# Model evaluation



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

Amazon Bedrock supports model evaluation jobs. The results of a model evaluation job allow you to evaluate and compare a model's outputs, and then choose the model best suited for your downstream generative AI applications.

Model evaluation jobs support common use cases for large language models (LLMs) such as text generation, text classification, question and answering, and text summarization.

To evaluate a model's performance you can use either built-in prompt datasets or your own prompt datasets.

You can choose to create either an automatic model evaluation job or a model evaluation job that uses a human workforce.

#### Overview: Automatic model evaluation jobs

Automatic model evaluation jobs allow you to quickly evaluate a model's ability to perform a task. You can either provide your own custom prompt dataset that you've tailored to a specific use case, or you can use an available built-in dataset.

## Overview: Model evaluation jobs that use human workers

Model evaluation jobs that use human workers allow you to bring human input to the model evaluation process. They can be employees of your company or a group of subject-matter experts from your industry.

The following topics describe the available model evaluation tasks, and the kinds of metrics you can use. They also describe the available built-in datasets and how to specify your own dataset.

### **Topics**

- Getting started with model evaluations
- Model evaluation tasks
- Using prompt datasets in model evaluation jobs
- Creating good worker instructions
- Creating and managing work teams in Amazon Bedrock
- Model evaluation job results
- Required IAM permissions and service roles to create a model evaluation job

# Getting started with model evaluations



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

You can create a model evaluation job that is either automatic or uses human workers. When you create a model evaluation job, you can define the model used, the inference parameters, the type of task the model tries to perform, and the prompt data used in the job.

#### Model evaluation jobs support the following task types.

- General text generation: The production of natural human language in response to text prompts.
- **Text summarization**: The generation of a summary of a provided text within your prompt.
- Question and answering: The generation of a response to a question within a prompt.
- Classification: Correctly assigning a category, such as a label or score, to text based on its content.
- **Custom**: You define the metric, description, and a rating method.

To create a model evaluation job, you must have access to Amazon Bedrock models. Model evaluation jobs support using Amazon Bedrock base models. To learn more about model access, see Model access.

Get started 228

The procedures in the following topics show you how to set up a model evaluation job using the Amazon Bedrock console.

To create model evaluation job with the help of an AWS-managed team, choose Create AWS managed evaluation. Then, fill out the request form with details about your model evaluation job requirements, and an AWS team member will get in touch with you.

#### **Topics**

- Creating an automatic model evaluation
- Creating a model evaluation job that uses human workers

# Creating an automatic model evaluation



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

## Prerequisites

To complete the following procedures you must do the following.

- 1. You must have access to the model in Amazon Bedrock.
- 2. You must have an Amazon Bedrock service role. The attached policy must grant access to the following resources. Any S3 buckets used in the model evaluation job, and the ARN of the model specified in the job. The service role must also have Amazon Bedrock defined as a service principal in the role's trust policy. To learn more, see Required permissions.
- 3. The user, group, or role accessing the Amazon Bedrock console must have the required permissions to access the required Amazon S3 buckets. To learn more, see Required permissions

Automatic model evaluations allow you to evaluate the aggregate responses from a single model using recommended metrics. You can also use built-in prompt datasets or use your own custom

Automatic model evaluations 229

prompt dataset. You can have a maximum of 10 automatic model evaluation jobs In progress in your account per AWS Region.

When you set up an automatic model evaluation job, the available metrics and the built-in datasets best suited for the selected task type are automatically added to the job. You can add or remove any of the preselected datasets. You also supply your own custom prompt dataset.

#### ★ Viewing the model evaluation job results using the Amazon Bedrock console

When a model evaluation job finishes, the results are stored in the Amazon S3 bucket you specified. If you modify the location of the results in any way, the model evaluation report card is no longer visible in the console.

#### To create an automatic model evaluation

- Open the Amazon Bedrock console: https://console.aws.amazon.com/bedrock/ 1.
- In the navigation pane, choose **Model evaluation**. 2.
- 3. In the **Build an evaluation** card, under **Automatic** choose **Create automatic evaluation**.
- On the Create automatic evaluation page, provide the following information 4.
  - **Evaluation name** Give the model evaluation job a name that describes the job. This a. name is shown in your model evaluation job list. The name must be unique in your account in an AWS Region.
  - **Description** (Optional) Provide an optional description.
  - **Models** Choose the model you want to use in the model evaluation job. C.

To learn more about available models in Amazon Bedrock, see Model access.

- (Optional) To change the inference configuration choose **update**.
  - Changing the inference configuration changes the responses generated by the selected models. To learn more about the available inferences parameters, see Inference parameters for foundation models.
- **Task type** Choose the type of task you want the model to attempt to perform during the model evaluation job.
- f. Metrics and datasets — The list of available metrics and built-in prompt datasets change based on the task you select. You can choose from the list of Available built-in datasets

Automatic model evaluations 230

or you can choose **Use your own prompt dataset**. If you choose to use your own prompt dataset, enter the exact S3 URI where your dataset is stored.

- g. **Model evaluation results** You must specify the S3 URI of the directory where you want the results of your model evaluation job saved.
- h. **IAM role** Choose a service role that has the required permissions.
- i. (Optional) Choose **Customize encryption settings (advanced)** Provide the ARN of the AWS KMS key used to encrypt your data in Amazon S3.
- 5. To start your model evaluation job, choose **Create**.

Once the job has successfully started, the status changes to **In progress**. When the job has finished, the status changes to **Ready**.

To learn how to evaluate, view, and download the results of your model evaluation job, see <u>Model</u> evaluation job results.

# Creating a model evaluation job that uses human workers

### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

# Prerequisites

To complete the following procedure you must do the following.

- 1. You must have access to the models in Amazon Bedrock.
- 2. You must have an Amazon Bedrock service role. The attached policy must grant access to the following resources. Any S3 buckets used in the model evaluation job, and the ARNs of any models specified in the job. It must also have the sagemaker:StartHumanLoop and sagemaker:DescribeFlowDefinition SageMaker IAM actions defined in the policy. The service role must also have Amazon Bedrock defined as a service principal in the role's trust policy. To learn more, see Required permissions.

3. You must have a Amazon SageMaker service role. The attached policy must grant access to the following resources and IAM actions. Any S3 buckets used in the model evaluation job. The role's trust policy must have SageMaker defined as the service principal. To learn more, see Required permissions.

4. The user, group, or role accessing the Amazon Bedrock console must have the required permissions access the required Amazon S3 buckets.

In a model evaluation job that uses human workers you can compare the responses for up to two models. You can choose from a list of recommended metrics or use metrics that you define yourself. You can have a maximum of 20 model evaluation jobs that use human workers In progress in your account per AWS Region.

For each metric that you use, you must define a **Rating method**. The rating method defines how your human workers will evaluate the responses they get from the models you've selected. To learn more about the different available rating methods and how to create high quality instructions for workers, see Creating and managing work teams in Amazon Bedrock

#### ★ Viewing the model evaluation job results using the Amazon Bedrock console

When a model evaluation job finishes, the results are stored in the Amazon S3 bucket you specified. If you modify the location of the results in any way, the model evaluation report card is no longer visible in the console.

#### To create a model evaluation job that uses human workers

- Open the Amazon Bedrock console: https://console.aws.amazon.com/bedrock/home 1.
- 2. In the navigation pane, choose **Model evaluation**.
- 3. In the **Build an evaluation** card, under **Human: bring your own team** choose **Create human** evaluation.
- On the **Specify job details** page provide the following.
  - **Evaluation name** Give the model evaluation job a name that describes the job. This a. name is shown in your model evaluation job list. The name must be unique in your account in an AWS Region.
  - **Description** (Optional) Provide an optional description. b.

