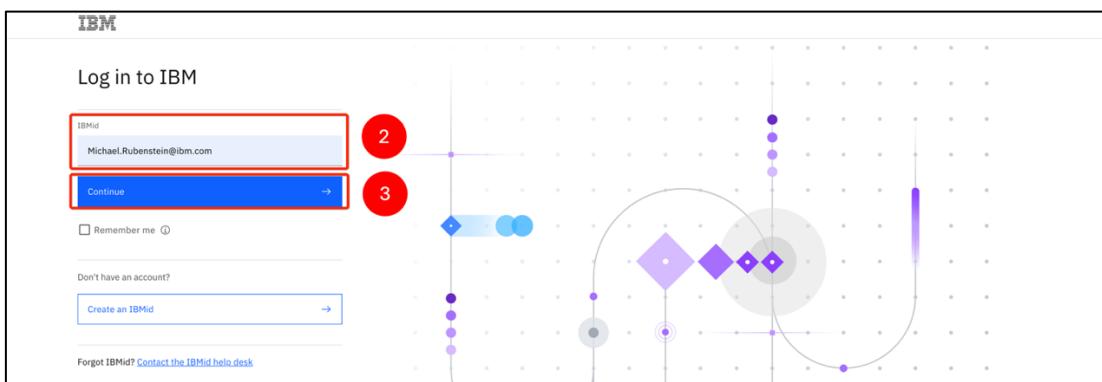
If you have logged out (or if your watsonx.ai session has timed out), you can log back in as follows.

1. Go to <https://www.ibm.com/cloud> and click **Log in**.



2. Enter your IBMid

3. Click **Continue**.

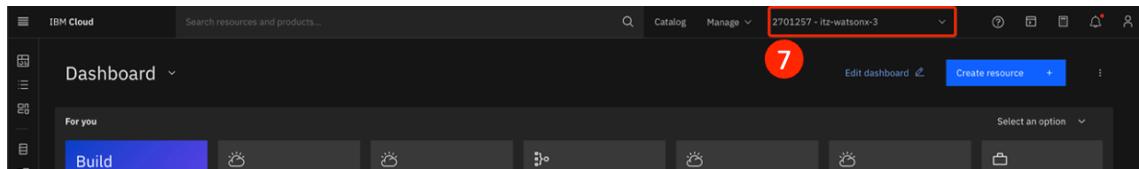


4. You may be asked to sign in. If so, click **w3id Credentials** (or use whichever authentication method you are set up with).

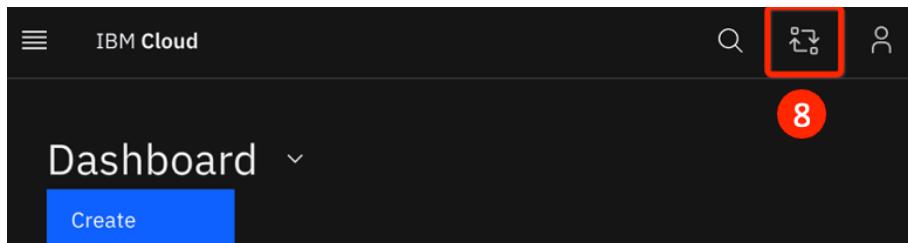


5. Enter your ID and password and sign in.

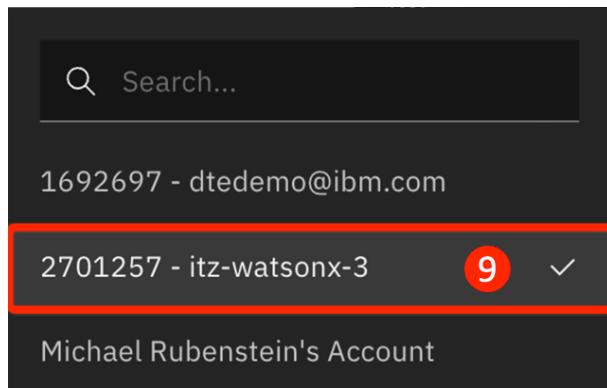
6. You are at the IBM Cloud dashboard.
7. If you are on full-screen mode (if you see an account name in the field, you are in full-screen mode), check to ensure that you are using the account TechZone assigned to you. If you are not in full-screen mode, skip to Step 8.



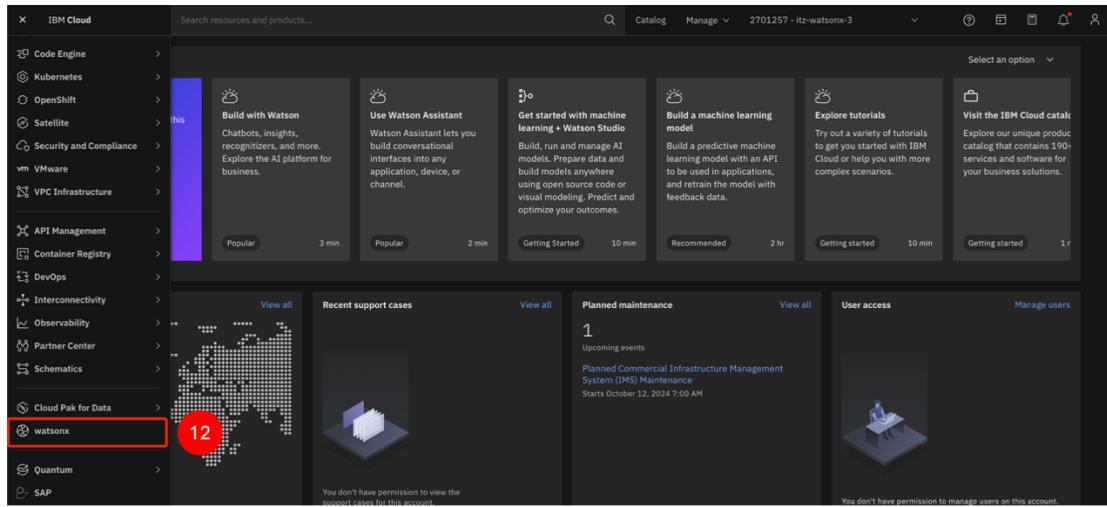
8. If you are not on full screen, click the Account icon (✉) on the upper right.



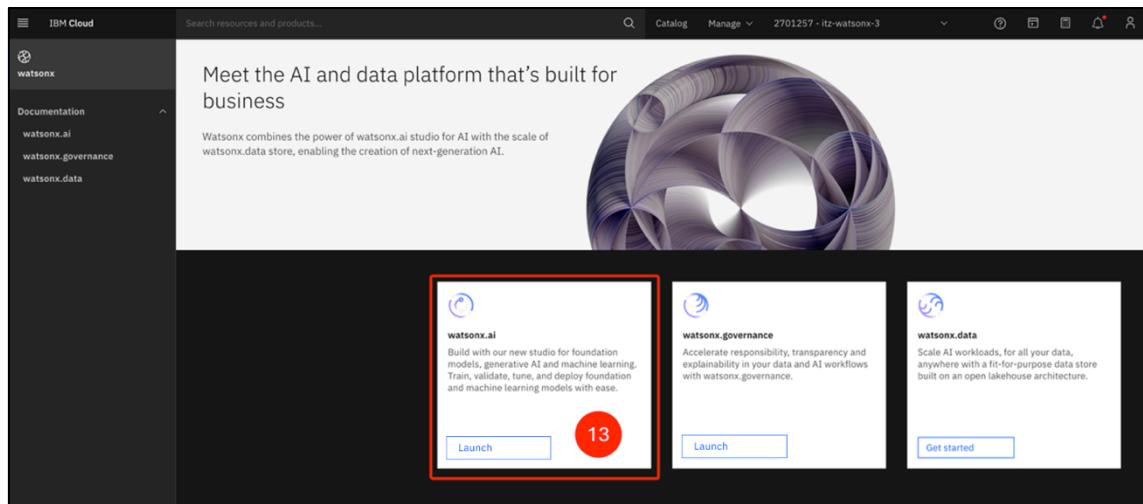
9. Ensure you are using the TechZone account currently assigned to you.



10. Click the Navigation Menu icon (☰) on the top left of the screen.
11. Scroll down on the Navigation Menu until the bottom.
12. Click on the **watsonx** navigation button.



13. You will now see the AI and data platform page. Click **Launch** from the watsonx.ai tile.



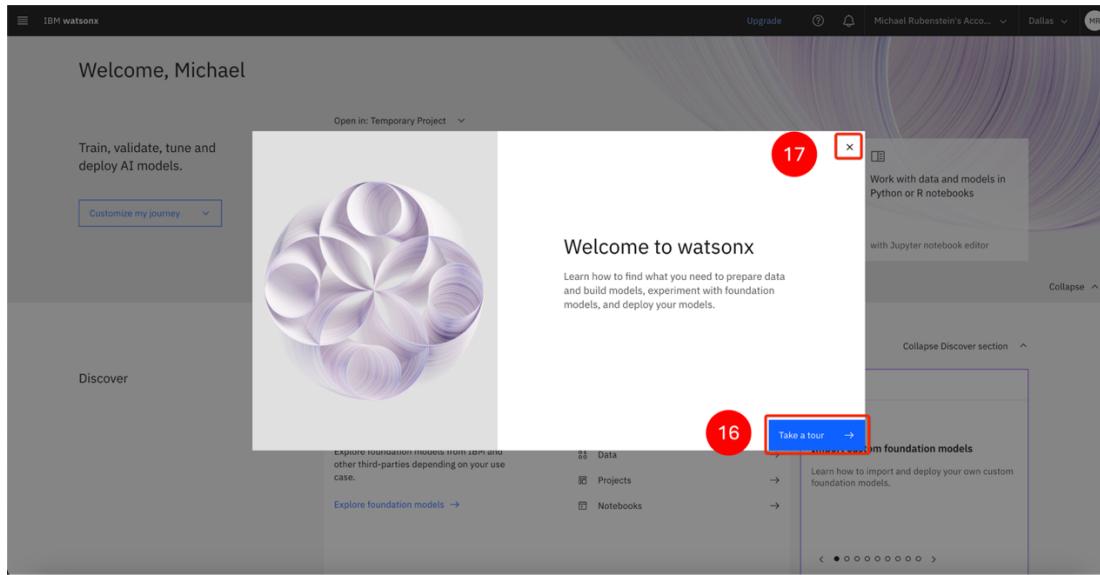
14. If you do not see **Launch**, enter the following into the URL:

- Dallas:** <https://dataplatform.cloud.ibm.com>
- Frankfurt:** <https://eu-de.dataplatform.cloud.ibm.com>
- London:** <https://eu-gb.dataplatform.cloud.ibm.com>
- Tokyo:** <https://jp-tok.dataplatform.cloud.ibm.com>

15. You are now in the watsonx.ai console and ready to begin the first part of this lab. You might see the following page.

16. You can click on the **Take a tour →** button on the bottom-right if you are interested in taking a tour of watsonx.ai

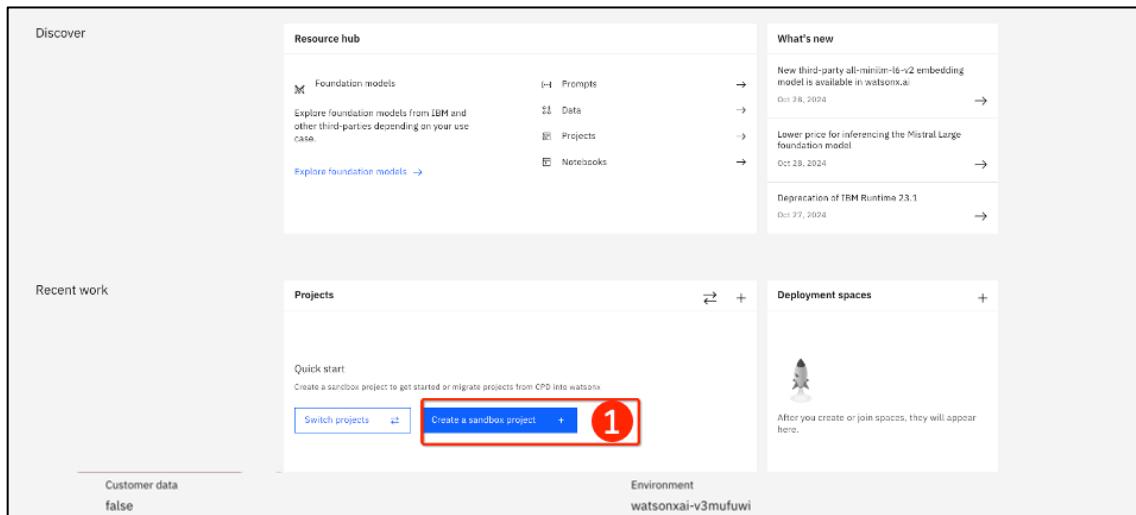
17. Or you can simply click on the **X** on the upper right to close this window.



You are now ready to explore watsonx.ai console to learn about models.

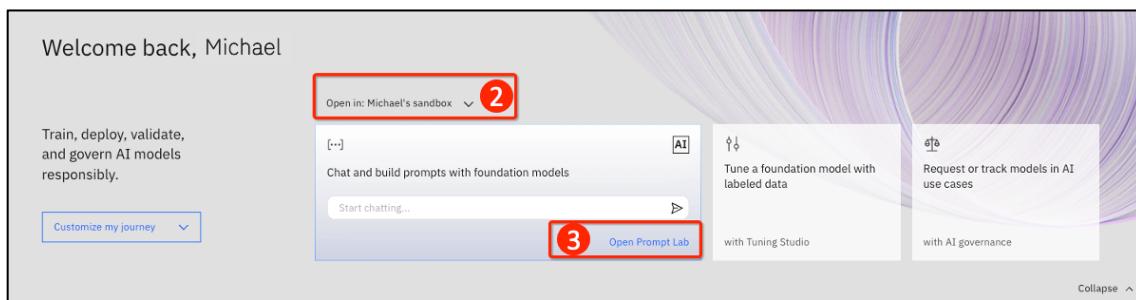
5 Watsonx.ai Foundation Models

29. If this is the first time you log into watsonx.ai using this TechZone account, you will need to create a sandbox project. If a sandbox project gets created when you log in, skip to the next step. If not, scroll down to the **Projects** area and click **Create a sandbox project**.



30. The sandbox project has a name like <Your username>'s sandbox. In this example it is **Michael's sandbox**. Once the sandbox project is created

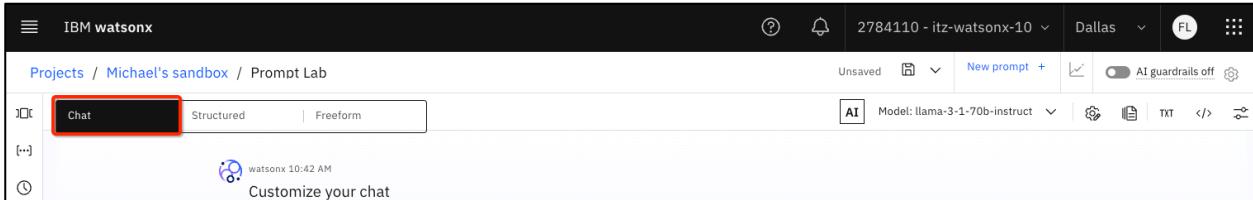
31. From the watsonx.ai console, click on **Open Prompt Lab** to open it.



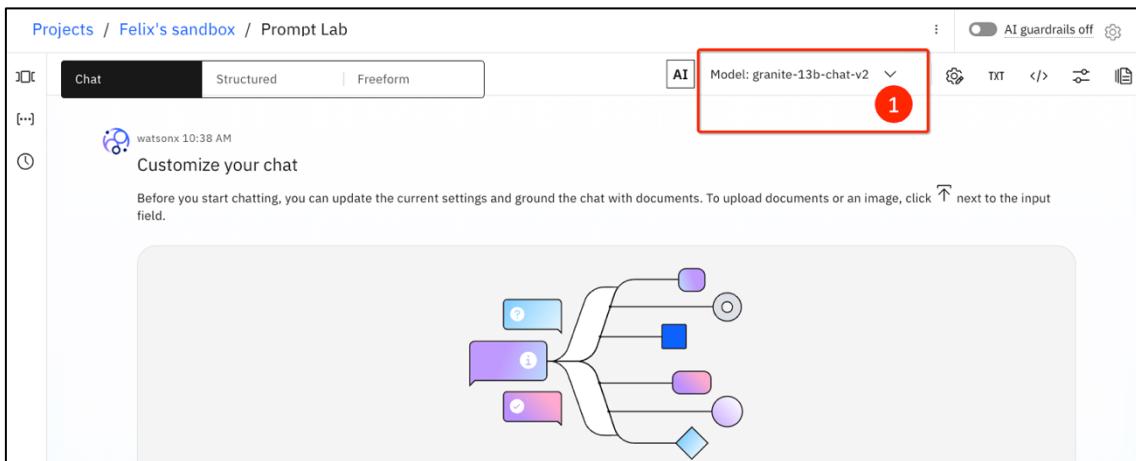
5.1 Finding Model Details

The Prompt Lab provides many capabilities, but this lab only examines a subset of them. One of the most important tasks for users is to identify the models they have access to.

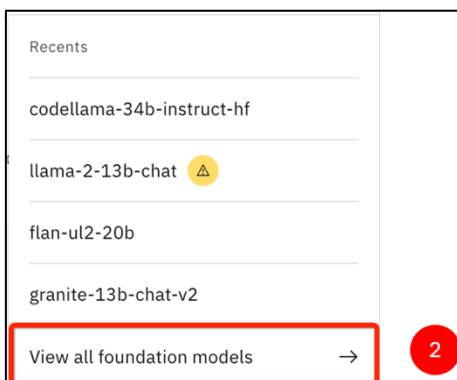
The **Prompt Lab** by default opens to the **Chat** tab. If not, select the **Chat** tab.



