ModelArts

Model Evaluation

Issue	01
Date	2026-07-02

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are the property of Huawei Technologies Co., Ltd.
All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei Cloud and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Huawei Cloud Computing Technologies Co., Ltd.

Address: Huawei Cloud Data Center Jiaoxinggong Road
Qianzhong Avenue
Gui'an New District
Gui Zhou 550029
People's Republic of China

Website: https://www.huaweicloud.com/intl/en-us/

Contents

1 Live Comparison

Overview

A critical challenge in the industrialization of AI is selecting the most suitable model from a vast array of foundation models and fine-tuned versions. The live comparison feature provides an intuitive evaluation platform, allowing you to perform side-by-side benchmarking of different models using identical inputs.

Core functions and value:

- **Foundation model selection**: In the early stages of a project, compare the performance of different models like DeepSeek, Qwen3, and GLM to find the best fit for your business needs.
- **Fine-tuning validation**: By performing synchronous comparisons between a native foundation model and its fine-tuned version, visually verify whether domain-specific knowledge has been successfully injected or if any catastrophic forgetting has occurred.
- **Parameter strategy optimization (A/B testing)**: Compare output variations of the same model using different hyperparameters (such as **Temperature** or **Top_P**) to determine the optimal inference configuration.

Prerequisites

An inference service has been deployed. For details, see **Deploying a Model as a Real-Time Service**.

Constraints

- **Model type restrictions**: Currently, only text generation models in the LLM domain are supported for comparison. Models in other domains are not supported.
- **Quantity limits**: To ensure optimal frontend rendering performance and facilitate manual side-by-side evaluation, a maximum of three models can be compared simultaneously in a single task.
- **Timeout limits**: During a real-time comparison task, if a model fails to complete its response within 5 minutes due to reasoning latency or performance issues, the corresponding window will trigger a timeout alert and terminate the generation.

- **Chat history limits**: The system automatically saves the 100 most recent chat history records for each IAM user within the same workspace. When the number of records exceeds 100, the oldest record is immediately deleted to maintain the 100-record limit. Additionally, for chat histories older than 7 days, the system automatically deletes the oldest records daily at 00:00.

Procedure

1. Log in to the **ModelArts console**. In the navigation pane on the left, choose **Model Evaluation** > **Live Comparison**.

   NOTE

   There are multiple entry points for live comparison. In addition to the left navigation pane, you can also use the following methods:

   - In the navigation pane on the left, choose **Model Inference** > **Real-Time Inference**. Click **Live Comparison** in the **Operation** column on the right.
   - In the navigation pane on the left, choose **Model Inference** > **Real-Time Inference**. Click the target service name to go to the service details page. Click **Live Comparison** in the upper right corner.

2. In the upper right corner of the **Live Comparison** page, click **Service Comparison**. In the **Live Comparison | Select Service** dialog box, select one to three services as required and click **OK**.
3. Configure service parameters as required.

   Click next to the service name to set parameters like system persona, temperature, and top_p. This changes how random and diverse the output can be. To ensure a fair evaluation (controlled variables), keep the parameter configurations consistent across all selected services.

   For details about service parameters and typical scenarios, see **Service Parameter Configuration**.
4. Click a preset question in the middle of the page or enter a question in the text box. Click or press **Enter** on the keyboard to send the question. Press **Shift+Enter** to start a new line.
   - The system will send this question to all selected models.
   - Both single-turn Q&A and multi-turn dialogues in the current context are supported.

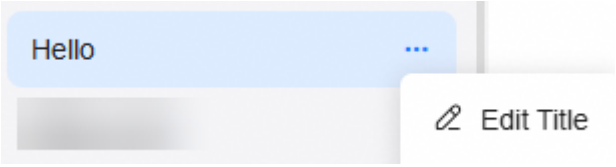
   If multiple models are selected for comparison, the system displays the generation results of each model in a side-by-side view. This allows you to evaluate and compare the models' performance regarding logical consistency, formatting accuracy, and semantic precision.