- 5. Then, choose **Next**.
- 6. On the **Set up evaluation** page provide the following.
  - a. **Models** You can choose up to two models you want to use in the model evaluation job.
    - To learn more about available models in Amazon Bedrock, see Model access.
  - b. (Optional) To change the inference configuration choose update.
    - Changing the inference configuration changes the responses generated by the selected models. To learn more about the available inferences parameters, see <a href="Inference">Inference</a> parameters for foundation models.
  - c. **Task type** Choose the type of task you want the model to attempt to perform during the model evaluation job. All instructions for the model must be included in the prompts themselves. The task type does not control the model's responses.
  - d. **Evaluation metrics** The list of recommended metrics changes based on the task you select. For each recommended metric, you must select a **Rating method**. You can have a maximum of 10 evaluation metrics per model evaluation job.
  - e. (Optional) Choose Add custom metric to add a custom metric. You must define the Metric, Description, and Rating method.
  - f. In the **Datasets** card your must provide the following.
    - i. **Evaluation dataset** Specify the S3 URI where your prompt dataset is stored. You can have a maximum of 1000 prompts in a custom prompt dataset.
    - ii. **Evaluation results destination** You must specify the S3 URI of the directory where you want the results of your model evaluation job saved.
    - iii. (Optional) **Encryption key** Provide the ARN of the AWS KMS key used to encrypt your data in Amazon S3.
- 7. Then, choose Next.
- 8. In the **Permissions** card, specify the following. To learn more about the required permissions for model evaluations, see <u>Required IAM permissions and service roles to create a model evaluation job</u>.
  - a. **Amazon Bedrock IAM role** Specify an Amazon Bedrock service role that has the required permissions.
  - b. **Human workflow IAM role** Specify a SageMaker service role that has the required permissions.

In the Work team card, specify the following. 9.

#### ★ Human worker notification requirements

When you add a new human worker to a model evaluation job, they automatically receive an email inviting them to participate in the model evaluation job. When you add an existing human worker to a model evaluation job, you must notify and provide them with worker portal URL for the model evaluation job. The existing worker will not receive an automated email notification that they are added to the new model evaluation job.

- Using the **Select team** dropdown, specify either **Create a new work team** or the name of a. an existing work team.
- (Optional) Number of workers per prompt Update the number of workers who evaluate each prompt. After the responses for each prompt have been reviewed by the number of workers you selected, the prompt and its responses will be taken out of circulation from the work team. The final results report will include all ratings from each worker.
- (Optional) Existing worker email Choose this to copy an email template containing the worker portal URL.
- (Optional) New worker email Choose this to view the email new workers receive automatically.



#### 

Large language models are known to occasionally hallucinate and produce toxic or offensive content. Your workers may be shown toxic or offensive material during this evaluation. Ensure you take proper steps to train and notify them before they work on the evaluation. They can decline and release tasks or take breaks during the evaluation while accessing the human evaluation tool.

- 10. Then, choose Next.
- 11. On the **Provide instruction page** use the text editor to provide instructions for completing the task. You can preview the evaluation UI that your work team uses to evaluate the responses, including the metrics, rating methods, and your instructions. This preview is based on the configuration you have created for this job.

- Then, choose Next.
- 13. On the Review and create page, you can view a summary of the options you've selected in the previous steps.

14. To start your model evaluation job, choose **Create**.

Once the job has successfully started, the status changes to In progress. When the job has finished, the status changes to Completed.

To learn how to evaluate, view, and download the results of your model evaluation job, see Model evaluation job results.

# Model evaluation tasks



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

In a model evaluation job, an evaluation task is a task you want the model to perform based on information in your prompts.

You can choose one task type per model evaluation job. Use the following topics to learn more about each task type. Each topic also includes a list of available built-in datasets and their corresponding metrics that can be used only in automatic model evaluation jobs.

#### **Topics**

- General text generation
- Text summarization
- Question and answer
- Text classification

Model evaluation tasks 235

# General text generation



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

General text generation is a task used by applications that include chatbots. The responses generated by a model to general questions are influenced by the correctness, relevance, and bias contained in the text used to train the model.

The following built-in datasets contain prompts that are well-suited for use in general text generation tasks.

#### Bias in Open-ended Language Generation Dataset (BOLD)

The Bias in Open-ended Language Generation Dataset (BOLD) is a dataset that evaluates fairness in general text generation, focusing on five domains: profession, gender, race, religious ideologies, and political ideologies. It contains 23,679 different text generation prompts.

#### RealToxicityPrompts

RealToxicityPrompts is a dataset that evaluates toxicity. It attempts to get the model to generate racist, sexist, or otherwise toxic language. to This dataset contains 100,000 different text generation prompts.

#### T-Rex: A Large Scale Alignment of Natural Language with Knowledge Base Triples (TREX)

TREX is dataset consisting of Knowledge Base Triples (KBTs) extracted from Wikipedia. KBTs are a type of data structure used in natural language processing (NLP) and knowledge representation. They consist of a subject, predicate, and object, where the subject and object are linked by a relation. An example of a Knowledge Base Triple (KBT) is "George Washington was the president of the United States". The subject is "George Washington", the predicate is "was the president of", and the object is "the United States".

#### WikiText2

WikiText2 is a HuggingFace dataset that contains prompts used in general text generation.

General text generation 236

The following table summarizes the metrics calculated, and recommended built-in dataset that are available for automatic model evaluation jobs.

#### Available built-in datasets for general text generation in Amazon Bedrock

Task type	Metric	Built-in datasets	Computed metric
General text generation	Accuracy	TREX	Real world knowledge (RWK) score
	Robustness	BOLD	Word error rate
		WikiText2	
		TREX	
	Toxicity	RealToxicityPrompts	Toxicity
		BOLD	

For general text generation, there is a known system issue that prevents Cohere Command and Anthropic Claude models from completing a robustness evaluation successfully.

To learn more about how the computed metric for each built-in dataset is calculated, see Model evaluation job results

## **Text summarization**



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

Text summarization is used for tasks including creating summaries of news, legal documents, academic papers, content previews, and content curation. The ambiguity, coherence, bias, and

Text summarization 237

fluency of the text used to train the model as well as information loss, accuracy, relevance, or context mismatch can influence the quality of responses.

The following built-in dataset is supported for use with the task summarization task type.

#### Gigaword

The Gigaword dataset consists of news article headlines. This dataset is used in text summarization tasks.

The following table summarizes the metrics calculated, and recommended built-in dataset.

#### Available built-in datasets for text summarization in Amazon Bedrock

Task type	Metric	Built-in datasets	Computed metric
Text summarization	Accuracy	Gigaword	BERTScore
	Toxicity	Gigaword	Toxicity
	Robustness	Gigaword	BERTScore and deltaBERTScore

For text summarization, there is a known system issue that prevents Anthropic Claude models from completing a Robustness evaluation successfully.

To learn more about how the computed metric for each built-in dataset is calculated, see Model evaluation job results

# **Question and answer**



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

Question and answer 238

Question and answer is used for tasks including generating automatic help-desk responses, information retrieval, and e-learning. If the text used to train the foundation model contains issues including incomplete or inaccurate data, sarcasm or irony, the quality of responses can deteriorate.

The following built-in datasets are recommended for use with the question and answer task type.

#### BoolQ

BoolQ is a dataset consisting of yes/no question and answer pairs. The prompt contains a short passage, and then a question about the passage. This dataset is recommended for use with question and answer task type.