1. You can change to different foundation models by clicking the dropdown menu beside the currently selected model.



2. For this exercise, click **View all foundation models**.



3. The **Select a foundation model** watsonx.ai page opens, showing the foundation models currently available to your account (partially shown below). Note that by default the **Chat** filter is turned on.

The screenshot shows a search interface for foundation models. At the top, there's a search bar and a 'Reset Filters' button. Below that is a 'Filter' section with dropdown menus for 'Task', 'Provider', 'Modality', and 'Model type'. The main area displays a grid of model cards. The first three cards are highlighted with a red circle containing the number '3':

- granite-13b-chat-v2**: Provider IBM, Type InstructLab. A blue circular icon with a question mark is next to it.
- granite-13b-instruct-v2**: Provider IBM, Type Provided model. A yellow circular icon with an upward arrow is next to it.
- granite-20b-code-instruct**: Provider IBM, Type Provided model. A blue circular icon with a question mark is next to it.

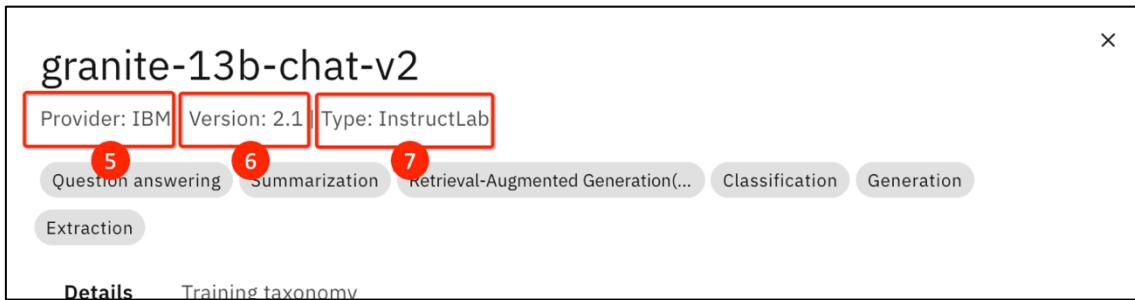
Each card contains a brief description of the model's purpose and provider.

Notes:

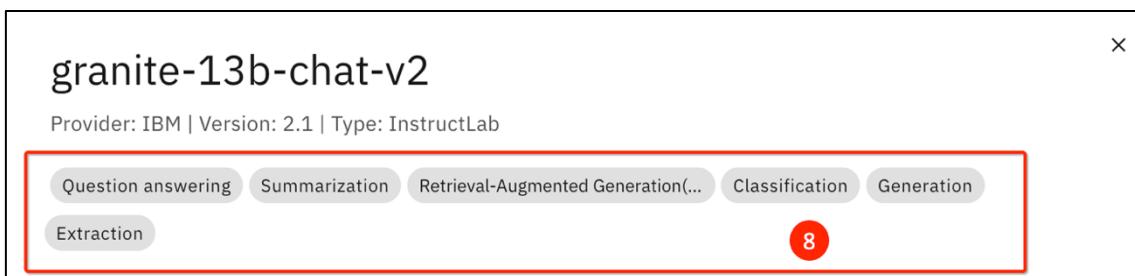
- You might see models marked by a blue icon . This means the models are recommended.
- You also might see some models being marked by a yellow icon . This means that this model is being deprecated soon. To learn more information about a model's deprecation date, please hover your mouse over this icon.
- You will see open-source models, IBM models, and any Custom Foundation Models (CFMs) that you may have added. The Granite models are developed by IBM. As of October 2024, these are the granite models available:

- granite-13b-chat-v2
- granite-13b-instruct-v2
- granite-20b-code-instruct
- granite-20b-multilingual
- granite-34b-code-instruct
- granite-3b-code-instruct
- granite-7b-lab (marked for deprecation)
- granite-8b-code-instruct
- granite 3-2b-instruct
- granite 3-8b-instruct
- granite-guardian-3-2b
- granite-guardian-3-8b

- Click the **granite-13b-chat-v2** tile, and its model card opens. On the top of the model card, there will be three important pieces of information that is covered in the next steps.
- At the top, **Provider** indicates the vendor that trained and provided the model. **IBM**, in this example.
- At the top, **Version** shows the version of the model. It is useful to mark this as new versions of the models are being released. In production, clients should be aware of the version they are using. Different versions (of the same model) can lead to different results. **Version 2.1**, in this example.
- At the top, **Type** shows whether this model is fine-tuned by InstructLab or directly pre-trained by the model provider. In this example, this model was **InstructLab** tuned.



8. Another useful piece of information is being able to see the use cases recommended for the model.



As the image above shows, there are seven use cases supported by this model:

- Chat
- Question Answering
- Summarization
- Retrieval-Augmented Generation (RAG)
- Classification
- Generation
- Extraction

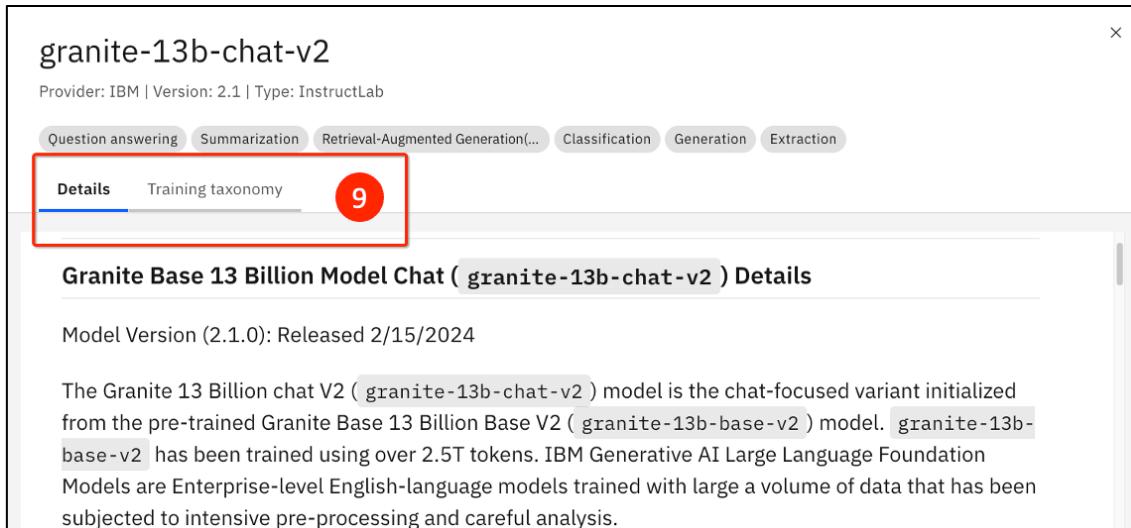
Note that Chat is not explicitly listed in this model card because the ability to chat is a baseline for all models (except the new granite 3.0 guardrail models).

One typical generative AI use case not listed is the Code Generation and Conversion use case. This means that the **granite-13b-chat-v2** model should not be used for Code Generation and Conversation.

Note: Clients looking for a model should ensure that the model is recommended for their use case(s).

9. Scrolling down (do not click Select model yet), you can find the following information for the models:
 - **Details:** This tab shows the **Model Datasheet**, which includes information such as training set details, model architecture, use cases, validation metrics, version releases, and more.
 - **Training Taxonomy:** This tab shows a detailed tree breaking down the taxonomy of training data used to fine-tune the model via InstructLab in this case.

Note: The Training Taxonomy tab is only available for InstructLab models.

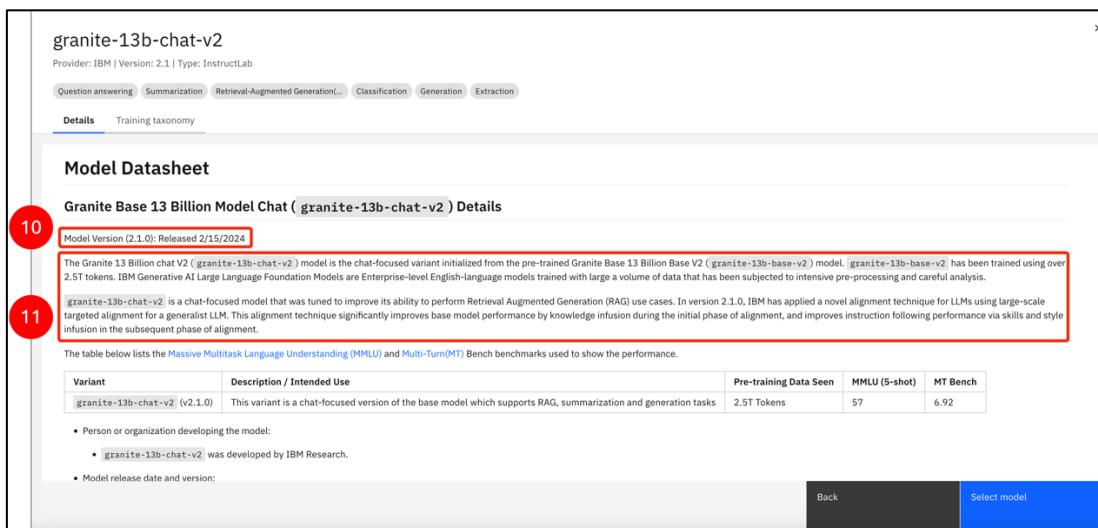


The screenshot shows the 'granite-13b-chat-v2' model details page. At the top, it says 'Provider: IBM | Version: 2.1 | Type: InstructLab'. Below that is a navigation bar with tabs: Question answering, Summarization, Retrieval-Augmented Generation(...), Classification, Generation, and Extraction. The 'Details' tab is selected and highlighted with a red box and a red circle containing the number 9. The main content area is titled 'Granite Base 13 Billion Model Chat (granite-13b-chat-v2) Details'. It includes the release date 'Model Version (2.1.0): Released 2/15/2024' and a detailed description of the model's capabilities and training.

Model Version (2.1.0): Released 2/15/2024

The Granite 13 Billion chat V2 (granite-13b-chat-v2) model is the chat-focused variant initialized from the pre-trained Granite Base 13 Billion Base V2 (granite-13b-base-v2) model. granite-13b-base-v2 has been trained using over 2.5T tokens. IBM Generative AI Large Language Foundation Models are Enterprise-level English-language models trained with large a volume of data that has been subjected to intensive pre-processing and careful analysis.

10. The **Details** tab provides key information about the model, including its release date and version. For example, this model example was released on **2/15/2024**, and the version is **2.1.0**. Clients can use the release date to assess how current the model is.
11. Below the model version, a summary and description of the model offers clients insights into the model's capabilities. For this example as noted in the datasheet, the **granite-13b-chat-v2** model was designed for chat purposes and further fine-tuned for RAG use cases. This information can furthermore help clients determine if the model is suitable to fit their needs.



The screenshot shows the 'Model Datasheet' page for the same model. It includes the 'Details' tab and a 'Model Datasheet' section. A red circle with the number 10 highlights the 'Model Version (2.1.0): Released 2/15/2024' text. A red circle with the number 11 highlights the descriptive text about the model's capabilities and training. Below this is a table of benchmarks and a list of developer information.

Model Version (2.1.0): Released 2/15/2024

The Granite 13 Billion chat V2 (granite-13b-chat-v2) model is the chat-focused variant initialized from the pre-trained Granite Base 13 Billion Base V2 (granite-13b-base-v2) model. granite-13b-base-v2 has been trained using over 2.5T tokens. IBM Generative AI Large Language Foundation Models are Enterprise-level English-language models trained with large a volume of data that has been subjected to intensive pre-processing and careful analysis.

granite-13b-chat-v2 is a chat-focused model that was tuned to improve its ability to perform Retrieval Augmented Generation (RAG) use cases. In version 2.1.0, IBM has applied a novel alignment technique for LLMs using large-scale targeted alignment for a generalist LLM. This alignment technique significantly improves base model performance by knowledge infusion during the initial phase of alignment, and improves instruction following performance via skills and style infusion in the subsequent phase of alignment.

Variant	Description / Intended Use	Pre-training Data Seen	MMLU (5-shot)	MT Bench
granite-13b-chat-v2 (v2.1.0)	This variant is a chat-focused version of the base model which supports RAG, summarization and generation tasks	2.5T Tokens	57	6.92

• Person or organization developing the model:
• granite-13b-chat-v2 was developed by IBM Research.

• Model release date and version:

Back Select model

For example, in the case of **granite-13b-chat-v2**, the following is highlighted: “**granite-13b-chat-v2 is a chat-focused model...**”. In this case, the name of the model already gave this away. However, for other models, it is useful to read the details to see what the model is **focused on** addressing versus just “**also should work**” for other use cases.

12. Scrolling further down, you can find even more detailed information about the organization that developed the model, in this case, it was **IBM Research**.

13. More details about a model's release date and version information are listed next.

14. As well as details about the model type and training architecture details. This information is useful:

- For explainability. IBM models like the Granite series provide this information for transparency. This is important if enterprises are required to use models that they may need to explain.
- For Data Scientists to understand if the model was built/tested in a manner that is acceptable for their use cases. Some subtle differences and nuances may not surface in general tests but are fundamental based on how the model was built. This information can help a data scientist to decide if the model is suitable.

The screenshot shows a web-based interface for viewing a specific AI model. At the top, it displays the model name 'granite-13b-chat-v2', its provider ('IBM'), version ('2.1'), and type ('InstructLab'). Below this, there are several tabs: 'Question answering', 'Summarization', 'Retrieval-Augmented Generation...', 'Classification', 'Generation', and 'Extraction'. The 'Details' tab is currently selected, showing a 'Training taxonomy' section. Three specific sections are highlighted with red boxes and circled with numbers: 12 points to the 'Person or organization developing the model' section, which lists 'granite-13b-chat-v2' was developed by IBM Research; 13 points to the 'Model release date and version' section, which lists Model version 2.1.0 released 2/15/2024 and originally released on 11/30/2023; 14 points to the 'Model type' section, which details the model's architecture (decoder-only transformer), features (Decoder-only model, Multi-Query Attention, 50K GPT-NeoX tokenizer, Full Attention, 5.8k context length, Absolute (learned) position embeddings), and training parameters (using 4x Tensor Parallel + 4x Pipeline Parallel + Megatron distributed optimizer Megatron-LM, Cluster: CCC, GPUs: 256x A100 80GB, Interconnect: 200 gigabit Infiniband, Dataset streamed over GPFS).

15. Scrolling down more, you can find a link to the model's research paper, providing in-depth insights into its development and underlying methodologies.

16. Additionally, information about the licensing them for using the model is included, detailing who is eligible to access and utilize it. In this case, this model is only available through IBM products and offerings.

17. The release notes for the current version of the model is also listed, outlining any updates or changes made.

The screenshot shows the 'granite-13b-chat-v2' model card. At the top, it says 'Provider: IBM | Version: 2.1 | Type: InstructLab'. Below that are tabs for 'Question answering', 'Summarization', 'Retrieval-Augmented Generation...', 'Classification', 'Generation', and 'Extraction'. A 'Details' tab is selected, showing 'Training taxonomy'. The 'Intended Use' section contains a list of primary intended uses, one of which is highlighted with a red box and circled with a red number '15'. The 'License' section also has a red box around its content and is circled with a red number '16'. Below these sections is a 'Version Release notes' section, which is also highlighted with a red box and circled with a red number '17'. At the bottom right are 'Back' and 'Select model' buttons.

18. Scrolling even further down, you can see details about the intended use case for this model. For example, the current model is intended for **English-based closed-domain Question and Answering (e.g., RAG), summarization, and generation, extraction, and classification**. This provides clients with more information about whether this model is intended for their specific use case(s).

Note: Not all model cards include this much detailed information (this is a benefit of an IBM model). Clients may need to read through the model card to extract this information. However, it is important to understand what the model card says on how the model should be used. Applying a model to a use case it was not designed and built for will not yield good results.

19. There are also sections regarding evaluation metrics and the performance of the model. This gives clients a better quantitative metric to compare models for their specific use cases.

The screenshot shows the 'granite-13b-chat-v2' model card. It includes sections for 'Intended Use', 'Factors', and 'Metrics'. The 'Intended Use' section is highlighted with a red box and circled with a red number '18'. The 'Factors' section is also highlighted with a red box and circled with a red number '19'. The 'Metrics' section shows a table with two rows: 'MMLU' and 'MT-Bench'. The table has columns for 'Benchmark' and 'Description'.

Benchmark	Description
MMLU	Benchmark used to measure a text model's multitask accuracy. The test covers 57 tasks including elementary mathematics, US history, computer science, law, and more.
MT-Bench	Benchmark consisting of 80 high-quality multi-turn questions. MT-bench is designed to test multi-turn conversation and instruction-following ability, covering common use cases and focusing on challenging questions to differentiate models.

20. At the bottom are details of the data used (and limitations) for pretraining. Models are highly dependent on their training data. Hence, clients need to know about what the model trained on to see if that is suitable for their own use case or to identify any knowledge gaps needed.

granite-13b-chat-v2

Provider: IBM | Version: 2.1 | Type: InstructLab

Question answering · Summarization · Retrieval-Augmented Generation · Classification · Generation · Extraction

Details Training taxonomy 20

Data, Limitations, and Recommendations

- Data selection for training:
 - The `granite-13b-base-v2` model underwent extended training using 2.5 Trillion tokens of IBM's curated pre-training dataset. On top of the 2.5 trillion tokens from `granite-13b-base-v2`, the `granite-13b-chat.v2` model underwent knowledge and skill focused tuning. A breakdown of the sampling data used for training is shown in the table below.