   The total time taken and the thinking time will show below the model's answer. For details, see **Metrics**.

   You can perform the following operations on the **Live Comparison** page:

   Table 1-1 Operations

Operation	Description
Switch services	Click the service name and select the target service in the **Switch services** pop-up window.

Operation	Description
Delete comparison services	Click × to the right of the service name to remove the service from the comparison.
Stop generating	While the model is responding, click **Stop response** in the input box to interrupt the response.
Regenerate response	Click ↻ below the model's response to regenerate it.
Copy response	Click ⧉ below the model's response to copy it.
Provide feedback	Click ✎ below the model's response to provide feedback on the output.
Start new chat	Click **New Chat** in the upper-right corner to clear the current conversation. You can then click **Service Comparison** in the upper-right corner to select services again and start a new chat.
Clear chat	Click **Clear Chat** in the upper-right corner to clear the context of the current conversation. Subsequent inputs will not be affected by previous turns.
View history	View your past conversations in the left pane. You can click any history record to resume the conversation and continue asking questions. By default, your first question is used as the title of that history record.
Edit history titles	In the left pane, click ··· to the right of a conversation title and choose **Edit Title**. In the **Edit Title** dialog box, modify the title as needed and click **OK**. **Figure 1-1** Editing a title 
Delete history	In the left pane, click ··· to the right of a conversation title and choose **Delete**. In the **Delete Chat** dialog box, click **OK** to delete all records of that conversation.

Service Parameter Configuration

When calling LLMs, you may find that the generated responses deviate significantly from your expectations. You can refine the model's output by adjusting decoding parameters to control its randomness and creativity. In essence, these parameters determine whether the model responds like a rigorous scientist or creates like a romantic poet.

Table 1-2 Service parameters

Parameter	Description	Example	Recommended Tuning Order
System persona	System persona of a custom model. Enter up to 1,000 characters.	You are a system AI assistant.	-
Temperature	Controls the randomness and creativity of the model's output. A higher temperature produces more unpredictable, more creative results. A lower temperature produces more predictable, more conservative results. • Low (0.1): The model is extremely conservative, always choosing the highest-probability tokens. Ideal for scenarios requiring factual accuracy. • High (0.9): The model becomes more "expressive," selecting lower-probability tokens. Ideal for creative tasks, though it may lead to hallucinations.	Prompt: Write a sentence using the word "sky." • Temperature = 0.1 (Conservative) – **Result**: The sky is blue with a few white clouds. – **Characteristics**: Accurate, straightforward, and highly reproducible. • Temperature = 0.9 (Creative) – **Result**: The sky resembled a jar of overturned blueberry jam, with stars floating within it. – **Characteristics**: Vivid, varied, and inconsistent across runs.	Primary adjustment

Parameter	Description	Example	Recommended Tuning Order
Top_P	Controls the diversity of the model's output. A larger value indicates stronger diversity of the generated text. Dynamically selects the top tokens based on cumulative probability. Higher values allow for a richer (though potentially rarer) vocabulary. Rather than a fixed **count**, **Top_P** uses a cumulative threshold. The model ranks tokens by probability and keeps only those whose sum reaches the P-value (e.g., **0.9**).	• **Top_P** = **0.1**: Only the most stable, top-tier tokens are considered. • **Top_P** = **0.9**: Allows long-tail vocabulary into the candidate pool, resulting in more diverse wording. Note: **Top_P** is dynamic. If the next set of tokens is certain, the pool is small; if uncertain, the pool expands. This is generally more intelligent than **Top-K**.	Secondary fine-tuning (used with **Temperature**)
Top_K	Controls the creativity or randomness of the generated text. A smaller K value produces smoother, more logically consistent sentences, but they may be dull or repetitive. A larger K value produces richer and more creative sentences, but also increases the chance of implausible words (hallucinations). Caps the candidate pool to a fixed number (*K*) of top-ranked tokens. Higher values keep more candidates.	• **Top-K** = **1**: Greedy decoding. The model only ever picks the top 1 candidate (equivalent to an extremely low **Temperature**). • **Top-K** = **50**: Typically used to prevent the model from generating low-probability gibberish or incoherent characters.	Supplementary parameter (usually kept at default or a high value)

The following table describes the parameter configurations for typical scenarios.