#### **Natural Questions**

Natural questions is a dataset consisting of real user questions submitted to Google search.

### **TriviaQA**

TriviaQA is a dataset that contains over 650K question-answer-evidence-triples. This dataset is used in question and answer tasks.

The following table summarizes the metrics calculated, and recommended built-in dataset.

### Available built-in datasets for the question and answer task type in Amazon Bedrock

Task type	Metric	Built-in datasets	Computed metric
Question and	Accuracy	BoolQ	NLP-F1
answer		<u>NaturalQuestions</u>	
		<u>TriviaQA</u>	
	Robustness	BoolQ	F1 and deltaF1
		<u>NaturalQuestions</u>	
		TriviaQA	
	Toxicity	BoolQ	Toxicity
		<u>NaturalQuestions</u>	

Question and answer 239

Task type	Metric	Built-in datasets	Computed metric
		TriviaQA	

For question and answer, there is a known system issue that prevents Anthropic Claude models from completing a robustness evaluation successfully.

To learn more about how the computed metric for each built-in dataset is calculated, see Model evaluation job results

### Text classification



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

Text classification is used to categorize text into pre-defined categories. Applications that use text classification include content recommendation, spam detection, language identification and trend analysis on social media. Imbalanced classes, ambiguous data, noisy data, and bias in labeling are some issues that can cause errors in text classification.

The following built-in datasets are recommended for use with the text classification task type.

#### **Women's E-Commerce Clothing Reviews**

Women's E-Commerce Clothing Reviews is a dataset that contains clothing reviews written by customers. This dataset is used in text classification tasks.

The following table summarizes the metrics calculated, and recommended built-in datasets.

#### Available built-in datasets in Amazon Bedrock

Task type	Metric	Built-in datasets	Computed metric
Text classific ation	Accuracy	Women's Ecommerce Clothing Reviews	

Text classification 240

Task type	Metric	Built-in datasets	Computed metric
			Accuracy (Binary Accuracy from classificatio n_accuracy_score)
	Robustness	Women's Ecommerce Clothing Reviews	classification_acc uracy_score and delta_classificati on_accuracy_score

For text classification, there is a known system issue that prevents Anthropic Claude models from completing a Robustness evaluation successfully.

To learn more about how the computed metric for each built-in dataset is calculated, see Model evaluation job results

# Using prompt datasets in model evaluation jobs



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

To create a model evaluation job you must specify a prompt dataset the model uses during inference. Amazon Bedrock provides built-in datasets that can be used in automatic model evaluations, or you can bring your own prompt dataset. For model evaluation jobs that use human workers you must use your own prompt dataset.

Use the following sections to learn more about available built-in prompt datasets and creating your custom prompt datasets.

To learn more about creating your first model evaluation job in Amazon Bedrock, see Model evaluation.

Input prompt datasets 241

#### **Topics**

- Using built-in prompt datasets in automatic model evaluation jobs
- Custom prompt dataset

# Using built-in prompt datasets in automatic model evaluation jobs



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

Amazon Bedrock provides multiple built-in prompt datasets that you can use in an automatic model evaluation job. Each built-in dataset is based off an open-source dataset. We have down sampled each open-source dataset to include only 100 prompts.

All built-in prompt datasets use the format that Anthropic Claude models require. This means each prompt contains "Human: " at the start of each prompt, and "Assistant: " at the end. To avoid this scenario, you can use the original open source versions of the prompt datasets, and upload them as a custom dataset when you create a model evaluation job.

When you create an automatic model evaluation job and choose a **Task type** Amazon Bedrock provides you with a list of recommended metrics. For each metric, Amazon Bedrock also provides recommended built-in datasets. To learn more about available task types, see Model evaluation tasks.

#### Bias in Open-ended Language Generation Dataset (BOLD)

The Bias in Open-ended Language Generation Dataset (BOLD) is a dataset that evaluates fairness in general text generation, focusing on five domains: profession, gender, race, religious ideologies, and political ideologies. It contains 23,679 different text generation prompts.

### RealToxicityPrompts

RealToxicityPrompts is a dataset that evaluates toxicity. It attempts to get the model to generate racist, sexist, or otherwise toxic language. This dataset contains 100,000 different text generation prompts.

242 Built-in prompt datasets

#### T-Rex: A Large Scale Alignment of Natural Language with Knowledge Base Triples (TREX)

TREX is dataset consisting of Knowledge Base Triples (KBTs) extracted from Wikipedia. KBTs are a type of data structure used in natural language processing (NLP) and knowledge representation. They consist of a subject, predicate, and object, where the subject and object are linked by a relation. An example of a Knowledge Base Triple (KBT) is "George Washington was the president of the United States". The subject is "George Washington", the predicate is "was the president of", and the object is "the United States".

#### WikiText2

WikiText2 is a HuggingFace dataset that contains prompts used in general text generation.

#### **Gigaword**

The Gigaword dataset consists of news article headlines. This dataset is used in text summarization tasks.

#### **BoolQ**

BoolQ is a dataset consisting of yes/no question and answer pairs. The prompt contains a short passage, and then a question about the passage. This dataset is recommended for use with question and answer task type.

#### **Natural Questions**

Natural question is a dataset consisting of real user questions submitted to Google search.

#### **TriviaQA**

TriviaQA is a dataset that contains over 650K question-answer-evidence-triples. This dataset is used in question and answer tasks.

#### **Women's E-Commerce Clothing Reviews**

Women's E-Commerce Clothing Reviews is a dataset that contains clothing reviews written by customers. This dataset is used in text classification tasks.

In the following table, you can see the list of available datasets grouped task type. To learn more about how automatic metrics are computed, see <u>Automated model evaluation job report cards</u> (console).

Built-in prompt datasets 243

# Available built-in datasets for automatic model evaluation jobs in Amazon Bedrock

Task type	Metric	Built-in datasets	Computed metric
General text generation	Accuracy	TREX	Real world knowledge (RWK) score
	Robustness	BOLD	Word error rate
		WikiText2	
		English Wikipedia	
	Toxicity	RealToxicityPrompts	Toxicity
		BOLD	
Text summarization	Accuracy	Gigaword	BERTScore
Sullillalization	Toxicity	Gigaword	Toxicity
	Robustness	Gigaword	BERTScore and deltaBERTScore
Question and	Accuracy	BoolQ	NLP-F1
answer		<u>NaturalQuestions</u>	
		<u>TriviaQA</u>	
	Robustness	BoolQ	F1 and deltaF1
		<u>NaturalQuestions</u>	
		<u>TriviaQA</u>	
	Toxicity	BoolQ	Toxicity
		NaturalQuestions	

Built-in prompt datasets 244

User Guide Amazon Bedrock

Task type	Metric	Built-in datasets	Computed metric
		<u>TriviaQA</u>	
Text classific ation	Accuracy	Women's Ecommerce Clothing Reviews	Accuracy (Binary accuracy from classification_accuracy_score)
		Women's Ecommerce Clothing Reviews	
		Women's Ecommerce Clothing Reviews	
	Robustness	Women's Ecommerce Clothing Reviews	classification_acc uracy_score and delta _classification_ac curacy_score

To learn more about the requirements for creating and examples of custom prompt datasets, see Custom prompt dataset.

# **Custom prompt dataset**



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

You can use a custom prompt dataset in model evaluation jobs.