Dataset sampling for Granite (13B) Base V2.0

Dataset	Description
Common Crawl	Open repository of web crawl data.
Webbase	Unstructured web content converted into machine-readable data feeds acquired by IBM.
arXiv	Over 1.8 million scientific paper pre-prints posted to arXiv.
Wikimedia	Eight English Wikimedia projects (enwiki, enwikibooks, enwikinews, enwikisource, enwikiversity, enwikivoyage, enwiktionary) containing extracted plain text from pages and articles.
OpenWeb Text	Open-source version of OpenAI's Web Text corpus containing web pages through 2019.
Stack Exchange	Anonymized set of all user-contributed content on the Stack Exchange network, a popular collection of websites centered around user-contributed questions and answers.
Hacker News	News on computer science and entrepreneurship, taken between 2007-2018.
Project Gutenberg PG19	A repository of free e-books with focus on older works for which U.S. copyright has expired.
Github Clean	Code data from CodeParrot covering a variety of coding languages.
Pubmed Central	Biomedical and life sciences papers.
Free Law	Public-domain legal opinions from US federal and state courts.
SEC Filings	10-K/Q filings from the US Securities and Exchange Commission (SEC) for the years 1934-2022.
Patents	US patents granted from 1975 to May 2023, excluding design patents.
DeepMind Mathematics	Mathematical question and answer pairs data.
Earning Calls Transcript	Transcripts from the quarterly earnings calls that companies hold with investors. The dataset reports a collection of earnings call transcripts, the related stock prices, and the sector index.

Remarks:

- The level of detail differs from vendor to vendor and even from model to model. In general, this information is useful in helping data scientists (less so casual users).
- Clients should always review the information on models they intend to use.
- One differentiating factor for IBM models is the amount of information available. IBM is transparent on its models, so typically, a wealth of information is included in IBM's model cards.

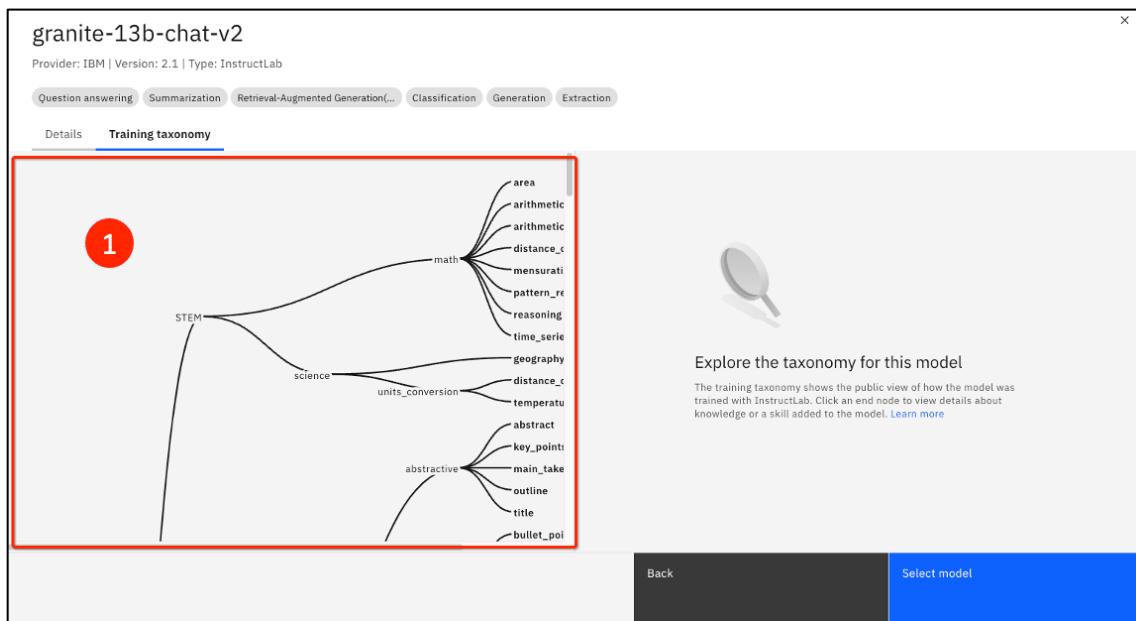
5.1.1 Taxonomy information

IBM watsonx.ai provides several InstructLab-trained models. As of Oct 2024, they are:

- granite-13b-chat-v2
- granite-20b-multilingual

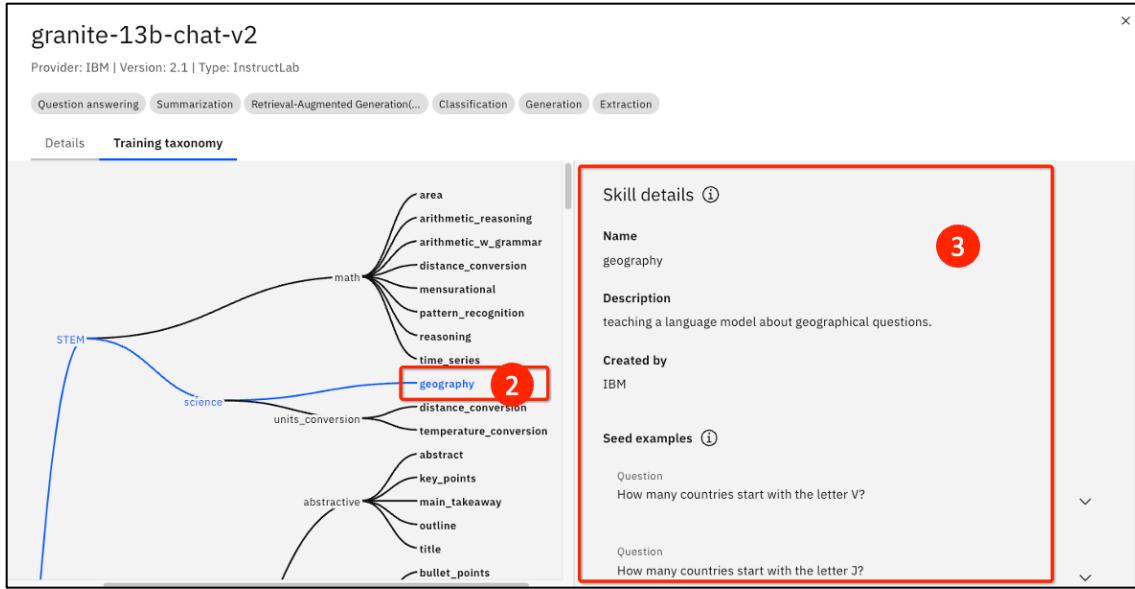
Additional information is provided for these models, namely, the taxonomy tree that reveals more specifics about the model.

1. If you have not closed the model card, click on the **Training taxonomy** tab. If you have closed the model card, then follow Steps 1-3 or Section 3.1 to pick the **granite-13b-chat-v2** model and then click on **Training taxonomy**. You will see a detailed tree of the taxonomy of training data used to train the model.



Note: To view the full tree, hover over the tree and scroll.

2. To learn more details, click an end node on the tree (which represents a skill or knowledge).
 - a. **Knowledge:** Data and facts that a model needs to more accurately answer questions. This is information available to the model.
 - b. **Skill:** Teaches the model to do something – interpretation of the information.
3. When you click on a knowledge or skill, the right side panel shows more details such as the skill name, skill description, entity that created this skill, and seed examples for this skill in InstructLab.



Note: Seed examples are given to InstructLab to synthetically generate new data to further fine-tune the model.

4. To view the details of a seed example, click on the seed example **question**.

Note that this only shows samples of seed examples used to train the model. Many more data points are generated to train a model.

The taxonomy displays the type of skills and knowledge that the model was trained on. Since models are highly dependent on the data used to train and fine-tune them, the taxonomy serves two main purposes:

- Allows a client to assess whether the model was trained to handle the use case they have in mind.
- Guides a client to provide additional training data for existing entries or any skills/knowledge gaps if the client wishes to train the model.

The screenshot shows the "Seed examples" section of the InstructLab interface. It contains three entries, each with a red border and a red circle labeled "4" to its right. The first entry is "Question: How many countries start with the letter V?", the second is "Question: How many countries start with the letter J?", and the third is "Question: How many countries start with the letter W?". Each entry has a small downward arrow to its right, indicating it can be collapsed.

5. To hide a seed example's details, click the question again.

Seed examples ⓘ

Question
How many countries start with the letter V?

Answer
Here are few countries that starts with letter "V" - Vanuatu. - Vatican City. - Venezuela. - Vietnam.

Question
How many countries start with the letter J?

Question
How many countries start with the letter W?

21. Now click the **Back** button to go back to the main model selection page.

granite-13b-chat-v2

Provider: IBM | Version: 2.1 | Type: InstructLab

Question answering Summarization Retrieval-Augmented Generation... Classification Generation Extraction

Details Training taxonomy

Skill details ⓘ

Name
geography

Description
teaching a language model about geographical questions.

Created by
IBM

Seed examples ⓘ

Question
How many countries start with the letter V?

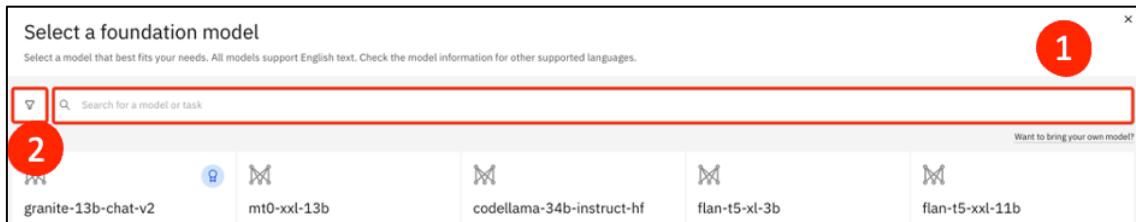
Question
How many countries start with the letter J?

Back 6 **Select model**

5.1.2. Filtering by model types

You can search for models using their name or use case through the search bar. In this section, you will search for multi-modal models and InstructLab models.

1. IBM watsonx.ai includes multi-modal models. As of October 2024, these include models that can work with images (text handling is implicitly included). On the **Select a foundation model** page, you can search for a model by name (if known).
2. You can also filter models by clicking the filter icon next to the **Search for a model or task** filter.



- After clicking the filter icon, a drop-down appears where you can filter the models using:

- Their intended **Task** (use case)
- Their **Provider**
- Their **Modalities**
- Their **Model type**

Note: As the number of models increases, filtering helps users focus on the models they should consider.

The screenshot shows the 'Select a foundation model' interface with the filter dropdown open. The dropdown is highlighted with a red box and a red circle labeled '3'. It contains four sections: 'Task', 'Provider', 'Modalities', and 'Model type'. The 'Model type' section has a radio button for 'All' selected. To the right of the dropdown, there is a grid of model cards. Each card includes a small icon, the model name, a brief description, and details about the provider and type. A link 'Want to bring your own model?' is located at the top right of the main content area.

Model	Description	Provider	Type
granite-13b-chat-v2	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	InstructLab
mt0-xxl-13b	An instruction-tuned iteration on mT5.	BigScience	Provided model
codellama-34b-instruct-hf	Code Llama is an AI model built on top of Llama 2, fine-tuned for generating and discussing code.	Code Llama	Provided model
flan-t5-xl-3b	A pretrained T5 - an encoder-decoder model pre-trained on a mixture of supervised / unsupervised tasks converted into a text-to-text format.	Google	Provided model
flan-t5-xxl-11b	flan-t5-xxl is an 11 billion parameter model based on the Flan-T5 family.	Google	Provided model
flan-ul2-20b	flan-ul2 is an encoder-decoder model based on the T5 architecture and instruction-tuned using the Fine-tuned Language Net.	Google	Provided model
granite-13b-instruct-v2	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-20b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-20b-multilingual	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	InstructLab
granite-34b-code-instruct			
granite-7b-lab			

- Click **Image** in the **Modalities** section.
- IBM watsonx.ai displays the models available for Image and Text. As of October 2024, there are four llama models in this collection (note below that the llama3-llava-next-8b-hf model is marked for deprecation). You will work with one of these models later in this lab.

Select a foundation model

To choose a model, review characteristics such as tasks that models perform. Compare model benchmarks with scores in the range 0–100. Higher scores are better.

All models Model benchmarks

Filter Search for a model or task

Task

- Chat (22)
- Question answering (24)
- Summarization (26)
- Retrieval-Augmented Generation(RAG) (22)
- Classification (26)
- Generation (26)
- Code generation and conversion (15)
- Extraction (26)

Provider

- BigScience (1)
- Code Llama (1)
- Google (3)
- IBM (8)
- Meta (12)
- Mistral AI (2)

Modalities

- Language (27)
- Image (4)

Want to bring your own model?

Image

 llama-3-2-11b-vision-instruct Llama-3-2-11b-vision-instruct is an auto-regressive language model that uses an optimized transformer architecture. Provider: Meta Type: Provided model	 llama-3-2-90b-vision-instruct Llama-3-2-90b-vision-instruct is an auto-regressive language model that uses an optimized transformer architecture. Provider: Meta Type: Provided model	 llama-guard-3-11b-vision Llama-guard-3-11b-vision is an auto-regressive language model that uses an optimized transformer architecture. Provider: Meta Type: Provided model
 llama3-llava-next-8b-hf Llama3-llava-next-8b-hf is an auto-regressive language model that uses an optimized transformer architecture. Provider: Meta Type: Provided model		

6. Now, uncheck Image and click InstructLab from the Model type filter.

Model type

- All (27)
- Provided model (24)
- InstructLab (3)

7. You will see the following (or similar) models listed:

Filter Search for a model or task

Task

- Chat (22)
- Question answering (24)
- Summarization (26)
- Retrieval-Augmented Generation(RAG) (22)
- Classification (26)
- Generation (26)
- Code generation and conversion (15)
- Extraction (26)

Provider

- BigScience (1)
- Code Llama (1)
- Google (3)
- IBM (8)
- Meta (12)
- Mistral AI (2)

Modalities

- Language (27)
- Image (4)

InstructLab

 granite-13b-chat-v2 The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: InstructLab	 granite-20b-multilingual The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: InstructLab
 granite-7b-lab The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: InstructLab	

Note that for the **Task**, **Provider**, and **Modalities** filters, you can select multiple options. However, you can only select one option from **All**, **Provided model**, or **InstructLab** from the **Models type** filter. The **Provided model** option and the **InstructLab** option are mutually exclusive.

8. You can use various combinations. For example, click:
 - **Provided model** from the **Model** type filter
 - Click **IBM** and **Mistral AI** from the **Provider** filter
 - Click **Chat** and **Generation** from the **Task** filter
9. You will see models that are provided, from either IBM or Mistral AI, supporting either Chat or Generation tasks listed on the right.

Select a foundation model

To choose a model, review characteristics such as tasks that models perform. Compare model benchmarks with scores in the range 0–100. Higher scores are better.

All models Model benchmarks

Filter X

Task

- Chat (22)
- Question answering (24)
- Summarization (26)
- Retrieval-Augmented Generation(RAG) (22)
- Classification (26)
- Generation (26)
- Code generation and conversion (15)
- Extraction (26)

Provider

- BigScience (1)
- Code Llama (1)
- Google (3)
- IBM (8)
- Meta (12)
- Mistral AI (2)

Modalities

- Language (27)
- Image (4)

Model type

- All (27)
- Provided model (24)
- InstructLab (3)

Search for a model or task

Chat X Generation X IBM X Mistral AI X Provided model X Reset filters

Want to bring your own model?

Model	Description	Provider	Type
granite-13b-instruct-v2	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-20b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-34b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-3b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-8b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
mistral-large	Mistral Large, the most advanced Large Language Model (LLM) developed by Mistral AI, is an exceptionally powerful model. Thanks to its state-of-the-art reasoning capabilities, it can be applied to any language-based task, including...	Mistral AI	Provided model
mixtral-8x7b-instruct-v01	The Mixtral-8x7B Large Language Model (LLM) is a pretrained generative Sparse Mixture of Experts.	Mistral AI	Provided model

10. It is important to keep in mind that these are model recommendations from watsonx.ai for best results, given the choices you have selected. There is nothing to stop you from using another model; for example, codellama-34b-instruct-hf for chat. It just means you are not likely to get the best results.

5.2 Why Model Choice Matters (Model Architecture)

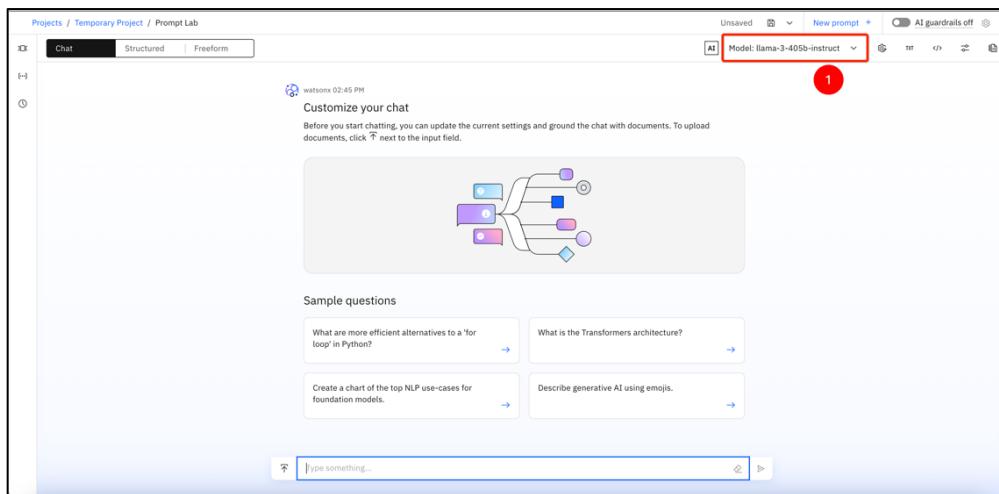
This section highlights the importance of selecting the right model. Each model comes with its own set of benefits and drawbacks, making the choice crucial for the success of a business.