Table 1-3 Parameter configurations for typical scenarios

Application Scenario	Recommended Configuration	Desired Effect	Typical Use Case
Code generation Mathematical problem solving	Temp: 0.0 - 0.2 Top_P: 0.1	**Highly precise** Eliminates randomness to ensure logical correctness and strict grammatical adherence.	AI-assisted coding, SQL generation, and logical reasoning
Knowledge Q&A Customer service	Temp: 0.3 - 0.5 Top_P: 0.7	**Stable & natural** Ensures factual accuracy while maintaining a more human-like tone than a standard bot.	Intelligent customer service and RAG-based document QA
Copywriting Chit-chat	Temp: 0.7 - 0.9 Top_P: 0.9	**Rich & diverse** Utilizes a broad vocabulary and varied sentence structures to maximize creativity.	Marketing copy, creative writing/ story extension, and role-playing
Brainstorming	Temp: 1.0+ Top_P: 0.95	**Unconstrained:** Breaks away from conventional logic to find unexpected inspiration (requires manual filtering).	Creative ideation and naming

Metrics

Besides comparing texts subjectively, you can see technical metrics below the model's answer to help with quantitative evaluation.

Figure 1-2 Technical metrics

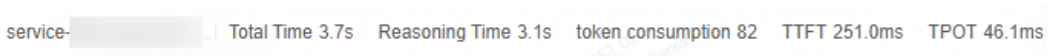


Table 1-4 Model metrics

Type	Name	Description
Performance	Total Time	Total time required to complete the entire response. A shorter duration indicates higher inference performance.
	Reasoning Time	Time spent on thinking.

Type	Name	Description
	TTFT	Time to first token (TTFT) is the time from when a user sends a question to when the first token of the AI's answer shows up on the screen. A lower TTFT indicates faster initial responsiveness.
	TPOT	Time per output token (TPOT) is the average time needed to create each token after the first one appears. A lower TPOT indicates faster and smoother text generation.
Consumption	token consumption	Displays the number of input tokens and output tokens for the session. This is used to estimate API call costs and resource consumption.

2 Model Evaluation

2.1 Model Evaluation Features

Description

Model evaluation is the process of testing and measuring the performance of a foundation model in real-world scenarios. It is crucial for understanding the performance of a foundation model.

A model with excellent performance must possess strong generalization capabilities. This means that the model should perform well not only on the provided data (training data) but also on unseen data. To achieve this goal, model evaluation is indispensable.

In the ModelArts model development process, model evaluation assesses new models after training. Only evaluated models can be deployed and used. This is a crucial step in the model development workflow.

Why Model Evaluation Matters

Model evaluation helps you identify the strengths and weaknesses of a model, ensuring its effectiveness in real-world applications and its ability to handle specific tasks while meeting relevant requirements.

When collecting evaluation datasets, ensure that the datasets are independent and random to guarantee that the collected data can represent a sample of real-world data. This helps to avoid biasing the evaluation result, thereby more accurately reflecting the performance of the model in different scenarios. By using the evaluation dataset to evaluate a model, developers can understand the advantages and disadvantages of the model and find the optimization direction.

Core value of model evaluation for developers:

- **Verify training effectiveness**: Measure the degree of capability improvement following fine-tuning or incremental pre-training.
- **Identify optimization paths**: Pinpoint model weaknesses in specific tasks to guide subsequent iterations.

- **Support deployment decisions**: Use quantitative metrics to determine if a model meets production standards.
- **Compare model selection**: Evaluate and select the most suitable model version for specific business scenarios from multiple candidates.
- **Ensure regulatory compliance**: Provide quantitative evidence of model capabilities to support auditing and compliance requirements.