Custom prompt datasets must be stored in Amazon S3, and use the JSON line format and use the . jsonl file extension. When you upload the dataset to Amazon S3 make sure that you update the Cross Origin Resource Sharing (CORS) configuration on the S3 bucket. To learn more about the

required CORS permissions, see <u>Required Cross Origin Resource Sharing (CORS) permission on S3</u> buckets.

#### **Topics**

- Requirements for custom prompt datasets used in automatic model evaluation jobs
- Requirements for custom prompt datasets in model evaluation job that use human workers

# Requirements for custom prompt datasets used in automatic model evaluation jobs

In automatic model evaluation jobs you can use a custom prompt dataset for each metric you select in the model evaluation job. Custom datasets use the JSON line format (.jsonl), and each line must be a valid JSON object. There can be up to 1000 prompts in your dataset per automatic evaluation job.

You must use the following keys in a custom dataset.

- prompt required to indicate the input for the following tasks:
  - The prompt that your model should respond to, in general text generation.
  - The question that your model should answer in the question and answer task type.
  - The text that your model should summarize in text summarization task.
  - The text that your model should classify in classification tasks.
- referenceResponse required to indicate the ground truth response against which your model is evaluated for the following tasks types:
  - The answer for all prompts in question and answer tasks.
  - The answer for all accuracy, and robustness evaluations.
- category– (optional) generates evaluation scores reported for each category.

As an example, accuracy requires both the question to ask and the answer to check the model response against. In this example, use the key prompt with the value contained in the question, and the key referenceResponse with the value contained in the answer as follows.

```
{
    "prompt": "Bobigny is the capital of",
    "referenceResponse": "Seine-Saint-Denis",
    "category": "Capitals"
```

}

The previous example is a single line of a JSON line input file that will be sent to your model as an inference request. Model will be invoked for every such record in your JSON line dataset. The following data input example is for a question answer task that uses an optional category key for evaluation.



#### Important

After your last prompt in your custom dataset, the file must end with a newline.

```
{"referenceResponse":"Cantal", "category":"Capitals", "prompt":"Aurillac is the capital
 of"}
{"referenceResponse": "Bamiyan Province", "category": "Capitals", "prompt": "Bamiyan city is
the capital of"}
{"referenceResponse": "Abkhazia", "category": "Capitals", "prompt": "Sokhumi is the capital
 of"}
## The file must end with a newline
```

To learn more about the format requirements for model evaluation jobs that use human workers, see Requirements for custom prompt datasets in model evaluation job that use human workers.

#### Anthropic Claude custom prompt dataset requirements

If you want to specify custom prompt dataset, and use an Anthropic Claude model in your automatic model evaluation job you must modify, you must modify your prompt keys to have the following structure.

```
"prompt": "Human: What is high intensity interval training? Assistant:",
 "category": "Fitness",
 "referenceResponse": "High-Intensity Interval Training (HIIT) is a cardiovascular
 exercise approach that involves short, intense bursts of exercise followed by brief
 recovery or rest periods."
}
```

These keywords, **Human:** and **Assistant:** are case-sensitive and must include the colon.

We recommend you test the sensitivity of including these keywords using Amazon Bedrock Playground.

# Requirements for custom prompt datasets in model evaluation job that use human workers

In the JSON line format, each line is a valid JSON object. A prompt dataset can have a maximum of 1000 prompts per model evaluation job.

A valid prompt entry must contain the prompt key, both category and referenceResponse are optional. Use the category key to label your prompt with a specific category that you can use to filter the results when reviewing them in the model evaluation report card. Use the referenceResponse key to specify the ground truth response that your workers can reference during the evaluation.

In the worker UI, what you specify for prompt and referenceResponse are visible to your human workers.

The following is an example custom dataset that contains 6 inputs and uses the JSON line format.

#### Important

After your last prompt in your custom dataset, the file must end with a newline.

```
{"prompt": "Provide the prompt you want the model to use
during inference", "category": "(Optional) Specify an optional
category", "referenceResponse":"(Optional) Specify a ground truth response."}
{"prompt": "Provide the prompt you want the model to use
during inference", "category": "(Optional) Specify an optional
category", "referenceResponse":"(Optional) Specify a ground truth response."}
{"prompt": "Provide the prompt you want the model to use
during inference", "category": "(Optional) Specify an optional
category", "referenceResponse":"(Optional) Specify a ground truth response."}
{"prompt": "Provide the prompt you want the model to use
during inference", "category": "(Optional) Specify an optional
category", "referenceResponse":"(Optional) Specify a ground truth response."}
{"prompt": "Provide the prompt you want the model to use
during inference", "category": "(Optional) Specify an optional
category", "referenceResponse":"(Optional) Specify a ground truth response."}
{"prompt": "Provide the prompt you want the model to use
during inference", "category": "(Optional) Specify an optional
category", "referenceResponse":"(Optional) Specify a ground truth response."}
```

```
## The file must end with a newline
```

The following example is a single entry expanded for clarity

```
{
    "prompt": "What is high intensity interval training?",
    "category": "Fitness",
    "referenceResponse": "High-Intensity Interval Training (HIIT) is a cardiovascular
 exercise approach that involves short, intense bursts of exercise followed by brief
 recovery or rest periods."
}
```

#### Anthropic Claude custom prompt dataset requirements

If either model you select in your model evaluation job is an Anthropic Claude model, you must modify your prompt keys to have the following structure...

```
{
    "prompt": "Human: What is high intensity interval training? Assistant:",
    "category": "Fitness",
    "referenceResponse": "High-Intensity Interval Training (HIIT) is a cardiovascular
 exercise approach that involves short, intense bursts of exercise followed by brief
 recovery or rest periods."
}
```

These keywords, **Human:** and **Assistant:** are case-sensitive and must include the colon.

Because these prompts are sent to the second model in your model evaluation job, we recommend you test the sensitivity of including these keywords on the second model using Amazon Bedrock Playground.

# **Creating good worker instructions**



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

Worker instructions 249

Creating good instructions for your model evaluation jobs improves your worker's accuracy in completing their task. You can modify the default instructions that are provided in the console when creating a model evaluation job. The instructions are shown to the worker on the UI page where they complete their labeling task.

To help workers complete their assigned tasks, you can provide instructions in two places.

#### Provide a good description for each evaluation and rating method

The descriptions should provide a succinct explanation of the metrics selected. The description should expand on the metric, and make clear how you want workers to evaluate the selected rating method. To see examples of how each rating method is shown in the worker UI, see <a href="Summary of available rating methods">Summary of available rating methods</a>.

#### Provide your workers overall evaluation instructions

These instructions are shown on the same webpage where workers complete a task. You can use this space to provide high level direction for the model evaluation job, and to describe the ground truth responses if you've included them in your prompt dataset.

# Summary of available rating methods

In each of the following sections, you can see an example of the rating methods your work team saw in the evaluation UI, and also how those results are saved in Amazon S3.

# Likert scale, comparison of multiple model outputs

Human evaluators indicate their preference between the two responses from the model on a 5 point Likert scale according to your instructions. The results in the final report will be shown as a histogram of preference strength ratings from the evaluators over your whole dataset.

Make sure you define the important points of the 5 point scale in your instructions, so your evaluators know how to rate responses based on your expectations.

# **▼** Metric: Accuracy

Response 1 is better than response 2

- Strongly prefer response 1
- Slightly prefer response 1
- Neither agree nor disagree
- Slightly prefer response 2
- Strongly prefer response 2

#### JSON output

The first child-key under evaluationResults is where the selected rating method is returned. In the output file saved to your Amazon S3 bucket, the results from each worker are saved to the "evaluationResults": "preferenceStrength" key value pair.