For this example, you will be prompting different models to do a simple math question in the form of:

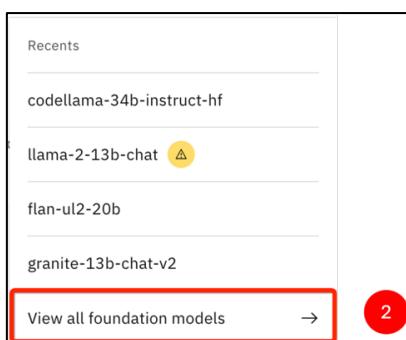
Answer the following question by reasoning step by step. The cafeteria had 23 apples. If they used 20 for lunch, and bought 6 more, how many apples do they have?

The first exercise will show how a model that was trained for doing Question answering and reasoning tasks performs for this prompt:

1. Click the model selection dropdown menu on the top right of the screen.



2. Click the **View all foundation models** button.



3. There may be a filter that is already applied. Click **Reset filters** to clear any filters.

Select a foundation model

Select a model that best fits your needs. All models support English text. Check the model information for other supported languages.

Chat Reset filters 3

Model	Description	Provider	Type
granite-13b-chat-v2	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	Provider: IBM	Type: InstructLab
granite-13b-instruct-v2	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	Provider: IBM	Type: Provided model
granite-20b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	Provider: IBM	Type: Provided model
granite-20b-multilingual	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	Provider: IBM	Type: InstructLab
granite-34b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	Provider: IBM	Type: Provided model
granite-3b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	Provider: IBM	Type: Provided model
granite-7b-lab	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	Provider: IBM	Type: InstructLab
granite-8b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	Provider: IBM	Type: Provided model
llama-2-13b-chat		Provider: Google	Type: Provided model
llama-2-70b-chat		Provider: Google	Type: Provided model
llama-3-1-70b-instruct		Provider: Google	Type: Provided model
llama-3-1-8b-instruct		Provider: Google	Type: Provided model

4. Either search or scroll through the model list and find the **flan-ul2-20b** model tile and click it.

Select a foundation model

Select a model that best fits your needs. All models support English text. Check the model information for other supported languages.

Chat Reset filters 4

Model	Description	Provider	Type
granite-13b-chat-v2	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	Provider: IBM	Type: InstructLab
mt0-xxl-13b	An instruction-tuned iteration on mT5.	Provider: BigScience	Type: Provided model
codellama-34b-instruct-hf	Code Llama is an AI model built on top of Llama 2, fine-tuned for generating and discussing code.	Provider: Code Llama	Type: Provided model
flan-t5-xl-3b	A pretrained T5 - an encoder-decoder model pre-trained on a mixture of supervised / unsupervised tasks converted into a text-to-text format.	Provider: Google	Type: Provided model
flan-t5-xxl-11b	Flan-T5-xxl is an 11 billion parameter model based on the Flan-T5 family.	Provider: Google	Type: Provided model
flan-ul2-20b	Flan-Ul2 is an encoder-decoder model based on the T5 architecture and instruction-tuned using the Fine-tuned Language Net.	Provider: Google	Type: Provided model
granite-13b-instruct-v2	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	Provider: IBM	Type: Provided model
granite-20b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	Provider: IBM	Type: Provided model
granite-20b-multilingual		Provider: IBM	Type: Provided model
granite-34b-code-instruct		Provider: IBM	Type: Provided model
granite-3b-code-instruct		Provider: IBM	Type: Provided model
granite-7b-lab		Provider: IBM	Type: Provided model

5. Looking at the tags for this model, note that it is tagged with the **Question answering** tag, which means that this model is good for reasoning tasks based on a question.

flan-ul2-20b

Provider: Google | Type: Provided model

Question answering Summarization Retrieval-Augmented Generation(...) Classification Generation Extraction 5

Model card for Flan-UL2

- If you scroll down to the model card and look at the Abstract for this model's paper, you can see that this model achieves state-of-the-art (SOTA) performance on a wide variety of tasks, including question answering, common sense reasoning, and long text reasoning.

flan-ul2-20b

Provider: Google | Type: Provided model

Question answering Summarization Retrieval-Augmented Generation(… Classification Generation Extraction

Task Paradigms

Abstract

Existing pre-trained models are generally geared towards a particular class of problems. To date, there seems to be still no consensus on what the right architecture and pre-training setup should be. This paper presents a unified framework for pre-training models that are universally effective across datasets and setups. We begin by disentangling architectural archetypes with pre-training objectives -- two concepts that are commonly conflated. Next, we present a generalized and unified perspective for self-supervision in NLP and show how different pre-training objectives can be cast as one another and how interpolating between different objectives can be effective. We then propose Mixture-of-Denoisers (MoD), a pre-training objective that combines diverse pre-training paradigms together. We furthermore introduce a notion of mode switching, wherein downstream fine-tuning is associated with specific pre-training schemes. We conduct extensive ablative experiments to compare multiple pre-training objectives and find that our method pushes the Pareto-frontier by outperforming T5 and/or GPT-like models across multiple diverse setups. Finally, by scaling our model up to 20B parameters, we achieve SOTA performance on 50 well-established supervised NLP tasks ranging from language generation (with automated and human evaluation), language understanding, text classification, question answering, commonsense reasoning, long text reasoning, structured knowledge grounding and information retrieval. Our model also achieves strong results at in-context learning, outperforming 175B GPT-3 on zero-shot SuperGLUE and tripling the performance of T5-XXL on one-shot summarization.

- Now click the blue **Select model** button on the bottom right.

flan-ul2-20b

Provider: Google | Type: Provided model

Question answering Summarization Retrieval-Augmented Generation(… Classification Generation Extraction

Model card for Flan-UL2

Inputs-to-targets "Autoregressive" models

- Decoder-only PrefixLM
- OR
- Encoder-Decoder

X-denoiser (long spans & low corruption)

X-denoiser (long spans & high corruption)

X-denoiser (short spans & high corruption)

X-denoiser (extreme denoising)

R-denoiser (short spans & low corruption)

S-denoiser (sequential denoising / prefix language modeling)

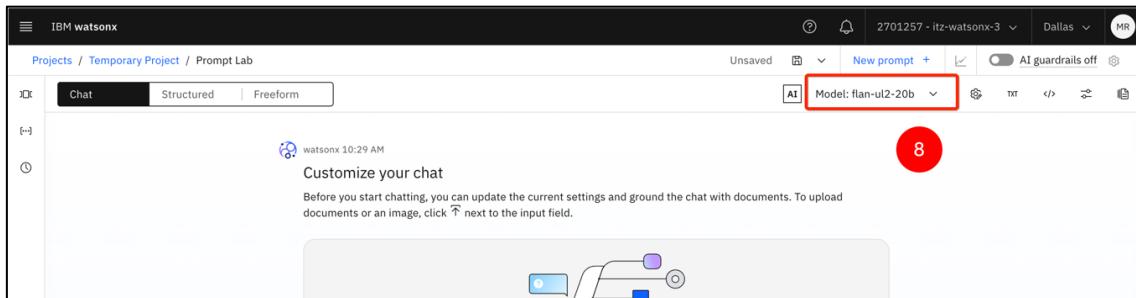
Learning Paradigms

- Supervised Finetuning
- In-context Learning
- Zero-Shot
- Language Generation
- Language Understanding
- Structured Knowledge Grounding

7

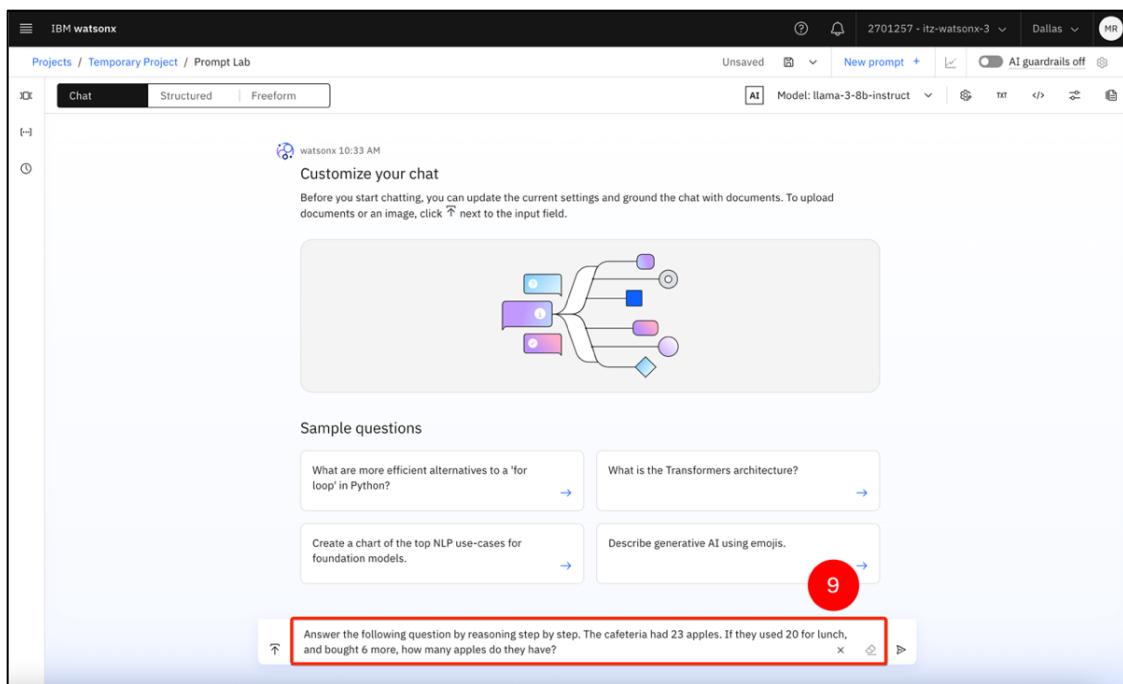
Back Select model

- Double-check that the model selected in the dropdown menu is now the **flan-ul2-20b** model. If it is then continue to the next step. If it displays another model's name, please go back to Step 1 and select the correct model.



- Copy this prompt and paste it into the chat bar on the bottom of the page in the **Type something...** field

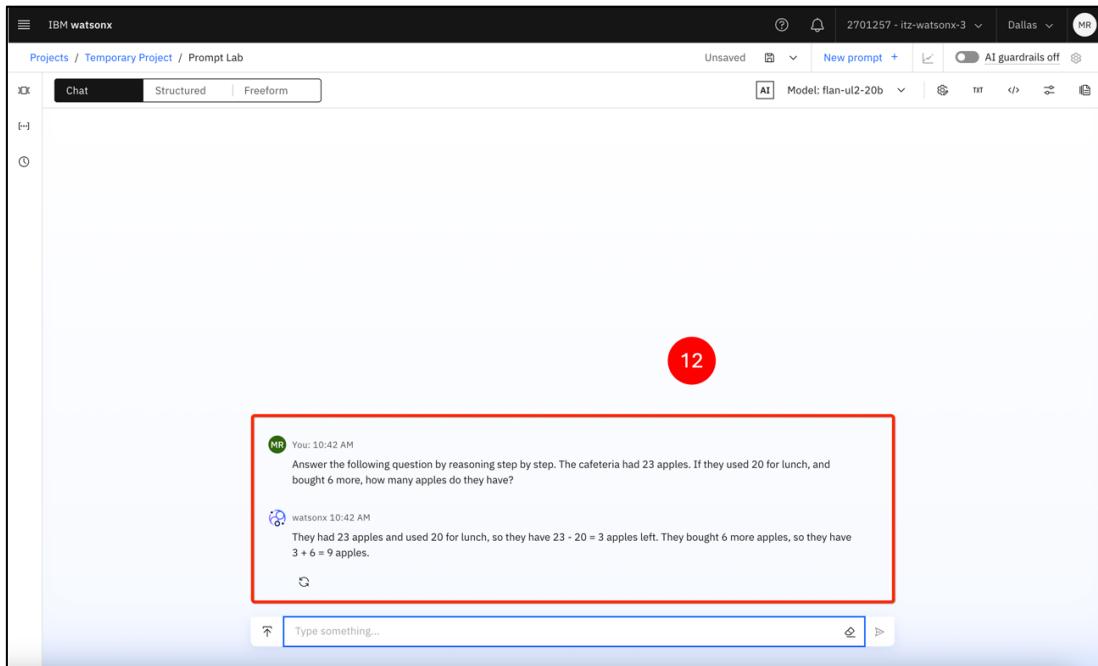
Answer the following question by reasoning step by step. The cafeteria had 23 apples. If they used 20 for lunch, and bought 6 more, how many apples do they have?



- Click the send icon (\triangleright) to send the prompt to the model and wait for a response

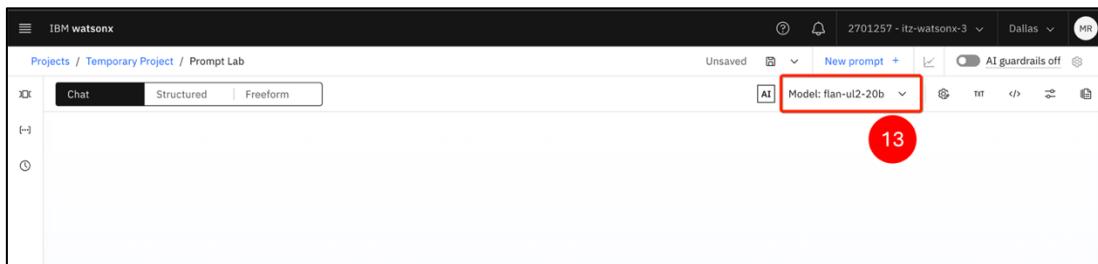


- Looking at the response, the chosen model performed well on this simple mathematical reasoning problem. It clearly showed the model was working in each step and arrived at the correct answer of 9 apples remaining.

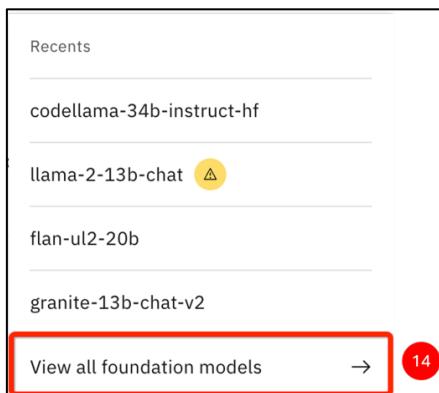


Now let's look at how a model that was NOT trained for doing Question answering and reasoning tasks performs for this prompt:

12. Click the model selection drop-down menu on the top right of the screen.



13. Click the **View all foundation models** button.



14. There may be a filter that is already applied. Click **Reset filters** to clear any filters.

Select a foundation model

Select a model that best fits your needs. All models support English text. Check the model information for other supported languages.

Chat Reset filters 14

 granite-13b-chat-v2 The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: InstructLab	 granite-13b-instruct-v2 The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: Provided model	 granite-20b-code-instruct The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: Provided model	 granite-20b-multilingual The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: InstructLab
 granite-34b-code-instruct The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: Provided model	 granite-3b-code-instruct The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: Provided model	 granite-7b-lab The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: InstructLab	 granite-8b-code-instruct The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: Provided model
 llama-2-13b-chat	 llama-2-70b-chat	 llama-3-1-70b-instruct	 llama-3-1-8b-instruct

Want to bring your own model?

15. Now either scroll or filter through the models and click on the **codellama-34b-instruct-hf** tile.

Select a foundation model

Select a model that best fits your needs. All models support English text. Check the model information for other supported languages.

15

 granite-13b-chat-v2 The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: InstructLab	 mt0-xxl-13b An instruction-tuned iteration on mT5. Provider: BigScience Type: Provided model	 codellama-34b-instruct-hf Code Llama is an AI model built on top of Llama 2, fine-tuned for generating and discussing code. Provider: Code Llama Type: Provided model	 flan-t5-xl-3b A pretrained T5 - an encoder-decoder model pretrained on a mixture of supervised / unsupervised tasks converted into a text-to-text format. Provider: Google Type: Provided model
 flan-t5-xxl-11b flan-t5-xxl is an 11 billion parameter model based on the Flan-T5 family. Provider: Google Type: Provided model	 flan-ul2-20b flan-ul2 is an encoder decoder model based on the T5 architecture and instruction-tuned using the Fine-tuned Language Net. Provider: Google Type: Provided model	 granite-13b-instruct-v2 The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: Provided model	 granite-20b-code-instruct The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: Provided model

16. Looking at this model's tags, note that this model is good for **Code generation and conversion**, which seems at first glance to match this example use case.

codellama-34b-instruct-hf

Provider: Code Llama | Type: Provided model

16

Code generation and conversion

Note: This illustrates the importance of understanding the task at hand and what a model was trained for. Coding is very much related to math. However, it does NOT mean it's good at chatting and reasoning from a Natural Language (NL) prompt involving a math problem. Codellama will perform well when the prompt includes instructions to generate code or code input for conversation. This prompt, however, asks for neither. This prompt asks a question that requires the model to understand the NL flow and reason from it. This task does not involve writing code so much as understanding and executing a “reasoning workflow” based on the input. You will select the model and see how this model responds to the same prompt in the chat.