Model Evaluation Scenario

Model evaluation primarily assesses a model's knowledge retention and text comprehension capabilities. These capabilities can be classified into general capabilities and industry-specific capabilities. The following sections describe the application scenarios of **general capability evaluation** and **industry capability evaluation**.

General Capability Evaluation

General capabilities: Primarily includes evaluation tasks using general-domain datasets, such as text classification, logical reasoning, sentiment analysis, and question-answering (QA) systems.

Typical scenarios

- Text classification accuracy evaluation
- Logical reasoning capability assessment
- Sentiment analysis accuracy evaluation
- Reading comprehension and QA system evaluation
- Text summarization quality assessment
- Machine translation fluency evaluation

Recommended dataset sources: ModelArts provides management features for open-source evaluation sets, enabling you to easily leverage these datasets for more precise and efficient LLM evaluations.

Industry-specific Capability Evaluation

Industry capabilities: Primarily focuses on evaluation tasks using domain-specific datasets, such as financial entity recognition, financial text classification, and debt collection intent recognition.

Typical scenarios

- Finance: Entity recognition, contract clause classification, and risk control intent recognition
- Healthcare: Medical Q&A, medical record summarization, and drug information extraction

Recommended dataset sources: Creating custom evaluation sets: To evaluate a model's domain-specific knowledge, you can use homologous datasets to build evaluation sets for tasks like entity recognition, text classification, or content generation. Use precision, recall, and F-score as the primary evaluation metrics.

Model Evaluation Types

ModelArts offers strong model evaluation features. It supports both human and automated evaluations.

Automated Evaluation

Automated evaluation: Supports two types: rule-based and LLM-based.

Rule-based: Automatically evaluates model-generated responses based on similarity or accuracy. You can use professional datasets pre-configured in evaluation templates or upload custom datasets.

Applicability: Closed-ended tasks with clear standard answers, such as classification, entity recognition, and multiple-choice QA.

Operation: The system automatically compares model outputs with reference answers in the dataset, calculating scores based on similarity algorithms or accuracy rules.

LLM-based: Uses an LLM to automatically score the outputs of the model under test. This is suitable for open-ended or complex QA scenarios and includes scoring mode and comparison mode.

Applicability: Open-ended tasks without a single correct answer, such as creative writing, open-ended QA, and dialogue generation.

Sub-modes: See **Table 2-1**.

Table 2-1 Table 1: LLM-based evaluation sub-modes

Sub-mode	Description	Typical Use Case
Scoring mode	Uses a judge LLM to provide multi-dimensional scores for a model's output.	Assessing the generation quality of a single model
Comparison mode	Uses a judge LLM to compare outputs from two models simultaneously and determine which is superior.	Model A/B testing and selection

Human Evaluation

Human evaluation: Evaluates model-generated responses using manually created datasets and specific evaluation criteria. During the process, human evaluators score the responses based on predefined metrics. Once completed, an evaluation report is generated based on these scores.

Applicability: Scenarios requiring subjective human judgment, such as dimensions like style, tone, professionalism, and safety, which are difficult to measure using automated rules.

Operation: Evaluate and score each data entry on the human evaluation page. Once all data has been reviewed, click the submit button to submit the results.

2.2 Creating a Model Evaluation Task

Prerequisites

- You have registered a Huawei ID and enabled Huawei Cloud services, performed real-name authentication, and ensure your account is not frozen or in arrears before using ModelArts. For details, see **Signing Up for a HUAWEI ID and Enabling Huawei Cloud Services** and **Real-Name Authentication Introduction**.
- You have configured an agency.

  Certain ModelArts functions require access to services like OBS. Before using ModelArts, ensure your account has been authorized to access these services.

Billing

If you store data in OBS, you will be billed for the storage resources used. For details, see **OBS Billing Overview**.

Constraints

Currently, only LLM evaluation is supported.