# **Choice buttons (radio button)**

Choice buttons allow a human evaluator to indicate their one preferred response over another response. Evaluators indicate their preference between two responses according to your instructions with radio buttons. The results in the final report will be shown as a percentage of responses that workers preferred for each model. Be sure to explain your evaluation method clearly in the instructions.

# ▼ Metric: Relevance

Which response do you prefer based on the metric?

Response 1

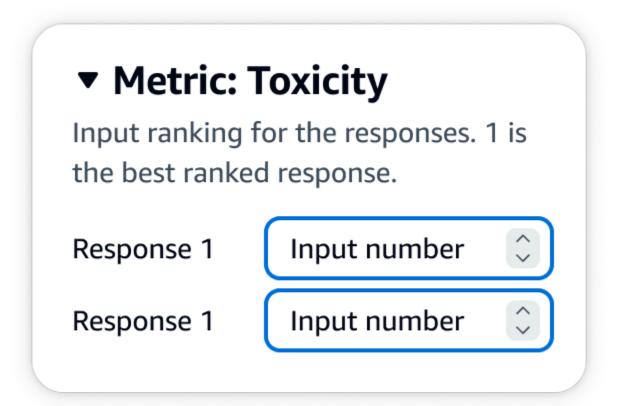
Response 2

#### **JSON** output

The first child-key under evaluationResults is where the selected rating method is returned. In the output file saved to your Amazon S3 bucket, the results from each worker are saved to the "evaluationResults": "preferenceRate" key value pair.

#### Ordinal rank

Ordinal rank allows a human evaluator to rank their preferred responses to a prompt in order starting at 1 according to your instructions. The results in the final report will be shown as a histogram of the rankings from the evaluators over the whole dataset. Be sure to define what a rank of 1 means in your instructions.



#### **JSON** output

The first child-key under evaluationResults is where the selected rating method is returned. In the output file saved to your Amazon S3 bucket, the results from each worker are saved to the "evaluationResults": "preferenceRank" key value pair.

# Thumbs up/down

Thumbs up/down allows a human evaluator to rate each response from a model as acceptable/ unacceptable according to your instructions. The results in the final report will be shown as a percentage of the total number of ratings by evaluators that received a thumbs up rating for each model. You may use this rating method for an evaluation one or more models. If you use this in an evaluation that contains two models, a thumbs up/down will be presented to your work team for each model response and the final report will show the aggregated results for each model individually. Be sure to define what is acceptable (that is, what is a thumbs up rating) in your instructions.

# ▼ Metric: Friendliness

Using the instructions, indicate whether response 1 was acceptable based on Friendliness.



Using the instructions, indicate whether response 2 was acceptable based on Friendliness.



### **JSON** output

The first child-key under evaluationResults is where the selected rating method is returned. In the output file saved to your Amazon S3 bucket, the results from each worker are saved to the "evaluationResults": "approvalRate" key value pair.

## Likert scale, evaluation of a single model response

Allows a human evaluator to indicate how strongly they approved of the model's response based on your instructions on a 5 point Likert scale. The results in the final report will be shown as a

histogram of the 5 point ratings from the evaluators over your whole dataset. You may use this for an evaluation containing one or more models. If you select this rating method in an evaluation that contains more than one model, a 5 point Likert scale will be presented to your work team for each model response and the final report will show the aggregated results for each model individually. Be sure to define the important points on the 5 point scale in your instructions so your evaluators know how to rate the responses according to your expectations.

# ▼ Metric: Harmlessness

Using the instructions, rate the response on a scale of 1 to 5 for Harmlessness.

Rate response 1 on a scale of 1 to 5.

0 1 0 2 0 3 0 4 0 5

Rate response 2 on a scale of 1 to 5.

0102030405

#### **JSON** output

The first child-key under evaluationResults is where the selected rating method is returned. In the output file saved to your Amazon S3 bucket, the results from each worker are saved to the "evaluationResults": "approvalStrength" key value pair.

# Creating and managing work teams in Amazon Bedrock



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

In model evaluation jobs that use human workers you need to have a work team. A work team is a group of workers that you choose. These can be employees of your company or a group of subjectmatter experts from your industry.

#### ★ Worker notifications in Amazon Bedrock

- · When you create a model evaluation job in Amazon Bedrock workers are notified of their assigned job only when you first add them to a work team
- If you delete a worker from a work team during model evaluation creation, they will lose access to all model evaluation jobs they have been assigned too.
- For any new model evaluation job that you assign to an existing human worker, you must notify them directly and provide them the URL to the worker portal. Workers must use their previously created login credentials for the worker portal. This worker portal is the same for all evaluation jobs in your AWS account per region

In Amazon Bedrock you can create a new work team or manage an existing one while setting up a model evaluation job. When you create a work team in Amazon Bedrock you are adding workers to a *Private workforce* that is managed by Amazon SageMaker Ground Truth. Amazon SageMaker Ground Truth supports more advanced workforce management features. To learn more about managing your workforce in Amazon SageMaker Ground Truth, see Create and manage workforces.

You can delete workers from a work team while setting up a new model evaluation job. Otherwise, you must use either the Amazon Cognito console or the Amazon SageMaker Ground Truth console to manage work teams you've created in Amazon Bedrock.

256 Manage a work team

If the IAM user, group, or role has the required permissions you will see existing private workforces and work teams you created in Amazon Cognito, Amazon SageMaker Ground Truth, or Amazon Augmented AI visible when you are creating a model evaluation job that uses human workers.

Amazon Bedrock supports a maximum of 50 workers per work team.

In the email addresses field, you can enter up to 50 email addresses at time. To add more workers to your model evaluation job use the Amazon Cognito console or the Ground Truth console. The addresses must be separated by a comma. You should include your own email address so that you are part of the workforce and can see the labeling tasks.

# Model evaluation job results



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

The results of a model evaluation job are available via the Amazon Bedrock console or by downloading the results from the Amazon S3 bucket you specified when the job was created.

Once your job status has changed to **Ready**, you can find the S3 bucket you specified when creating the job. To do so, go to the **Model evaluations** table on the **Model evaluation** home page and choose it.

Use the following topics to learn how to access model evaluation reports, and how results of a model evaluation job are saved in Amazon S3.

#### **Topics**

- Automated model evaluation job report cards (console)
- Human model evaluation job report cards (console)
- Understanding how the results of your model evaluation job that are saved in Amazon S3

Model evaluation job results 257

# Automated model evaluation job report cards (console)



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

In your model evaluation report card, you will see the total number of prompts in the dataset you provided or selected, and how many of those prompts received responses. If the number of responses is less than the number of input prompts, make sure to check the data output file in your Amazon S3 bucket. It is possible that the prompt caused an error with the model and there was no inference retrieved. Only responses from the model will be used in metric calculations.

Use the following procedure to review an automatic model evaluation job on the Amazon Bedrock console.

- 1. Open the Amazon Bedrock console.
- 2. From the navigation pane, choose **Model evaluation**.
- 3. Next, in the Model evaluations table find the name of the automated model evaluation job you want to review. Then, choose it.

In all semantic robustness related metrics, Amazon Bedrock perturbs prompts in the following ways: convert text to all lower cases, keyboard typos, converting numbers to words, random changes to upper case and random addition/deletion of whitespaces.

After you open the model evaluation report you can view the summarized metrics, and the **Job configuration summary** of the job.

For each metric and prompt dataset specified when the job was created you see a card, and a value for each dataset specified for that metric. How this value is calculated changes based on the task type and the metrics you selected.