17. Look at the model description, and note how this model is specifically designed for **code synthesis and understanding**. Therefore intuitively, this model shouldn't perform well on a mathematical reasoning Question answering task as in this example. You will try it now.

	Base Model	Python	Instruct
7B	codellama/CodeLlama-7b-hf	codellama/CodeLlama-7b-Python-hf	codellama/CodeLlama-7b-Instruct-hf
13B	codellama/CodeLlama-13b-hf	codellama/CodeLlama-13b-Python-hf	codellama/CodeLlama-13b-Instruct-hf
34B	codellama/CodeLlama-34b-hf	codellama/CodeLlama-34b-Python-hf	codellama/CodeLlama-34b-Instruct-hf
70B	codellama/CodeLlama-70b-hf	codellama/CodeLlama-70b-Python-hf	codellama/CodeLlama-70b-Instruct-hf

18. Click Select model.

To use this model, please make sure to install transformers:

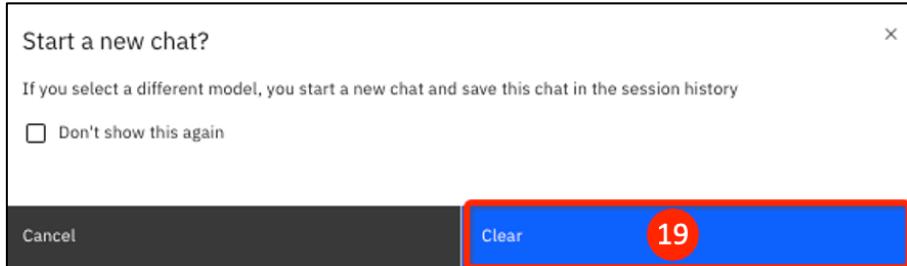
```
pip install transformers accelerate
```

Model capabilities:

18

Select model

19. When you switch models when you are in a chat, you will need to start a new chat. You will see the following dialog box. Click **Clear** to clear the current chat and start a new one with the model you have just selected.



20. Note that the previous prompt and results are cleared. Now use the same prompt (included below) you used in Step 9 and paste it into the **Type something...** field.

Answer the following question by reasoning step by step. The cafeteria had 23 apples. If they used 20 for lunch, and bought 6 more, how many apples do they have?

21. Click the send icon (>) to send the prompt to the model and wait for a response

22. Looking at the results, this completion completely missed out on the fact that the cafeteria used 20 apples for lunch and only based its calculations on how many apples the cafeteria had in the beginning and how many apples the cafeteria bought; therefore it arrived at the wrong conclusion that the cafeteria has 29 apples and as you know from Step 11 the correct answer is 9 apples.

 You: 11:05 AM

Answer the following question by reasoning step by step. The cafeteria had 23 apples. If they used 20 for lunch, and bought 6 more, how many apples do they have?

 watsonx 11:05 AM

1. The cafeteria had 23 apples.
2. They used 20 apples for lunch.
3. They bought 6 more apples.
4. The cafeteria now has $23 + 6 = 29$ apples.

Therefore, the cafeteria has 29 apples.



Note: Although this model performed poorly on this use case it does not mean that the model is bad. It only showed that this model isn't suitable for this type of task. If you asked the same two models a code generation or translation-related question, the **codellama-34b-instruct-hf** would likely outperform the **flan-ul2-20b** model.

As different models are specialized for different use cases, model choice is essential to support different clients' use cases.

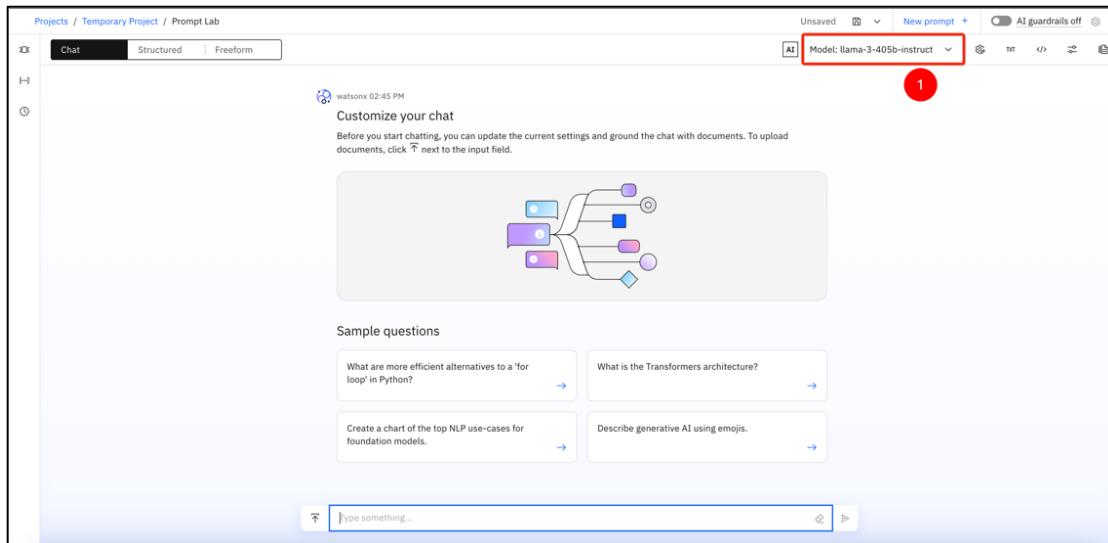
5.3 Why Model Choice Matters (Model Training)

This section emphasizes the importance of model choice in relation to model training. When it comes to choosing a model, it's essential to consider the type of model that best suits your needs. There are two primary types of text-based models: instruction-tuned and chat-aligned. The key differences between these models lie in their training objectives and the type of responses they generate.

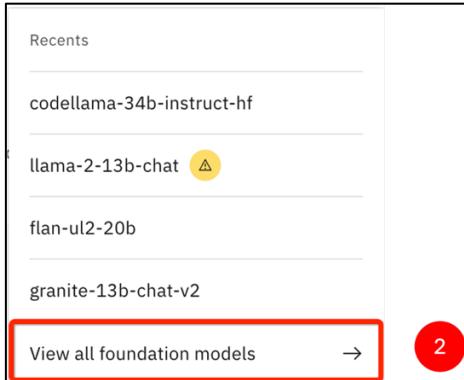
- **Instruction-tuned:** These base models are designed to complete tasks efficiently, without the need for human-like conversation or extraneous language. They provide direct, task-focused responses without additional padding or formatting.
- **Chat aligned:** These models are further trained on human-provided data using Reinforcement Learning. This training enables them to generate more natural human-like responses to instructions, making them ideal for chatbot applications or any scenario where human-like interaction is necessary.

You can use any model in the watsonx.ai Chat mode, but chat-aligned models typically perform better in chat-type tasks. In this section, you will choose two models, one larger granite model, and a smaller granite 3.0 model, to handle a complex task and see how each model performs.

1. Click the model selection dropdown menu on the top right of the screen.



2. Click View all foundation models.



3. There may be a filter that is already applied. Click **Reset filters** to clear any filters.

Select a foundation model

Select a model that best fits your needs. All models support English text. Check the model information for other supported languages.

Search for a model or task

Chat Reset filters

granite-13b-chat-v2 The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: InstructLab	granite-13b-instruct-v2 The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: Provided model	granite-20b-code-instruct The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: Provided model	granite-20b-multilingual The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: InstructLab
---	--	--	--

4. Either scroll or filter through the models to find the **granite-13b-instruct-v2** tile and click on it.

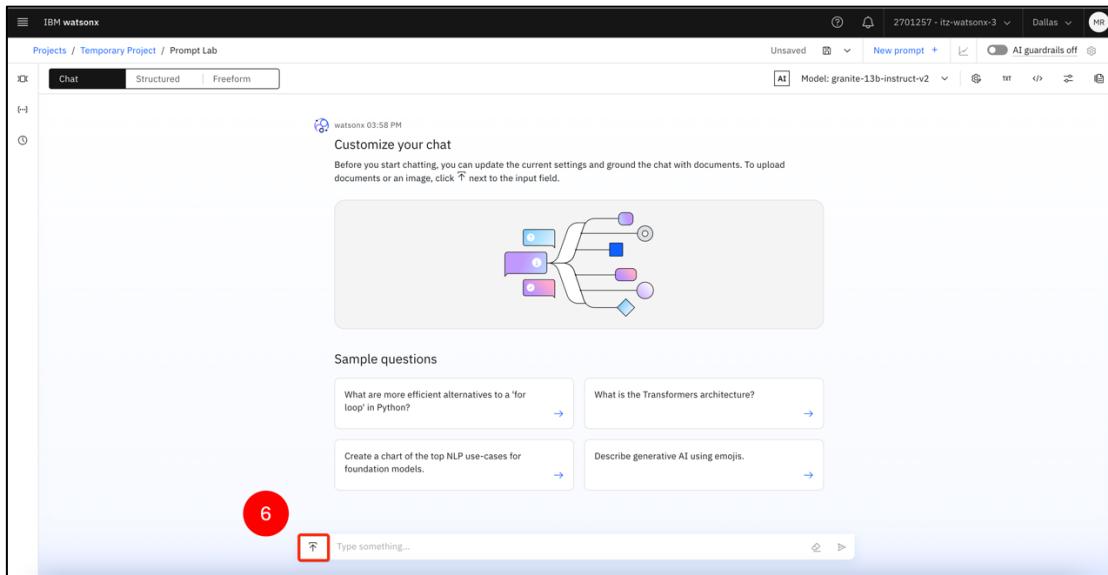
Select a foundation model

Select a model that best fits your needs. All models support English text. Check the model information for other supported languages.

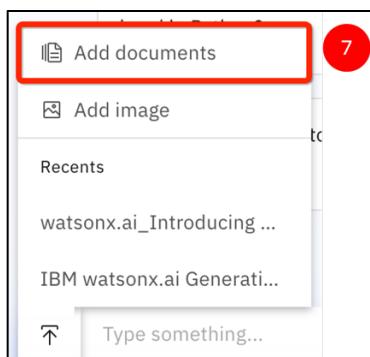
Search for a model or task

granite-13b-chat-v2 The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: InstructLab	m10-xxl-13b An instruction-tuned iteration on mT5. Provider: BigScience Type: Provided model	codellama-34b-instruct-hf Code Llama is an AI model built on top of Llama 2, fine-tuned for generating and discussing code. Provider: Code Llama Type: Provided model	flan-t5-xl-3b A pretrained T5 - an encoder-decoder model pre-trained on a mixture of supervised / unsupervised tasks converted into a text-to-text... Provider: Google Type: Provided model	flan-t5-xxl-11b flan-t5-xxl is an 11 billion parameter model based on the Flan-T5 family. Provider: Google Type: Provided model
flan-ul2-20b flan-ul2 is an encoder decoder model based on the T5 architecture and instruction-tuned using the Fine-tuned Language Net. Provider: Google Type: Provided model	granite-13b-instruct-v2 The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: Provided model	flan-t5e-20b-code-instruct The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: Provided model	granite-20b-multilingual The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: InstructLab	granite-34b-code-instruct The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks. Provider: IBM Type: Provided model
granite-3b-code-instruct	granite-7b-lab	granite-8b-code-instruct	llama-3-1-70b-instruct	llama-3-1-8b-instruct

5. Go to this link: <https://ibm.seismic.com/Link/Content/DC6cmQRjgHp8BGWTmb37gmgX3XHd> and download the Granite Foundation Models PDF.
6. Click the **Upload documents** button (↑) on the left side of the **Type something...** field located at the bottom.



7. Click the **Add documents** button.



8. Ensure that the **In memory** option on the left side tab is selected.
9. Click the **Browse** button.
10. Choose the file that you downloaded in Step 5 and click **Open** (for MacOS, you may have **Upload** instead).

Ground gen AI with vectorized documents

Add documents to vectorize and create a vector index in memory. Otherwise, select your vector database and specify index details.

Vector stores

- In memory** 8
- watsonx.data Milvus
- Elasticsearch

Add files

Drop data files or browse to upload

Add PPTX, DOCX, PDF, or TXT files or select from project.

Add up to 300 MB with PPTX files, 50 MB with PDF files, 10 MB with DOCX files, or 5 MB with TXT and other files. Max file size is the lowest limit for the included file types.

Browse
Select from project
9

Define details

Name

Description (optional)

What's the purpose of this vector index?

Advanced settings

Cancel
Create

11. Ensure the file you uploaded appears below the **Add files** box

12. Click **Create**.

Ground gen AI with vectorized documents

Add documents to vectorize and create a vector index in memory. Otherwise, select your vector database and specify index details.

Vector stores

- In memory** 8
- watsonx.data Milvus
- Elasticsearch

Add files

Drop data files or browse to upload

Add PPTX, DOCX, PDF, or TXT files or select from project.

Add up to 300 MB with PPTX files, 50 MB with PDF files, 10 MB with DOCX files, or 5 MB with TXT and other files. Max file size is the lowest limit for the included file types.

Browse
Select from project

granite-foundation-models.pdf
5.7MB
11
🔗
trash

Define details

Name

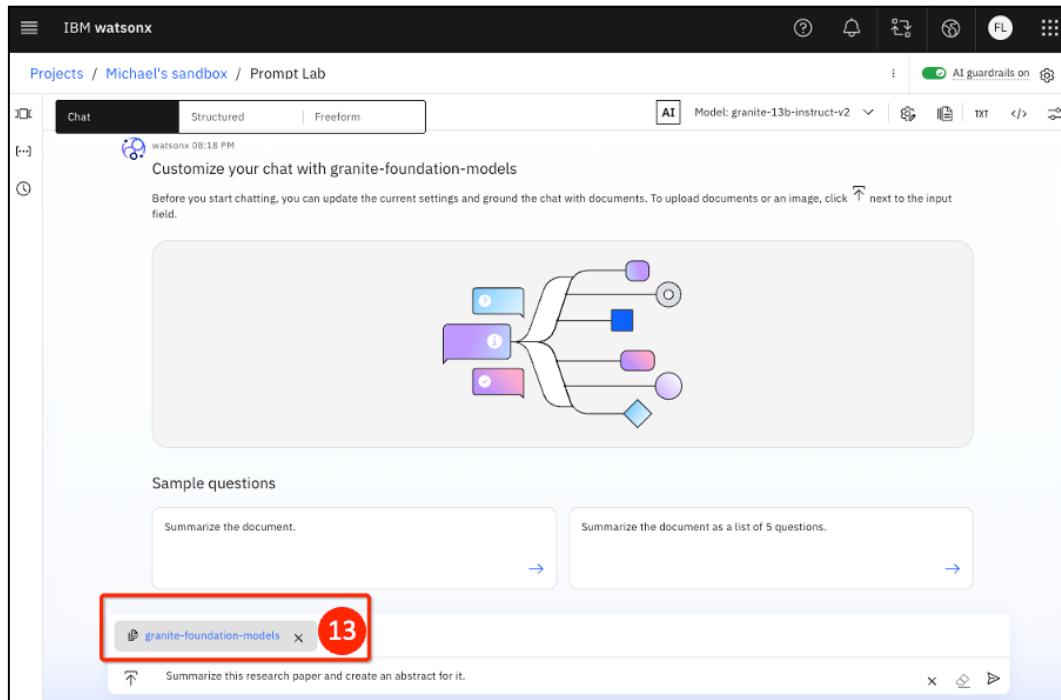
Description (optional)

What's the purpose of this vector index?

Advanced settings

Cancel
Create 12

13. The file you uploaded should now appear right above the chat when it is finished processing.



14. Copy the prompt below, and paste it into the Type something... field.

Summarize this research paper and create an abstract for it.



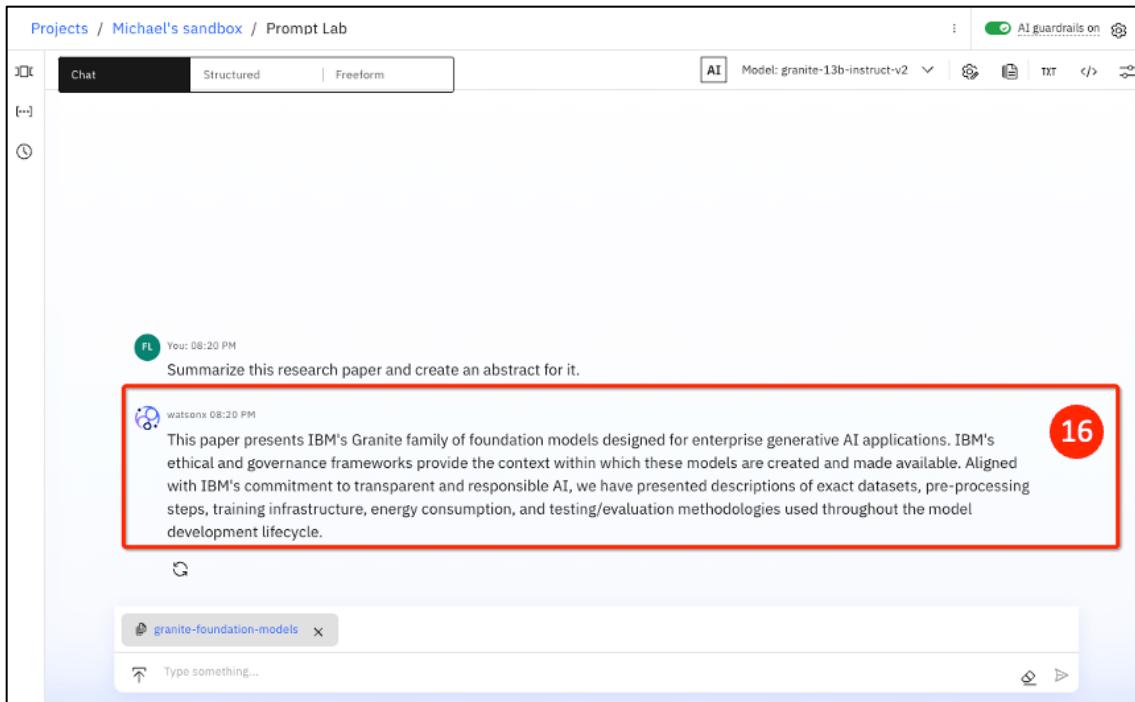
15. Click either the **Enter** key on your keyboard or the ➤ icon on the right of the **Type something...** field.