Creating a Rule-based Automated Evaluation Task

To create an automated evaluation task, follow these steps:

1. Log in to the **ModelArts console**.
2. In the navigation pane on the left, choose **Model Evaluation** > **Evaluation Tasks**. In the upper right corner of the **Automated Evaluation** tab, click **Create**.
3. On the displayed page, set parameters by referring to **Table 2-2**.

Table 2-2 Parameters for a rule-based automated evaluation task

Category	Parameter	Description
Basic Information	Task Name	Name of an evaluation task. The task name must start with a letter and end with a letter or digit. It can contain 2 to 32 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed.
	Description	Description of an evaluation task. This parameter is optional.

Category	Parameter	Description
Evaluation Object	Service Type	Currently, only the text generation type is supported.
	Add Service	Select a model deployed on ModelArts for evaluation. A maximum of 10 models can be evaluated at a time.
Evaluation Configuration	Evaluation Rule	Select **Rule-based**: Automatic scoring is performed based on rules. That is, scoring is performed based on similarity or accuracy, and the difference between the model's prediction and the labeled data is compared. It is applicable to standard multiple-choice questions or simple Q&A scenarios.
	Evaluation Datasets	• **Preset evaluation set**: Use a preset professional dataset for evaluation. • **Custom evaluation set**: Upload an evaluation dataset and select evaluation metrics (F1 score, accuracy, BLEU, and Rouge). Choose **Object Storage Service – Bucket** or **Object Storage Service – Parallel File System** as the storage type. Click to select an OBS storage address or manually enter an OBS storage address. The storage address must start with **obs://** or **/** and end with **.jsonl**. It cannot contain double slashes (**//**) except in the prefix. For example, **obs://bucketname/example.jsonl** or **/bucketname/example.jsonl**. Upload a single .jsonl file. The file size cannot exceed 10 MB, and the maximum number of records is 1,000. Example: `{"case_no":"1","corpus_no":"1","class_level1":"category1","class_level2":"category2","question":"hello","ref_answer":"What can I do for you today?","question_type":"common question type"}`

Category	Parameter	Description
	Custom Evaluation Metrics	This parameter is required only when **Evaluation Datasets** is set to **Custom evaluation set**. You can select evaluation metrics as required. • F1 score: The harmonic mean of precision and recall. A higher score indicates a better balance between precision and recall. • Accuracy: The percentage of correctly predicted samples (exact matches). A higher score indicates a higher proportion of correct predictions and better model performance. • BLEU: A metric used to measure the similarity between machine-translated text and reference text. A higher score indicates a better model machine translation. • ROUGE: A set of metrics for evaluating summarization and machine translation by comparing generated output to references. A higher score indicates a better model effect.
	Evaluation Results Storage	Path for storing the model evaluation result. Choose **Object Storage Service – Bucket** or **Object Storage Service – Parallel File System** as the storage type. Click to select an OBS storage address or manually enter an OBS storage address. The storage address must start with **obs://** or **/** and end with a slash (**/**). It cannot contain double slashes (**//**) except in the prefix. For example, **obs://bucketname/path/** or **/bucketname/path/**.

4. Click **Create Now**.
5. When the status is **Completed**, you can click **Report** in the **Operation** column to view the report and details of the evaluation job on the evaluation report page.

Creating an LLM-based Automated Evaluation Task

To create an automated evaluation task, follow these steps:

1. Log in to the **ModelArts console**.
2. In the navigation pane on the left, choose **Model Evaluation** > **Evaluation Tasks**. Click the automated evaluation tab and click **Create**.
3. On the **Create Auto-Evaluation Task** page, set parameters by referring to **Table 2-3**.