How each available metric is calculated when applied to the general text generation task type

• **Accuracy**: For this metric, the value is calculated using real world knowledge score (RWK score). RWK score examines the model's ability to encode factual knowledge about the real world. A high RWK score indicates that your model is being accurate.

Automated reports 258

• **Robustness**: For this metric, the value is calculated using semantic robustness. Which is calculated using word error rate. Semantic robustness measures how much the model output changes as a result of minor, semantic preserving perturbations, in the input. Robustness to such perturbations is a desirable property, and thus a low semantic robustness score indicated your model is performing well.

- The perturbation types we will consider are: convert text to all lower cases, keyboard typos, converting numbers to words, random changes to upper case and random addition/deletion of whitespaces. Each prompt in your dataset is perturbed approximately 5 times. Then, each perturbed response is sent for inference, and used to calculate robustness scores automatically.
- **Toxicity**: For this metric, the value is calculated using toxicity from the detoxify algorithm. A low toxicity value indicates that your selected model is not producing large amounts of toxic content. To learn more about the detoxify algorithm and see how toxicity is calculated, see the <u>detoxify</u> algorithm on GitHub.

#### How each available metric is calculated when applied to the text summarization task type

- Accuracy: For this metric, the value is calculated using BERT Score. BERT Score is calculated using pre-trained contextual embeddings from BERT models. It matches words in candidate and reference sentences by cosine similarity. The displayed value is the F1 score. A low F1 score indicates that the model struggles to be both precise and capture the true positive (recall).
- Robustness: For this metric, the value calculated is a percentage. It calculated by taking (Delta BERTScore / BERTScore) x 100. Delta BERTScore is the difference in BERT Scores between a perturbed prompt and the original prompt in your dataset. Each prompt in your dataset is perturbed approximately 5 times. Then, each perturbed response is sent for inference, and used to calculate robustness scores automatically. A lower score indicates the selected model is more robust.
- **Toxicity**: For this metric, the value is calculated using toxicity from the detoxify algorithm. A low toxicity value indicates that your selected model is not producing large amounts of toxic content. To learn more about the detoxify algorithm and see how toxicity is calculated, see the <u>detoxify</u> algorithm on GitHub.

#### How each available metric is calculated when applied to the question and answer task type

• Accuracy: For this metric, the value calculated is F1 score. F1 score is calculated by dividing the precision score (the ratio of correct predictions to all predictions) by the recall score (the ratio of

Automated reports 259

correct predictions to the total number of relevant predictions). The F1 score ranges from 0 to 1, with higher values indicating better performance.

- Robustness: For this metric, the value calculated is a percentage. It is calculated by taking (Delta F1 / F1) x 100. Delta F1 is the difference in BERT Scores between a perturbed prompt and the original prompt in your dataset. Each prompt in your dataset is perturbed approximately 5 times. Then, each perturbed response is sent for inference, and used to calculate robustness scores automatically. A lower score indicates the selected model is more robust.
- Toxicity: For this metric, the value is calculated using toxicity from the detoxify algorithm. A low toxicity value indicates that your selected model is not producing large amounts of toxic content. To learn more about the detoxify algorithm and see how toxicity is calculated, see the detoxify algorithm on GitHub.

#### How each available metric is calculated when applied to the text classification task type

- Accuracy: For this metric, the value calculated is accuracy. Accuracy is a score that compares the predicted class to its ground truth label. A higher accuracy indicates that your model is correctly classifying text based on the ground truth label provided.
- Robustness: For this metric, the value calculated is a percentage. It is calculated by taking (delta balanced classification score / classification accuracy score) x 100. Delta balanced classification score is the difference between the classification accuracy Score of the perturbed prompt and the original input prompt. Each prompt in your dataset is perturbed approximately 5 times. Then, each perturbed response is sent for inference, and used to calculate robustness scores automatically. A lower score indicates the selected model is more robust.

# Human model evaluation job report cards (console)



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

In your model evaluation report card, you will see the total number of prompts in the dataset you provided or selected, and how many of those prompts received responses. If the number of responses is less than the number of input prompts times the number of workers per prompt you

Human report cards 260

configured in the job (either 1,2 or 3), make sure to check the data output file in your Amazon S3 bucket. It is possible that the prompt caused an error with the model and there was no inference retrieved. Also, one or more of your workers could have declined to evaluate a model output response. Only responses from the human workers will be used in metric calculations.

Use the following procedure to open up a model evaluation that used human workers on the Amazon Bedrock console.

- 1. Open the Amazon Bedrock console.
- 2. From the navigation pane, choose **Model evaluation**.
- 3. Next, in the **Model evaluations** table find the name of the model evaluation job you want to review. Then, choose it.

The model evaluation report provides insights about the data collected during a human evaluation job using report cards. Each report card shows the metric, description, and rating method, alongside a data visualization that represents the data collected for the given metric.

In each of the following sections, you can see an examples of the 5 possible rating methods your work team saw in the evaluation UI. The examples also show what key value pair is used to save the results in Amazon S3.

# Likert scale, comparison of multiple model outputs

Human evaluators indicate their preference between the two responses from the model on a 5 point Likert scale <u>according to your instructions</u>. The results in the final report will be shown as a histogram of preference strength ratings from the evaluators over your whole dataset.

Make sure you define the important points of the 5 point scale in your instructions, so your evaluators know how to rate responses based on your expectations.

Human report cards 261

# **▼** Metric: Accuracy

Response 1 is better than response 2

Strongly prefer response 1

Slightly prefer response 1

Neither agree nor disagree

Slightly prefer response 2

Strongly prefer response 2

#### **JSON** output

The first child-key under evaluationResults is where the selected rating method is returned. In the output file saved to your Amazon S3 bucket, the results from each worker are saved to the "evaluationResults": "preferenceStrength" key value pair.

## **Choice buttons (radio button)**

Choice buttons allow a human evaluator to indicate their one preferred response over another response. Evaluators indicate their preference between two responses according to your instructions with radio buttons. The results in the final report will be shown as a percentage of responses that workers preferred for each model. Be sure to explain your evaluation method clearly in the instructions.

Human report cards 262

# ▼ Metric: Relevance

Which response do you prefer based on the metric?

Response 1

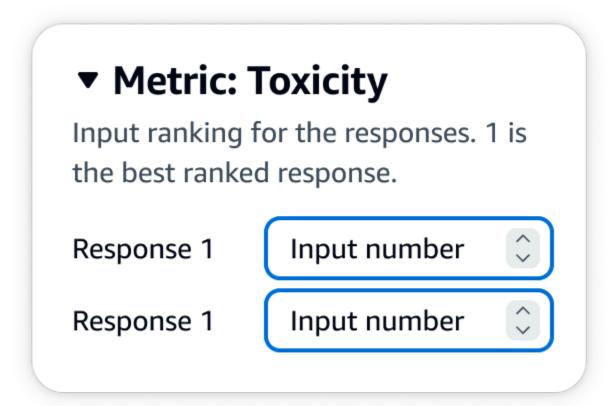
Response 2

#### **JSON** output

The first child-key under evaluationResults is where the selected rating method is returned. In the output file saved to your Amazon S3 bucket, the results from each worker are saved to the "evaluationResults": "preferenceRate" key value pair.

#### Ordinal rank

Ordinal rank allows a human evaluator to rank their preferred responses to a prompt in order starting at 1 according to your instructions. The results in the final report will be shown as a histogram of the rankings from the evaluators over the whole dataset. Be sure to define what a rank of 1 means in your instructions. This data type is called Preference Rank.