16. This is the output:

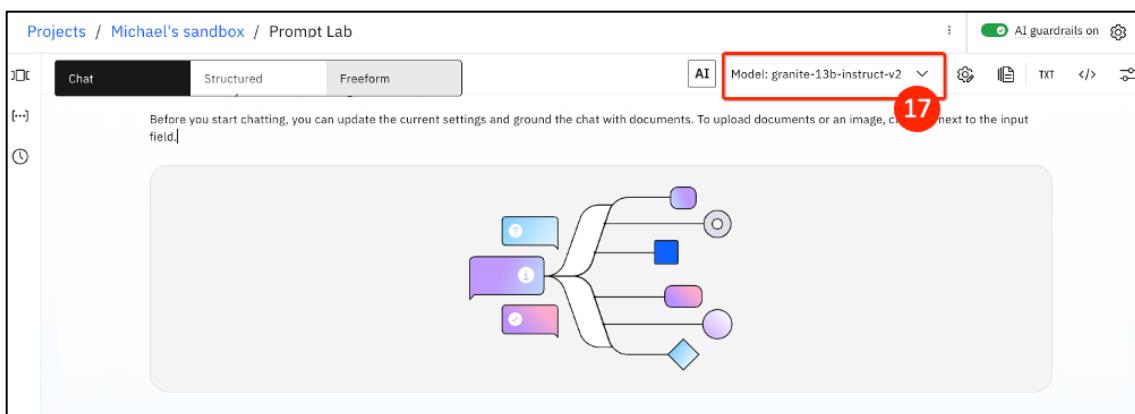
This paper presents IBM's Granite family of foundation models designed for enterprise generative AI applications. IBM's ethical and governance frameworks provide the context within which these models are created and made available. Aligned with IBM's commitment to transparent and responsible AI, we have presented descriptions of exact datasets, pre-processing steps, training infrastructure, energy consumption, and testing/evaluation methodologies used throughout the model development lifecycle.

Looking at the response, you can see that it is straight to the point, directly returning a generated abstract for the research paper. This is the one characteristic of instruction-tuned models. As the name suggests, they are trained to only follow instructions and do the instruction at hand. They provide a very terse summary, and it does not do justice to the granite model paper.

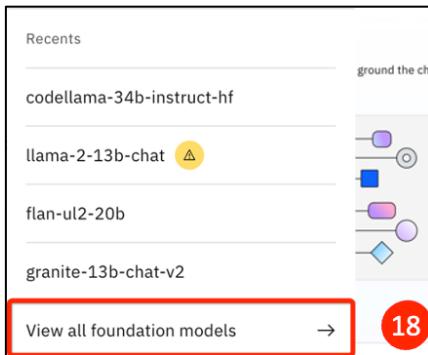


Now, you will compare the response of a chat-aligned model to the output of the instruct-trained model to showcase the differences in responses between the two models.

17. Click the model selection dropdown menu on the top right of the screen.



18. Click the **View all foundation models** button.



19. There may be a filter that is already applied. Click **Reset filters** to clear any filters.

Model	Description	Provider	Type
granite-13b-chat-v2	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	InstructLab
granite-13b-instruct-v2	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-20b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-20b-multilingual	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	InstructLab

20. Either scroll or filter through the models to find the **granite-3-8b-instruct** model tile and click it.

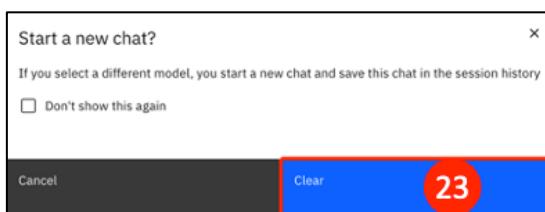
Model	Description	Provider	Type
granite-13b-chat-v2	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	InstructLab
granite-13b-instruct-v2	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-20b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-20b-multilingual	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	InstructLab
granite-3-2b-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-3-8b-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-34b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-3b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-7b-lab	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	InstructLab
granite-8b-code-instruct	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-guardian-3-2b	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model
granite-guardian-3-8b	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.	IBM	Provided model

21. Most chat-aligned models have the word “chat” in their name. However, the model name is not the best place to determine model capabilities. When the model card is opened, you will see that the **granite-3-8b-instruct** model is “... developed using a diverse set of techniques with a structured chat format, ...”

22. Click **Select model**.

The screenshot shows the model card for **Granite-3.0-8B-Instruct**. At the top, it says "Provider: IBM | Version: 1.0 | Type: Provided model". Below that is a horizontal bar with tabs: Question answering, Summarization, Classification, Generation, Extraction, and Function calling. A note at the top states: "Note: By using the model you agree to these terms." There is a "Read terms" link and a close button. The main content area starts with "Granite-3.0-8B-Instruct" and describes it as the first release of a set of releases. It mentions that the model will be expanded to 128k context length and receive multi-modal capabilities. The "Model Summary" section highlights that the model is a 8B parameter model finetuned from **Granite-3.0-8B-Base** using a combination of open source instruction datasets with permissive license and internally collected synthetic datasets. A red box highlights the sentence: "This model is developed using a diverse set of techniques with a structured chat format, including supervised finetuning, model alignment using reinforcement learning, and model merging." Below this, there is a list of details: Developers (Granite Team, IBM), GitHub Repository ([ibm-granite/granite-3.0-language-models](#)), Website ([Granite Docs](#)), Paper ([Granite 3.0 Language Models](#)), Release Date (October 21st, 2024), and License (Apache 2.0). The "Supported Languages" section lists English, German, Spanish, French, Japanese, Portuguese, Arabic, Czech, Italian, Korean, Dutch, and Chinese. The "Intended use" section states that the model is designed to respond to general instructions and can be used to build AI assistants for multiple domains, including business applications. At the bottom right, there are "Back" and "Select model" buttons. The "Select model" button is highlighted with a red box and the number 22.

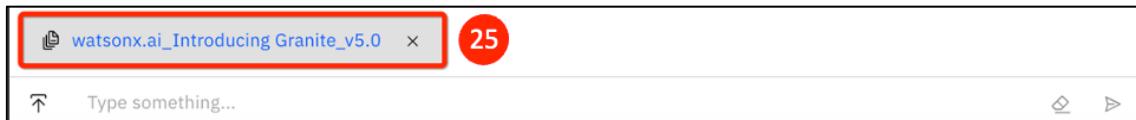
23. As you did before (Step 19 of Section 5.2), when you use a new model, you will need to clear the current chat. You will receive this dialog box. Click **Clear**.



24. Ensure that the correct model is selected on the top right model dropdown, if the model isn't **granite-3-8b-instruct**, repeat the process from Step 1

The screenshot shows the **Prompt Lab** interface. At the top, it says "Projects / Michael's sandbox / Prompt Lab". The toolbar includes "Unsaved", "AI guardrails on" (with a green switch), and other icons. The main area shows a list of chat types: Chat (selected), Structured, and Freeform. On the right, there is a dropdown menu labeled "Model: granite-3-8b-instruct". A red box highlights this dropdown and the number 24. Below the dropdown, there is a message from "watsonix 09:22 AM" and a "Customize your chat with granite-foundation-models" section. A note at the bottom says: "Before you start chatting, you can update the current settings and ground the chat with documents. To upload documents or an image, click ↑ next to the input field." A large input field is visible at the bottom.

25. The file should be carried over from the previous chat (note the name of the file above the Type something... field. If not, the process of uploading the file is the same as before.



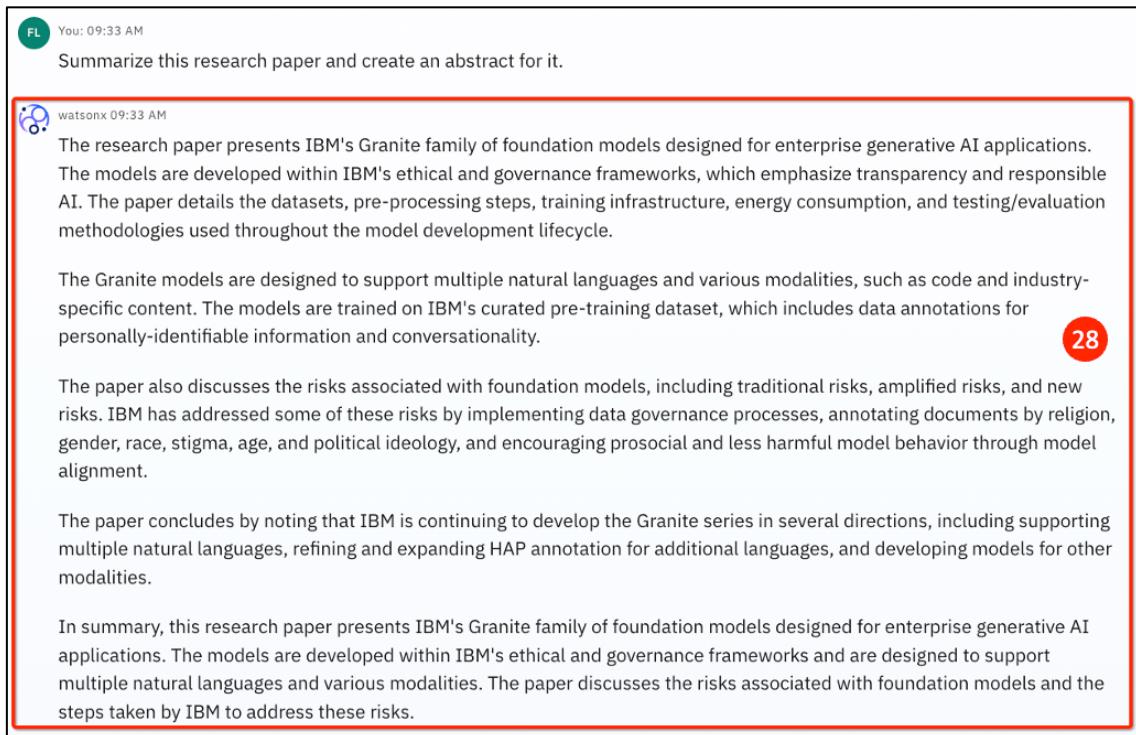
26. Use the same prompt by pasting it into the Type something... field.

Summarize this research paper and create an abstract for it.

27. Click either the **Enter** key on your keyboard or the icon on the right of the Type something... field.



28. The response here is more detailed, resembling a chat agent. The **granite-3-8b-instruct** model is designed to engage users more interactively. In addition to completing the task at hand, they provide extra context and insights, making the conversation less robotic and creating a more human-like interaction. All of this makes it easier for users to connect and understand the information being shared.



29. You will not need the **granite-foundation-models** PDF in the next section. Click on the X next to the PDF file to remove it.



Section Summary

In this section, you performed a summarization task in chat mode using two different instruct-tuned models:

- An earlier granite model. All models are capable of performing chat, and the **granite-13b-instruct-v2** model. This model performed reasonably well in the chat. It can follow the instructions and provide a good summary.
- A newer **granite-3-8b-instruct** model that is also good in chat. With this model, you obtained a more detailed output that reflected good chat-type behavior and the ability to follow instructions. This granite model is smaller and is designed to handle chat and instruct-based prompts.
- Clients with chat use cases can consider the Granite 3.0 models for such use cases.

Bonus exercise: You can also try using the **granite-13b-chat-v2** model (already marked for deprecation). This is an earlier chat-aligned model. You will find that it performs more like the **granite-3-8b-instruct** model.

5.4 Why Model Choice Matters (different instruct models)

Instruct models are models that are trained to follow a prompt's instructions. These models are useful if you want to provide, for example, structure in the output. In this section, you will compare two instruct-trained models. In particular, you will compare a llama instruct model with one of the new granite 3.0 models.

1. Select the **Freeform** tab from the **Prompt Lab**.



2. Copy the following in the text field (include a carriage return at the end after the word **Summary** and paste it into the prompt window).

The following document is a transcript from a financial earnings call. Read the document and then write a summary in point form.

Document: Financial Highlights

I'll start with the financial highlights of the fourth quarter. We delivered \$8 billion in revenue, \$1.5 billion of operating pre-tax income, and operating earnings per share of \$1.5. In our seasonally strongest quarter, we generated \$2.5 billion of free cash flow. Our revenue for the quarter was up over three percent at constant currency. While the dollar weakened a bit from 90 days ago, it still impacted our reported revenue by over \$500 million – and 3 points of growth. Revenue growth this quarter was again broad-based. Product Sales revenue was up eight percent and Services up nine percent. These are our growth vectors and represent over 35 percent of our revenue. Within each of these segments, our growth was pervasive. We also had good growth across our geographies, with mid-single-digit growth or better in the Americas, EMEA, and Asia Pacific. And for the year, we gained shares overall. We had strong transactional growth in Product Sales to close the year. At the same time, our recurring revenue, which provides a solid base of revenue and profit, also grew. Earnings Prepared Remarks Earnings were up 8.5 percent from last year. Looking at our profit metrics for the quarter, we expanded the operating pretax margin by 85 basis points. This reflects a strong portfolio mix and improving Product and Consulting margins. These same dynamics drove a 30-basis point increase in operating gross margin. Our expense was down year to year, driven by currency dynamics. Within our base expense, the work we're doing to digitally transform our operations provides flexibility to continue to invest in innovation and in talent. Our operating tax rate was 7 percent, which is flat versus last year. And our operating earnings per share of \$1.5 was up over seven percent. Turning to free cash flow, we generated \$2.5 billion in the quarter and \$4.5 billion for the year. Our full-year free cash flow is up \$1 billion from last year.

Summary:

Projects / Felix's sandbox / Prompt Lab

Chat Structured Freeform AI Model: flan-ul2-20b {#} TXT </> ⚙️

The following document is a transcript from a financial earnings call. Read the document and then write a summary in point form.

Document: Financial Highlights

I'll start with the financial highlights of the fourth quarter. We delivered \$8 billion in revenue, \$1.5 billion of operating pre-tax income and operating earnings per share of \$1.5. In our seasonally strongest quarter, we generated \$2.5 billion of free cash flow. Our revenue for the quarter was up over three percent at constant currency. While the dollar weakened a bit from 90 days ago, it still impacted our reported revenue by over \$500 million – and 3 points of growth. Revenue growth this quarter was again broad based. Product Sales revenue was up eight percent and Services up nine percent. These are our growth vectors and represent over 35 percent of our revenue. Within each of these segments, our growth was pervasive. We also had good growth across our geographies, with mid-single digit growth or better in Americas, EMEA and Asia Pacific. And for the year, we gained share overall. We had strong transactional growth in Product Sales to close the year. At the same time, our recurring revenue, which provides a solid base of revenue and profit, also grew. Earnings Prepared Remarks Earnings were up 8.5 percent from last year. Looking at our profit metrics for the quarter, we expanded operating pretax margin by 85 basis points. This reflects a strong portfolio mix and improving Product and Consulting margins. These same dynamics drove a 30-basis point increase in operating gross margin. Our expense was down year to year, driven by currency dynamics. Within our base expense, the work we're doing to digitally transform our operations provides flexibility to continue to invest in innovation and in talent. Our operating tax rate was 7 percent, which is flat versus last year. And our operating earnings per share of \$1.5 was up over seven percent. Turning to free cash flow, we generated \$2.5 billion in the quarter and \$4.5 billion for the year. Our full-year free cash flow is up \$1 billion from last year. As we talked about all year, we have a few drivers of our free cash flow growth. We had working capital improvements driven by efficiencies in operations. Despite strong collections, the combination of revenue performance above our model and the timing of the transactions in the quarter led to higher-than-expected working capital at the end of the year. This impacted our free cash flow performance versus expectations. Our year-to-year free cash flow growth also includes a modest tailwind from cash tax payments and lower payments for structural actions, partially offset by increased capex investment.

Summary:

- Click on the model name (in this example, it is **flan-ul2-20b**; yours might be different) to display the model drop-down.
- If you see **llama-3-1-8b-instruct** in the list, select it. In this example, it does not appear on the drop-down list. Click **View all foundation models** to bring up the full list.

Projects / Felix's sandbox / Prompt Lab

Chat Structured Freeform AI Model: flan-ul2-20b {#} TXT </> ⚙️

The following document is a transcript from a financial earnings call. Read the document and then write a summary in point form.