Table 2-3 Parameters for an LLM-based automated evaluation task

Category	Parameter	Description
Basic Information	Task Name	Name of an evaluation task. The task name must start with a letter and end with a letter or digit. It can contain 2 to 32 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed.
	Description	Description of an evaluation task. This parameter is optional.
Evaluation Object	Service Type	Currently, only the text generation type is supported.
	Add Service	Select a model deployed on ModelArts for evaluation. A maximum of 10 models can be evaluated at a time.
Evaluation Configuration	Evaluation Rule	Select **LLM-based**.
	Select Mode	• **Scoring**: The judge model will automatically score inference results based on preset criteria. • **Comparison**: The judge model compares each service against the baseline and outputs results as win, loss, or tie. At least 2 services must be selected for comparison mode.
	Evaluation Datasets	• **Preset evaluation set**: Use a preset professional dataset for evaluation. Only one preset evaluation set can be added. • **Custom evaluation set**: Choose **Object Storage Service – Bucket** or **Object Storage Service – Parallel File System** as the storage type. Click to select an OBS storage address or manually enter an OBS storage address. The storage address must start with **obs://** or **/** and end with **.jsonl**. It cannot contain double slashes (**//**) except in the prefix. For example, **obs://bucketname/example.jsonl** or **/bucketname/example.jsonl**. Upload a single .jsonl file. The file size cannot exceed 10 MB, and the maximum number of records is 1,000. Example: `{"case_no":"1","corpus_no":"1","class_level1":"category1","class_level2":"category2","question":"hello","ref_answer":"What can I do for you today?","question_type":"common question type"}`

Category	Parameter	Description
	Evaluation Results Storage	Path for storing the model evaluation result. Choose **Object Storage Service – Bucket** or **Object Storage Service – Parallel File System** as the storage type. Click to select an OBS storage address or manually enter an OBS storage address. The storage address must start with **obs://** or **/** and end with a slash (**/**). It cannot contain double slashes (**//**) except in the prefix. For example, **obs://bucketname/path/** or **/bucketname/path/**.
Judge Settings	Referee Model	Select a model deployed on ModelArts for evaluation.
	Scoring Rules	You can select a preset or custom scoring prompt template. Preset prompts cannot be modified. To create a custom prompt template, click **New** in the dialog box on the right of **Edit Custom Rule**, enter the name, persona, task description, whether to include questions, whether to include reference answers, scoring policy, and evaluation metrics, and click **Save**.

4. Click **Create Now**. A single tenant can create a maximum of 2,000 evaluation tasks.
5. When the status is **Completed**, you can click **Report** in the **Operation** column to view the report and details of the evaluation job on the evaluation report page.

Creating a Human Evaluation Task

To create a human evaluation task, follow these steps:

1. Log in to the **ModelArts console**.
2. In the navigation pane on the left, choose **Model Evaluation** > **Evaluation Tasks**. In the upper right corner of the **Human Evaluation** tab, click **Create**.
3. On the **Create Human Evaluation Task** page, set parameters by referring to **Table 2-4**.

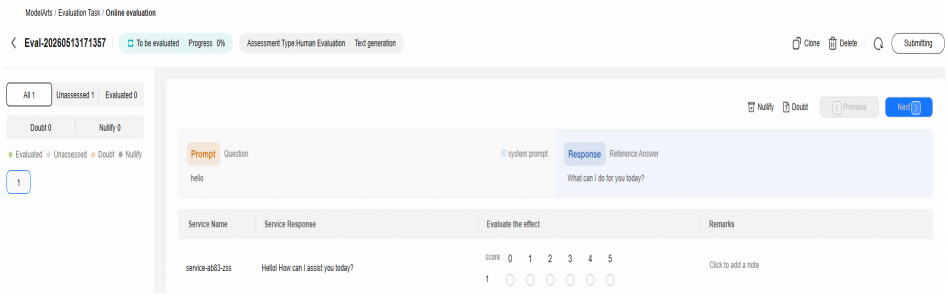
Table 2-4 Parameters for a human evaluation task