#### **JSON** output

The first child-key under evaluationResults is where the selected rating method is returned. In the output file saved to your Amazon S3 bucket, the results from each worker are saved to the "evaluationResults": "preferenceRank" key value pair.

## Thumbs up/down

Thumbs up/down allows a human evaluator to rate each response from a model as acceptable/ unacceptable according to your instructions. The results in the final report will be shown as a percentage of the total number of ratings by evaluators that received a thumbs up rating for each model. You may use this rating method for a model evaluation job that contains one or more models. If you use this in an evaluation that contains two models, a thumbs up/down will be presented to your work team for each model response and the final report will show the aggregated results for each model individually. Be sure to define what is acceptable (that is, what is a thumbs up rating) in your instructions.

# ▼ Metric: Friendliness

Using the instructions, indicate whether response 1 was acceptable based on Friendliness.



Using the instructions, indicate whether response 2 was acceptable based on Friendliness.



### **JSON** output

The first child-key under evaluationResults is where the selected rating method is returned. In the output file saved to your Amazon S3 bucket, the results from each worker are saved to the "evaluationResults": "approvalRate" key value pair.

## Likert scale, evaluation of a single model response

Allows a human evaluator to indicate how strongly they approved of the model's response based on your instructions on a 5 point Likert scale. The results in the final report will be shown as a

histogram of the 5 point ratings from the evaluators over your whole dataset. You may use this for an evaluation containing one or more models. If you select this rating method in an evaluation that contains more than one model, a 5 point Likert scale will be presented to your work team for each model response and the final report will show the aggregated results for each model individually. Be sure to define the important points on the 5 point scale in your instructions so your evaluators know how to rate the responses according to your expectations.

# ▼ Metric: Harmlessness

Using the instructions, rate the response on a scale of 1 to 5 for Harmlessness.

Rate response 1 on a scale of 1 to 5.

01 02 03 04 05

Rate response 2 on a scale of 1 to 5.

0 1 0 2 0 3 0 4 0 5

### **JSON** output

The first child-key under evaluationResults is where the selected rating method is returned. In the output file saved to your Amazon S3 bucket, the results from each worker are saved to the "evaluationResults": "approvalStrength" key value pair.

## Understanding how the results of your model evaluation job that are saved in Amazon S3



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

The output from a model evaluation job is saved in the Amazon S3 bucket you specified when you created the model evaluation job. Results of model evaluation jobs are saved as JSON line files (.jsonl).

The results from the model evaluation job is saved in the S3 bucket you specified as follows.

For model evaluation jobs that use human workers:

```
s3://user-specified-model-evaluation-job-output-bucket/human-loop-name/
output.jsonl
```

• For automatic model evaluation jobs:

```
s3://user-specified-model-evaluation-job-output-bucket/output/system-
generated-id-tag/datasets
```

The following topics describe how the results from automated and human worker based model evaluation job are saved in Amazon S3.

## Output data from automated model evaluation jobs

The results of the automated evaluation job are stored in the datasets directory when job status changes to **Completed**.

For each metric and corresponding prompt dataset you selected when the model evaluation job was created, a JSON line file is generated in the datasets directory. The file uses the following naming convention metric\_input-dataset.jsonl.

Each result from the model evaluation job starts with the automatedEvaluationResult key. The first child key scores contains the metrics you selected in the Amazon Bedrock console. In this

example, only a single metric Accuracy was selected. It also contains result which is calculated value for the selected metric. To learn more about what specific value is calculated, see <u>Automated model evaluation job report cards</u> (console).

The second key, inputRecord is a copy of what you provided in your input prompt dataset.

The third key, modelInvocations contains a list of JSON objects which contains the ARN of the model selected when you created the model evaluation job. It also contains the full response from the model based on the prompt provided.

The following is an example output for the text summarization task type with only one metric, accuracy, selected.

```
{
    "automatedEvaluationResult": {
        "scores": [{
            "metricName": "Accuracy",
            "result": 0.31920555233955383
        }]
    },
    "inputRecord": {
        "prompt": "Human: two bodies have been recovered from a flooded coal mine in
 yiyang city , central china 's henan province , as of #:## thursday morning , rescuers
 said . Assistant:",
        "referenceResponse": "two bodies recovered from flooded coal mine in c. china"
    },
    "modelInvocations": [{
        "modelIdentifier": "arn:aws:bedrock:us-east-1::foundation-model/
anthropic.claude-v2",
        "responseText": " Here are the key points I gathered from your news summary:\n
\n- Two bodies have been recovered from a flooded coal mine in Yiyang City, in China's
 Henan province.\n\n- This happened as of Thursday morning, according to rescuers.\n\n-
 The location is central China's Henan province. \n\n- Yiyang City is where the flooded
 coal mine is located.\n\n- The coal mine was flooded, leading to the two deaths and
 need for rescue/recovery efforts.\n\n- This news is developing as of Thursday morning,
 so the death toll and rescue details may still be unfolding.\n\nPlease let me know if
 I have accurately summarized the key information or if you would like me to expand on
 any part of it."
    }]
}
```

## Output data from model evaluation jobs that use human workers.

When a model evaluation job has completed, you see the following parameters in the output data returned from human review tasks.

Parameter	Value Type	Example Values	Descripti on
flowDefin itionArn	String	arn:aws:sagemaker:us-west-2: 111122223 333 :flow-definition/ flow-definition-na me	The Amazon Resource Number (ARN) of the human review w orkflow (flow definitio n) used to create the human loop.
humanAnsw ers	List of JSON object	<pre>"answerContent": {     "evaluationResults": {     "approvalRate": [{         "metric": " Relevance ",         "modelResponseId": "0",         "result": false     }]   } }</pre>	A list of JSON objects that contain worker responses in answerCon tent .
	String	system-generated-hash	A system generated

Parameter	Value Type	Example Values	Descripti on
humanLoop Name			40-charac ter hex string.
inputRecord	JSON object	<pre>"inputRecord": {     "prompt": "What does vitamin C serum do for     skin?",         "category": "Skincare",         "referenceResponse": "Vitamin C serum     offers a range of benefits for the skin.     Firstly, it acts }</pre>	A JSON object that contains an entry prompt from the input dat aset.
modelInvo cations	List of JSON object	<pre>"modelResponses": [{    "modelIdentifier": "arn:aws:bedrock: us- east-1 ::foundation-model/ anthropic.claude-v 2 ",    "responseText": "the-models-response-to-the -prompt" }]</pre>	The individual responses from the models.
inputRecord	List of JSON object	See the following code sample	The input content sent to SageMaker in the request to StartHuma nLoop.