Document: Financial Highlights

I'll start with the financial highlights of the fourth quarter. We delivered \$8 billion in revenue, \$1.5 billion of operating pre-tax income and operating earnings per share of \$1.5. In our seasonally strongest quarter, we generated \$2.5 billion of free cash flow. Our revenue for the quarter was up over three percent at constant currency. While the dollar weakened a bit from 90 days ago, it still impacted our reported revenue by over \$500 million – and 3 points of growth. Revenue growth this quarter was again broad based. Product Sales revenue was up eight percent and Services up nine percent. These are our growth vectors and represent over 35 percent of our revenue. Within each of these segments, our growth was pervasive. We also had good growth across our geographies, with mid-single digit growth or better in Americas, EMEA and Asia Pacific. And for the year, we gained share overall. We had strong transactional growth in Product Sales to close the year. At the same time, our recurring revenue, which provides a solid base of revenue and profit, also grew. Earnings Prepared Remarks Earnings were up 8.5 percent from last year. Looking at our profit metrics for the quarter, we expanded operating pretax margin by 85 basis points. This reflects a strong portfolio mix and improving Product and Consulting margins. These same dynamics drove a 30-basis point increase in operating gross margin. Our expense was down year to year, driven by currency dynamics. Within our base expense, the work we're doing to digitally transform our operations provides flexibility to continue to invest in innovation and in talent. Our operating tax rate was 7 percent, which is flat versus last year. And our operating earnings per share of \$1.5 was up over seven percent. Turning to free cash flow, we generated \$2.5 billion in the quarter and \$4.5 billion for the year. Our full-year free cash flow is up \$1 billion from last year. As we talked about all year, we have a few drivers of our free cash flow growth. We had working capital improvements driven by efficiencies in operations. Despite strong collections, the combination of revenue performance above our model and the timing of the transactions in the quarter led to higher-than-expected working capital at the end of the year. This impacted our free cash flow performance versus expectations. Our year-to-year free cash flow growth also includes a modest tailwind from cash tax payments and lower payments for structural actions, partially offset by increased capex investment.

View all foundation models →

- Select the **llama-3-1-8b-instruct** model.

Select a foundation model

To choose a model, review characteristics such as tasks that models perform. Compare model benchmarks with scores in the range 0–100. Higher scores are better.

All models Model benchmarks

Filter

Task

- Chat (26)
- Question answering (28)
- Summarization (30)
- Retrieval-Augmented Generation(RAG) (22)
- Classification (30)
- Generation (30)
- Code generation and conversion (19)
- Extraction (30)

Provider

- BigScience (1)
- Code Llama (1)
- Google (3)
- IBM (12)
- Meta (12)

	IBM	Provided model	IBM	Provided model
	granite-guardian-3-8b	The Granite model series is a family of IBM-trained, dense decoder-only models, which are particularly well-suited for generative tasks.		llama-2-13b-chat
Provider: IBM	Type: Provided model	Provider: Meta	Type: Provided model	
	llama-3-1-70b-instruct	Llama-3-1-70b-instruct is an auto-regressive language model that uses an optimized transformer architecture.		llama-3-1-8b-instruct
Provider: Meta	Type: Provided model	Provider: Meta	Type: Provided model	

6. Click Select model.

The screenshot shows the Llama-3-1-8b-instruct model interface. At the top, it says "llama-3-1-8b-instruct" and "Provider: Meta | Version: 3.1 | Type: Provided model". Below this is a navigation bar with tabs: Question answering, Summarization, Retrieval-Augmented Generation, Classification, Generation, Code generation and conversion, and Extraction. A note at the bottom of the navigation bar states: "Note: This model is a Non-IBM Product governed by a third-party license that may impose use restrictions and other obligations. By using this model you agree to these terms." There is a "Read terms" link next to it. The main content area is titled "Model Information". It contains text about the Meta Llama 3.1 collection of multilingual large language models (LLMs), mentioning sizes (8B, 70B, 405B) and their optimization for multilingual dialogue use cases. It also notes the license under META LLAMA 3.1 Community License. The "Model developer" is listed as Meta. The "Model Architecture" is described as an auto-regressive language model using an optimized transformer architecture. Below this is a table with columns: Training Data, Params, Input, Output modalities, Context, GQA, Token, and Knowledge. The "Output modalities" column is highlighted with a red box and the number 6. At the bottom right of the interface is a "Select model" button.

7. Click Generate.

The screenshot shows the Prompt Lab interface. At the top, it says "Select a foundation model" and "Projects / Felix's sandbox / Prompt Lab". Below this is a toolbar with "Chat", "Structured", and "Freeform" buttons, and an "AI" button. The "Freeform" button is highlighted with a red box and the number 7. The main content area shows a "Hint: This model works better when you provide at least 1 example." and "Document: Financial Highlights". The document content is a transcript from a financial earnings call. At the bottom right of the interface is a "Generate" button.

8. You get this output:

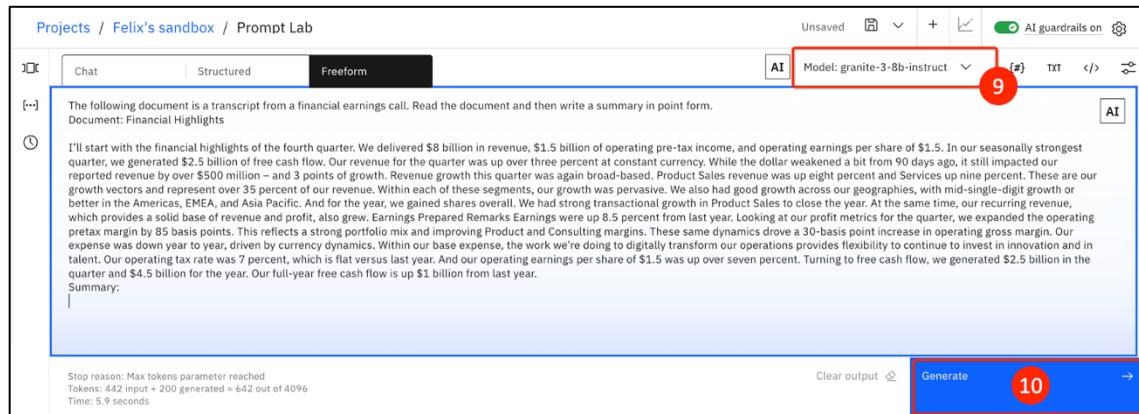
- * Revenue for the fourth quarter was \$8 billion, up 3% at constant currency.
- * Operating pre-tax income was \$1.5 billion, and operating earnings per share was \$1.5.
- * Free cash flow for the quarter was \$2.5 billion, and \$4.5 billion for the year.
- * Revenue growth was broad-based, with Product Sales up 8% and Services up 9%.
- * The company gained shares overall in the year. * Earnings were up 8.5% from last year.
- * Operating pretax margin expanded by 85 basis points.
- * Operating gross margin increased by 30 basis points.
- * Operating tax rate was 7%, flat versus last year.
- * Full-year free cash flow was up \$1 billion from last year. |||||||

This is a pretty good output (except for the series of vertical bars at the end). These may or may not be useful and may indeed require some pruning to remove.

9. **Erase the output from the previous completion** by clicking the **Clear output** icon . If you do not erase it, that input in Step 8 will become part of the input prompt.

Now re-run Steps 4-6, this time selecting the newly available (as of October 21, 2024) **granite-3-8b-instruct** model.

10. Click **Generate**.



11. You will get this output (it's OK if bullet six is not completed):

1. The company reported \$8 billion in revenue for the fourth quarter, with operating pre-tax income of \$1.5 billion and operating earnings per share of \$1.5.
2. The company generated \$2.5 billion in free cash flow for the quarter and \$4.5 billion for the year, marking a \$1 billion increase from the previous year.
3. Revenue growth was driven by Product Sales (up 8%) and Services (up 9%), which together represent over 35% of the company's revenue.
4. The company experienced mid-single-digit growth across all geographies: Americas, EMEA, and Asia Pacific.
5. Operating pretax margin expanded by 85 basis points, driven by a strong portfolio mix and improving Product and Consulting margins.
6. Operating gross margin increased by 30 basis points due to

This output is similar to the **llama-3-1-8b-instruct** model. The main differences are:

- This model is more “wordy” – it did not finish its task because it ran out of tokens (bullet six is cut off mid-sentence). You can see the error message at the bottom of the window: **Stop reason: Max tokens parameter reached**.
- This completion provides a numbered list, which is easier to read.
- The output text is in full-sentence format, even though it is a point form.

12. You can try to increase **Max tokens** to see the full output. Click the **Model parameters** icon  to display the model's parameters.

The screenshot shows the Prompt Lab interface with the following details:

- Projects / Felix's sandbox / Prompt Lab**
- Chat | Structured | Freeform**
- AI Model: granite-3-8b-instruct**
- 12** (highlighted in red)
- [...]** The following document is a transcript from a financial earnings call. Read the document and then write a summary in point form.
Document: Financial Highlights

13. Change **Max tokens** to 300. Ensure you clear the current output (i.e., if you don't, the last completion will become part of the input prompt).

14. Click **Generate**.

The screenshot shows the Prompt Lab interface with the following details:

- Projects / Felix's sandbox / Prompt Lab**
- Chat | Structured | Freeform**
- AI Model: granite-3-8b-instruct**
- 14** (highlighted in red)
- [...]** The following document is a transcript from a financial earnings call. Read the document and then write a summary in point form.
Document: Financial Highlights
- I'll start with the financial highlights of the fourth quarter. We delivered \$8 billion in revenue, \$1.5 billion of operating pre-tax income, and operating earnings per share of \$1.5. In our seasonally strongest quarter, we generated \$2.5 billion of free cash flow. Our revenue for the quarter was up over three percent at constant currency. While the dollar weakened a bit from 90 days ago, it still impacted our reported revenue by over \$500 million – and 3 points of growth. Revenue growth this quarter was again broad-based. Product Sales revenue was up eight percent and Services up nine percent. These are our growth vectors and represent over 35 percent of our revenue. Within each of these segments, our growth was pervasive. We also had good growth across our geographies, with mid-single-digit growth or better in the Americas, EMEA, and Asia Pacific. And for the year, we gained shares overall. We had strong transactional growth in Product Sales to close the year. At the same time, our recurring revenue, which provides a solid base of revenue and profit, also grew. Earnings Prepared Remarks Earnings were up 8.5 percent from last year. Looking at our profit metrics for the quarter, we expanded the operating pretax margin by 85 basis points. This reflects a strong portfolio mix and improving Product and Consulting margins. These same dynamics drove a 30-basis point increase in operating gross margin. Our expense was down year to year, driven by currency dynamics. Within our base expense, the work we're doing to digitally transform our operations provides flexibility to continue to invest in innovation and in talent. Our operating tax rate was 7 percent, which is flat versus last year. And our operating earnings per share of \$1.5 was up over seven percent. Turning to free cash flow, we generated \$2.5 billion in the quarter and \$4.5 billion for the year. Our full-year free cash flow is up \$1 billion from last year.**
- Summary:**
- Model parameters**
- Decoding**: Greedy (highlighted in red)
- Repetition penalty**: 1 (highlighted in red)
- Stopping criteria**
- Stop sequences**
- Min tokens**: 0
- Max tokens**: 300 (highlighted in red)
- 13** (highlighted in red)
- Stop reason: Max tokens parameter reached**
- Tokens: 511 input + 200 generated = 711 out of 4096**
- Time: 5.5 seconds**
- Clear output**
- Generate** (highlighted in red)
- 14** (highlighted in red)
-
- Reset to default**

You will see this output (note the model has been allowed to use enough tokens to finish the tasks at hand).

1. The company reported \$8 billion in revenue for the fourth quarter, with operating pre-tax income of \$1.5 billion and operating earnings per share of \$1.5.
2. The company generated \$2.5 billion in free cash flow for the quarter and \$4.5 billion for the year, marking a \$1 billion increase from the previous year.
3. Revenue growth was driven by Product Sales (up 8%) and Services (up 9%), which together represent over 35% of the company's revenue.
4. The company experienced mid-single-digit growth across all geographies: Americas, EMEA, and Asia Pacific.
5. Operating pretax margin expanded by 85 basis points, driven by a strong portfolio mix and improving Product and Consulting margins.
6. Operating gross margin increased by 30 basis points due to the same factors.
7. Expenses were down year to year, driven by currency dynamics, and the company continues to invest in innovation and talent.
8. The operating tax rate remained flat at 7%.
9. Operating earnings per share increased by over seven percent.

The **granite-3-8b-instruct** model's completion has similar information now to the **llama-3-1-8b-instruct** completion.

Section Summary

1. In this section, you used two different instruct models, both with 8b parameters. The new Granite 3 model performed extremely well when compared with the llama 3 model.
2. The **llama-3-1-8b-instruct** model provided a bulleted output that may work better for populating a presentation. The model also provided extra information that may require pruning.
3. The **granite-3-8b-instruct** model provided a numbered output that is more verbose (it needed a higher **Max tokens** setting) and may be better in an email or a report. This output can be used as-is without any pruning.
4. This does NOT necessarily mean one model is better than the other. It depends on the use case and what is needed. This example showcases the ability of the highlights of the new granite 3.0 model as well as the need to test out different models.
5. You are encouraged to try out the other Granite 3.0 models on watsonx.ai:
 - **Instruct** models – designed to excel in instruction-following tasks such as summarization, problem-solving, text translation, reasoning, code tasks, function-calling, and more.
 - granite-3-2b-instruct
 - granite 3-8b-instruct
 - **Guardian** models – designed to detect risks in prompts and responses.
 - granite-guardian-3-2b
 - granite-guardian-3-8b
6. There are two other types of Granite models that you can access on Hugging Face (not watsonx.ai).
 - **Mixture of Expert (MoE)** models – designed for efficient inferencing
 - granite-3.0-3b-a800m-instruct (or just granite-3.0-3b-a800m)
 - granite-3.0-1b-400m-instruct (or just granite-3.0-1b-a400m)
 - **Speculative decoding** model – use speculative decoding to accelerate inference time
 - granite-3.0-8b-instruct-accelerator

5.5 Why Model Choice Matters (Multimodal vs Text-only)

The watsonx.ai foundation model library recently had some new multimodal model additions (Q3 2024), including:

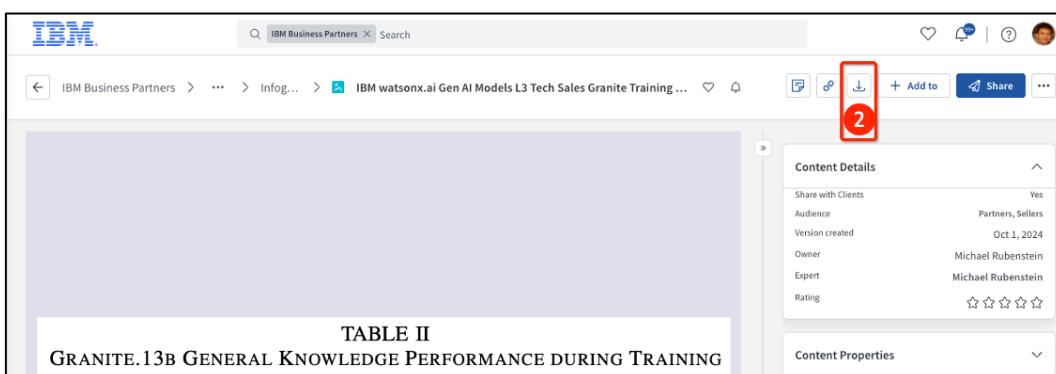
- llama-3-2-11b-vision-instruct
- llama-3-2-90b-vision-instruct
- llama-guard-3-11b-vision
- llama3-llava-next-8b-hf (marked for deprecation)

These are IBM watsonx.ai's first multimodal models designed to work with text and image input to generate output.

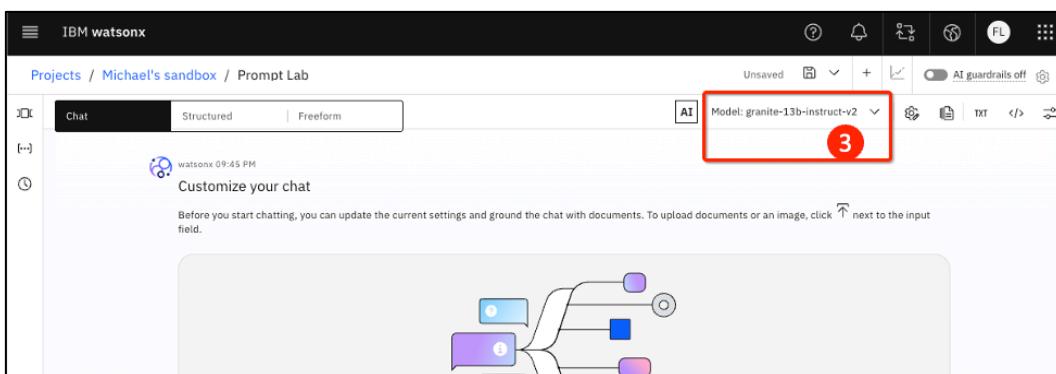
In contrast, the previous examples utilized text-only models that operated on a text-input-to-text-output basis, focusing solely on chat-based interactions where both prompts and responses consisted exclusively of text.

This section shows the differences between multimodal models and text-only models.

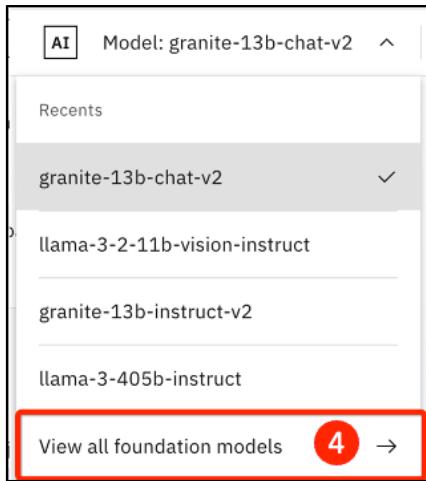
1. Download the table image (a PNG file) used for the lab at:
<https://ibm.seismic.com/Link/Content/DC26X6HpcJjb88qP4q847CVFGPWG>. The file name is **IBM watsonx.ai Gen AI Models L3 Tech Sales Granite Training Performance.PNG**.
2. Click the download button () on the top right of the page.



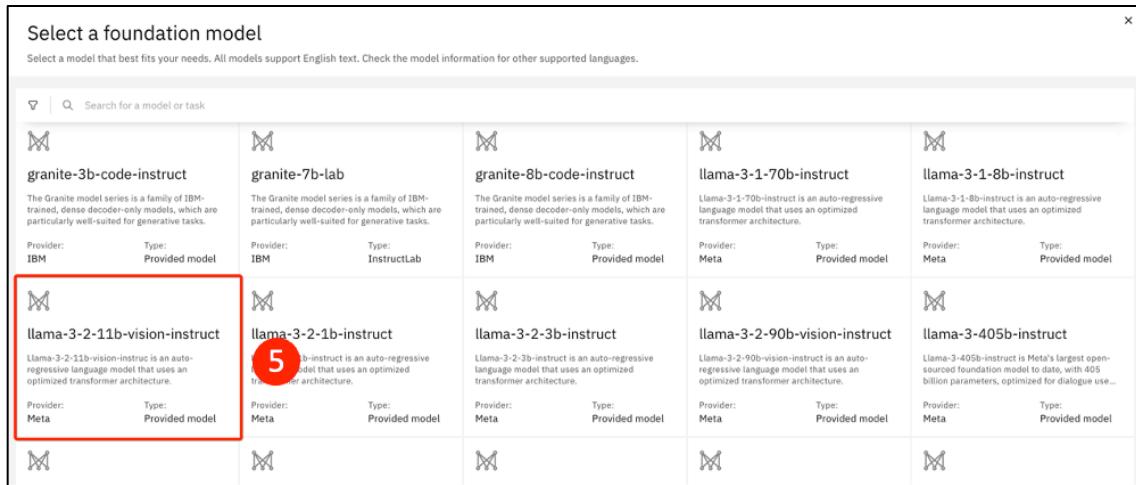
3. Open up the Prompt Lab (or click on the Prompt Lab window if it is still open) and click on the **Chat** tab. Click the model selection dropdown menu.



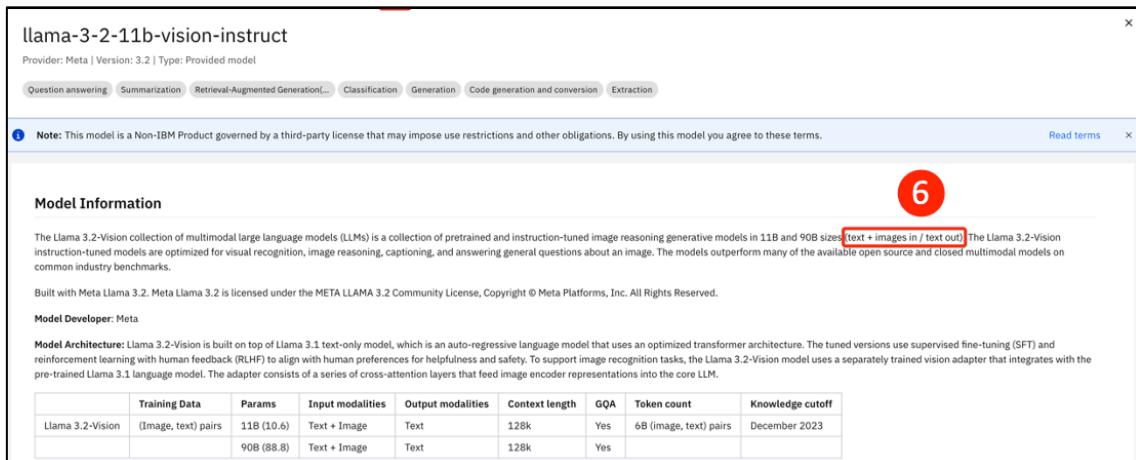
4. Click View all foundation models.



5. Either scroll or filter through the models to find the **llama-3-2-11b-vision-instruct** model tile and click on it.



6. As you can see, this model was trained with both text and images to output text responses.



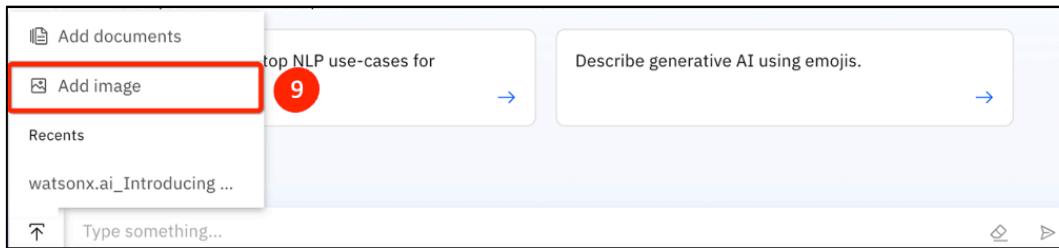
7. Click **Select model**.

The screenshot shows the 'llama-3-2-11b-vision-instruct' model page. At the top, it says 'Provider: Meta | Version: 3.2 | Type: Provided model'. Below that is a note about a third-party license. The 'Model Information' section contains details like training data, parameters, and architecture. A table provides specific data points. The 'Supported Languages' section notes that English is the only language supported. At the bottom right, there is a red box around the 'Select model' button.

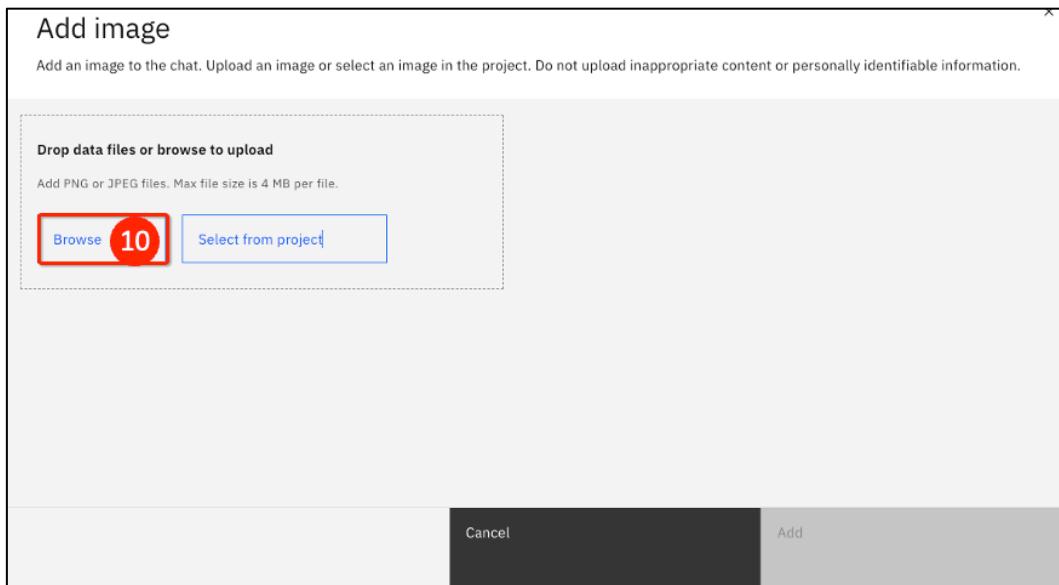
8. Click the upload button (⬇) on the left side of the **Type something...** field.

The screenshot shows the 'Prompt Lab' interface. It features a header with 'Projects / Michael's sandbox / Prompt Lab' and various toolbars. Below the header is a 'Sample questions' section with four examples. At the bottom, there is a text input field labeled 'Type something...' with an upload button (represented by a downward arrow icon). A red box highlights this upload button.

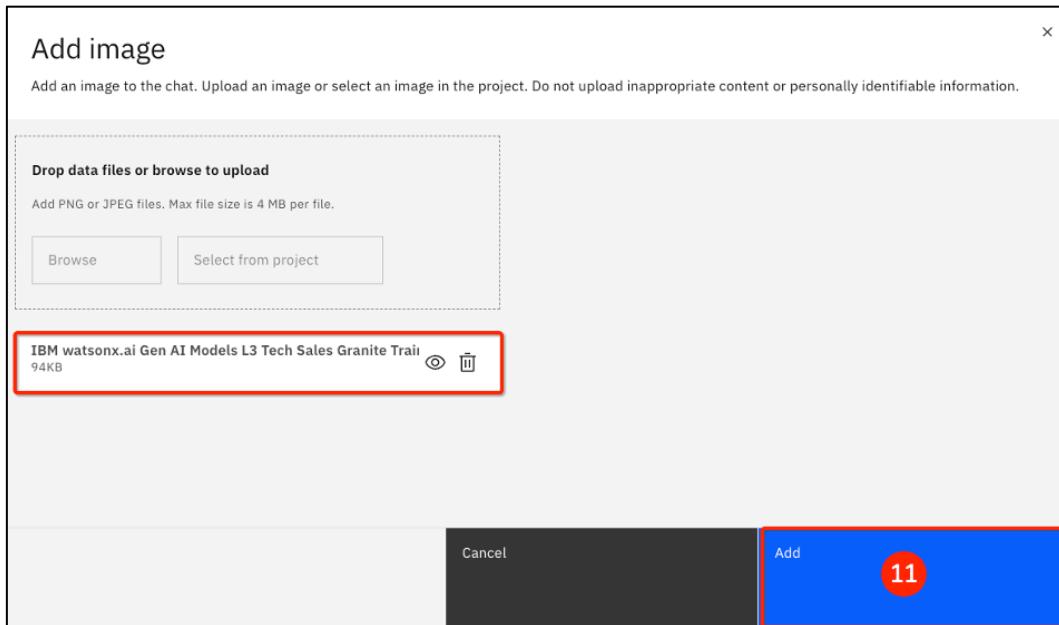
9. Click **Add image** on the popup menu.



10. Click the **Browse** button, navigate to the location of the image you downloaded in Step 2, select it, and click **Open** (for MacOS, you may have **Upload** instead).



11. Once the image is correctly listed under the **Add image** box, click **Add**.



12. When the image is added correctly, it shows up above the chat bar.

Note: Don't worry about the sizing of the image (the displayed image is stretched), this shouldn't affect your results

13. Now copy and paste the prompt below into the Type something... field:

Extract the data from this table and turn it into a markdown table.

The screenshot shows the Prompt Lab interface. At the top, there are tabs for Chat, Structured, and Freeform. The Model dropdown is set to "llama-3-2-11b-vision-instruct". Below the tabs, there are four sample questions in boxes. In the center, there is a table titled "TABLE II GRANITE.13B GENERAL KNOWLEDGE PERFORMANCE DURING TRAINING". The table has columns for Model, Tokens (B), Avg Accuracy (Zero-Shot), and Avg Accuracy (Few-Shot). The last row of the table is highlighted with a red box and circled with a red number "12". Below the table, there is a file preview for "IBM watsonx.ai Gen AI Models L3 Tech Sales Granite Training Performance.PNG". At the bottom, there is a text input field containing the instruction "Extract the data from this table and turn it into a markdown table." This input field is also circled with a red box and labeled with a red number "13".

Model	Tokens (B)	Avg Accuracy (Zero-Shot)	Avg Accuracy (Few-Shot)
granite.13b (base)	100	49.0	53.3
granite.13b (base)	200	50.8	55.2
granite.13b (base)	300	53.7	56.1
granite.13b (base)	400	52.9	57.1
granite.13b (base)	500	55.6	57.8
granite.13b (base)	600	55.7	58.1
granite.13b (base)	700	56.8	59.3
granite.13b (base)	800	56.5	59.9
granite.13b (base)	900	57.8	60.0
granite.13b (base)	1000	58.5	61.0

14. Click either the **Enter** key on your keyboard or the ➤ icon on the right of the **Type something...** field.

The screenshot shows the same Prompt Lab interface as the previous one. The table and the text input field are identical. However, the ➤ button to the right of the text input field is now highlighted with a red box and circled with a red number "14", indicating it has been clicked.

Extract the data from this table and turn it into a markdown table.



watsonx 01:39 PM

Model	Tokens (B)	Avg Accuracy (Zero-Shot)	Avg Accuracy (Few-Shot)
granite.13b (base)	100	49.0	53.3
granite.13b (base)	200	50.8	55.2
granite.13b (base)	300	53.7	56.1
granite.13b (base)	400	52.9	57.1
granite.13b (base)	500	55.6	57.8
granite.13b (base)	600	55.7	58.1
granite.13b (base)	700	56.8	59.3
granite.13b (base)	800	56.5	59.9
granite.13b (base)	900	57.8	60.0
granite.13b (base)	1000	58.5	61.0

The **llama-3-2-11b-vision-instruct** understood the table in the uploaded image and can also translate that into code output in markdown format.

15. Enter the following text into the Type something... field and press Enter.

Which single step represents the largest increase in accuracy for Zero-Shot?

You will get the following output:



You: 02:06 PM

Which single step represents the largest increase in accuracy for Zero-Shot?



watsonx 02:06 PM

To determine the single step with the largest increase in accuracy for Zero-Shot, we need to calculate the difference in accuracy between each step.

- Step 1: 49.0 (100 tokens) to 50.8 (200 tokens) = 1.8
- Step 2: 50.8 (200 tokens) to 53.7 (300 tokens) = 2.9
- Step 3: 53.7 (300 tokens) to 52.9 (400 tokens) = -0.8
- Step 4: 52.9 (400 tokens) to 55.6 (500 tokens) = 2.7
- Step 5: 55.6 (500 tokens) to 55.7 (600 tokens) = 0.1
- Step 6: 55.7 (600 tokens) to 56.8 (700 tokens) = 1.1
- Step 7: 56.8 (700 tokens) to 56.5 (800 tokens) = -0.3
- Step 8: 56.5 (800 tokens) to 57.8 (900 tokens) = 1.3
- Step 9: 57.8 (900 tokens) to 58.5 (1000 tokens) = 0.7

The largest increase in accuracy for Zero-Shot is 2.9, which occurs between 200 tokens and 300 tokens. Therefore, the single step that represents the largest increase in accuracy for Zero-Shot is going from 200 tokens to 300 tokens.

The llama-3-2-11b-vision-instruct can reason from the table data and deduce that the biggest jump in accuracy is from 100 tokens to 200 tokens.

Section Summary

- Multi-modal foundation models allow clients to run various use cases against an image in addition to text. Clients can perform classification (for example, identify fruit, car brand, tool, or various objects), summarize (for example, what the picture is about, how damaged the vehicle is shown in the picture), or generate text in markdown format.
- These models offer exciting possibilities for client use cases; however, keep in mind that they are large, and they are more costly to use (this is typical when images are involved). Clients should be aware of these models' availability and select them based on specific needs.

6 Custom Foundation Model (CFM)

In addition to the models available as part of watsonx.ai, clients can deploy custom foundation models (often referred to as Bring Your Own Model or BYOM for short) to watsonx.ai. The details of how to do this are included in the [watsonx.ai Custom Foundation Model for Technical Sales lab](#) and will not be covered here. Several important notes to keep in mind with CFM:

- Clients should check to see if the models they want to deploy to watsonx.ai are supported by the **Custom Foundation Model** feature. Refer to [Planning to deploy a custom foundation model](#) to find information on supported model architecture and other requirements.
- As of October 2024, a single account can be used to deploy four small models, two medium models, or one large model. See [Planning to deploy a custom foundation model](#) for information on different configurations available to different architecture types.
- The **Custom Foundation Model** feature is supported both in the cloud (Software as a Service SaaS) or on-premises. For on-premises, clients may need to provision additional hardware.
- For billing questions, consult [Billing details for generative AI assets](#).

The **Custom Foundation Model** feature offers clients the flexibility to leverage models they might have already tuned/used and allows them to leverage various features of watsonx.ai, such as automation, development, and more.

7 Summary

In generative AI, no one model can fit all of a client's use cases. There is often a trade-off.

1. Larger models “know” more and can generally perform better “out-of-box” for most tasks (the question is: how good is “good enough”). However, they are more creative and may inject higher variability and hallucination. They are also expensive to use or to tune.
2. Smaller models may perform well against some tasks and less so for others. They tend to be terser in output and may or may not be what the clients want. While tuning and running are relatively less costly, it may take more data to tune a smaller model satisfactorily (when compared to a larger model).
3. Certain models are tuned for conversation or instructions. Clients should consider their use case when selecting an appropriate model.
4. There are various restrictions a client may need to consider. For example:
5. European clients may prefer a local vendor's model, such as the **mistral-large** model.
6. Budget considerations may require adapting and tuning a smaller model for less costly runtime.
7. Clients may require on-premises deployment (hybrid capability) or Custom Foundation Model capabilities.
8. Clients may need ways to compare model outputs. Initial investigations can be performed using the Model comparison tool (or **ComFM** for short – see the [IBM watsonx.ai Models comparison](#) lab). That is useful in *eliminating* some model choices. However, the client needs to test their use cases with their data and an appropriate environment to settle on model choices.

Appendix A. Revision History

Date	Changes
November 2024	Original version.