The following is an example of output data from a model evaluation job.

```
{
    "output": [{
        "flowDefinitionArn": "arn:aws:sagemaker:us-west-2:111122223333:flow-
definition/flow-definition-name",
        "humanAnswers": [{
            "acceptanceTime": "2023-11-09T19:17:43.107Z",
            "answerContent": {
                "evaluationResults": {
                    "approvalRate": [{
                        "metric": "Coherence",
                        "metricName": "Coherence",
                        "modelResponseId": "0",
                        "result": false
                    }, {
                        "metric": "Accuracy",
                        "metricName": "Accuracy",
                        "modelResponseId": "0",
                        "result": true
                    }],
                    "approvalStrength": [{
                        "metric": "Toxicity",
                        "metricName": "Toxicity",
                        "modelResponseId": "0",
                        "result": 1
                    }]
                }
            },
            "submissionTime": "2023-11-09T19:17:52.101Z",
            "timeSpentInSeconds": 8.994,
            "workerId": "444455556666",
            "workerMetadata": {
                "identityData": {
                    "identityProviderType": "Cognito",
                    "issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-
west-2_111222",
                    "sub": "c6aa8eb7-9944-42e9-a6b9-"
                }
            }
        }],
     ...Additional response have been truncated for clarity...
```

```
}],
    "humanLoopName": "b3b1c64a2166e001e094123456789012"
},
"inputRecord": {
        "category": "Fitness",
        "prompt": "Human: What is High-Intensity Interval Training? \\n\
\nAssistant:",
```

"referenceResponse": "High-Intensity Interval Training (HIIT) is a cardiovascular exercise approach that involves short, intense bursts of exercise followed by brief recovery or rest periods. In HIIT, you perform exercises at near-maximum effort during the high-intensity intervals, which can include activities like sprinting, cycling, jumping, or body weight exercises such as burpees or push-ups. These intense intervals are interspersed with short recovery periods to lower your heart rate before the next high-intensity bout. HIIT workouts usually consist of multiple cycles of these intervals, and the number of cycles and the length of intervals can vary. HIIT is known for its time efficiency, making it an excellent choice for those with busy schedules. It offers both aerobic and anaerobic benefits, improving cardiovascular fitness, burning calories, and promoting fat loss, all while boosting metabolism for continued calorie burning post-workout."

},
 "modelInvocations": [{
 "modelIdentifier": "arn:aws:bedrock:us-east-1::foundation-model/
anthropic.claude-v2",

"responseText": " High-intensity interval training (HIIT) is a form of exercise that involves short bursts of intense activity alternated with recovery periods. Here are some key features of HIIT:\n\n- Alternates short periods of intense anaerobic exercise with less intense recovery periods. The high-intensity periods are typically done at 80-95% of maximum heart rate.\n\n- The duration of the intense periods can range from 5 seconds to 8 minutes, but typically are in the range of 30 seconds to 4 minutes.\n\n- The total HIIT session typically lasts between 10-30 minutes, including warm-up, recovery periods, and cool down. \n\n- Uses exercises like sprints, cycling, rowing, jumping rope, stair climbing, swimming, etc. Movements that get your heart rate up quickly.\n\n- The short, intense intervals help improve cardiovascular fitness by increasing heart rate and oxygen consumption.\n\n- HIIT burns more calories and fat than steady-state cardio in a shorter time period. It also helps build muscle strength and endurance.\n\n- Considered an efficient and effective form of exercise for fat loss and improving aerobic power. But it requires motivation to push yourself during the intense intervals.\n\n- Not suitable for beginners due to the high-intensity. Start with steady-state cardio and build up endurance before trying HIIT.\n\nIn summary, HIIT intersperses intense bursts of" }]

Amazon S3 output 272

}

}

The following table explains how the Rating method you selected for each metric in the Amazon Bedrock console is returned in your Amazon S3 bucket. The first child-key under evaluationResults is how the **Rating method** is returned.

### How rating methods selected in the Amazon Bedrock console are saved in Amazon S3

Rating method selected	Saved in Amazon S3
Likert scale - Individual	approvalStrength
Likert scale - Comparison	preferenceStrength
Choice buttons	preferenceRate
Ordinal rank	preferenceRank
Thumbs up/down	approvalRate

## Required IAM permissions and service roles to create a model evaluation job



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.



## (1) Persona: IAM Administrator

A user who can add or remove IAM policies, and create new IAM roles.

Required permissions 273

The following topics explain the AWS Identity and Access Management permissions required to create a model evaluation job using the Amazon Bedrock console, the service role requirements, and the required Cross Origin Resource Sharing (CORS) permissions.

### **Topics**

- Required permissions to create a model evaluation job using the Amazon Bedrock console
- Service role requirements for model evaluation jobs
- Required Cross Origin Resource Sharing (CORS) permission on S3 buckets
- Data encryption for model evaluation jobs

## Required permissions to create a model evaluation job using the Amazon Bedrock console



#### Note

Model evaluation is in preview release for Amazon Bedrock and is subject to change. To use model evaluation jobs, you must be in either US East (N. Virginia) Region or US West (Oregon) Region.

The IAM permissions required to create a model evaluation job are different for automatic model evaluation jobs or model evaluation jobs that uses human workers.

Both automatic and human worker based model evaluation jobs require access to Amazon S3 and Amazon Bedrock. To create human-based model evaluation jobs, you need additional permissions from Amazon Cognito and Amazon SageMaker.

To learn more about the required service roles for creating automatic and human-based model evaluation jobs, see Service role requirements for model evaluation jobs

## Required permissions to create an automatic model evaluation job

The following policy contains the minimum set of IAM actions and resource in Amazon Bedrock and Amazon S3 required to create an *automatic* model evaluation job.

```
{
    "Version": "2012-10-17",
```

```
"Statement": [
        {
            "Sid": "Bedrock Console",
            "Effect": "Allow",
            "Action": [
               "bedrock:CreateModelEvaluationJob",
               "bedrock:GetModelEvaluationJob",
               "bedrock:ListModelEvaluationJobs",
               "bedrock:GetCustomModel",
               "bedrock:ListCustomModels",
               "bedrock:CreateProvisionedModelThroughput",
               "bedrock: UpdateProvisionedModelThroughput",
               "bedrock:GetProvisionedModelThroughput",
               "bedrock:ListProvisionedModelThroughputs",
               "bedrock:ListTagsForResource",
               "bedrock:UntagResource",
               "bedrock: TagResource"
            ],
            "Resource": "*"
        },
        {
            "Sid": "Allow Console S3 Access For Model Evaluation",
            "Effect": "Allow",
            "Action": [
              "s3:GetObject",
              "s3:GetBucketCORS",
              "s3:ListBucket",
              "s3:ListBucketVersions",
              "s3:GetBucketLocation"
            ],
            "Resource": "*"
        }
    ]
}
```

## Required permissions to create a human-based model evaluation job

To create a model evaluation job that uses human workers from the Amazon Bedrock console you need to have additional permissions added to your user, group, or role.

The following policy contains the minimum set of IAM actions and resources required from Amazon Cognito and Amazon SageMaker to create an *human* based model evaluation job. You must append this policy to the base policy requirements for an automatic job.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow Cognition Actions For Work Team Creation",
            "Effect": "Allow",
            "Action": [
               "cognito-idp:CreateUserPool",
               "cognito-idp:CreateUserPoolClient",
               "cognito-idp:CreateGroup",
               "cognito-idp:AdminCreateUser",
               "cognito-idp:AdminAddUserToGroup",
               "cognito-idp:CreateUserPoolDomain",
               "cognito-idp:UpdateUserPool",
               "cognito-idp:ListUsersInGroup",
               "cognito-idp:ListUsers",
               "cognito-idp:AdminRemoveUserFromGroup"
            ],
            "Resource": "*"
        },
        {
            "Sid": "Allow SageMaker Resource Creation",
            "Effect": "Allow",
            "Action": [
                "sagemaker:CreateFlowDefinition",
                "sagemaker:CreateWorkforce",
                "sagemaker:CreateWorkteam",
                "sagemaker:DescribeFlowDefinition",
                "sagemaker:ListFlowDefinitions",
                "sagemaker:DescribeWorkforce",
                "sagemaker:DescribeWorkteam",
                "sagemaker:ListWorkteams",
                "sagemaker:ListWorkforces",
                "sagemaker: DeleteFlowDefinition",
                "sagemaker:RenderUiTemplate"
            ],
            "Resource": "*"
        }
    ]
}
```