Category	Parameter	Description
Basic Information	Task Name	Name of an evaluation task. The task name must start with a letter and end with a letter or digit. It can contain 2 to 32 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed.
	Description	Description of an evaluation task. This parameter is optional.
Evaluation Object	Service Type	Currently, only the text generation type is supported.
	Add Service	Select a model deployed on ModelArts for evaluation. A maximum of 10 models can be evaluated at a time.
Evaluation Configuration	Evaluation Metrics	You can set up to six custom evaluation metrics and define their standards.
	Evaluation Datasets	Datasets for evaluation. Choose **Object Storage Service – Bucket** or **Object Storage Service – Parallel File System** as the storage type. Click to select an OBS storage address or manually enter an OBS storage address. The storage address must start with **obs://** or **/** and end with **.jsonl**. It cannot contain double slashes (**//**) except in the prefix. For example, **obs://bucketname/example.jsonl** or **/bucketname/example.jsonl**. Upload a single .jsonl file. The file size cannot exceed 10 MB, and the maximum number of records is 1,000. Example: {"case_no":"1","corpus_no":"1","class_level1":"category1","class_level2":"category2","question":"hello","ref_answer":"What can I do for you today?","question_type":"common question type"}
	Blind Test	When enabled, model names are hidden during scoring, and the order of models is mixed up.
	Evaluators	Only the assigned evaluators can score the task. The evaluation report is generated only after all evaluators have scored all cases.

Category	Parameter	Description
	Evaluation Results Storage	Path for storing the model evaluation result. Choose **Object Storage Service – Bucket** or **Object Storage Service – Parallel File System** as the storage type. Click to select an OBS storage address or manually enter an OBS storage address. The storage address must start with **obs://** or **/** and end with a slash (**/**). It cannot contain double slashes (**//**) except in the prefix. For example, **obs://bucketname/path/** or **/bucketname/path/**.

4. Click **Create Now**.
5. When the status changes to **Pending evaluation**, click **Evaluate Online** in the **Operation** column to go to the evaluation page.
6. Complete the evaluation as prompted. Then, click **Submit**.
   - Click **Doubt** or **Nullify** to doubt or nullify a case. To cancel this operation, click **Cancel Doubts** or **Cancel nullify**.
   - Score all evaluation metrics of the case and click **Save and Next** to save the scores and switch to the next case.
   - Click **Previous** to return to the previous user and rescore.
   - Click the **Click to add a note** area to add remarks.
   - On the evaluation page, hold down the left mouse button to select the text content to be marked and click **Mark** to mark the content as key content.

   Figure 2-1 Human evaluation



7. In the navigation pane, choose **Evaluation Platform** > **Evaluation task** > **Manual evaluation**. Click **Assessment report** in the **Operation** column to view the model evaluation result.

   After the evaluation is complete, go to the manual evaluation list page and click **Human review** to review the evaluation. After the review is complete, click **Submit** to submit the result.

2.3 Viewing the Model Evaluation Report

After an evaluation task is created, you can view the task report. The procedure is as follows:

1. Log in to the **ModelArts console**.
2. In the navigation pane, choose **Model Evaluation** > **Evaluation Tasks**.
3. Click **Report** in the **Operation** column. On the displayed page, you can view the basic information and overview of the evaluation task.

   For details about each evaluation metric, see **LLM evaluation metrics**.
4. Export the evaluation report.

   a. In the **Service Result Analysis** area in the **Report** tab, click **Export**, select the evaluation report you want to export, and click **OK**.

   b. Click **Export Record** on the right to view the exported task ID. Click **Download** in the **Operation** column to download the evaluation report to the local PC.

LLM evaluation metrics

LLMs support automated evaluation and human evaluation. For details about the metrics, see **Table 2-5**, **Table 2-6**, **Table 2-7**, and **Table 2-8**.

Table 2-5 Automated LLM evaluation metrics (not using preset evaluation sets)

Evaluation Metric (Automated Evaluation – Custom Evaluation Set)	Description
ACC	The percentage of correctly predicted samples (exact matches). A higher score indicates a higher proportion of correct predictions and better model performance.
F1 Score	The harmonic mean of precision and recall. A higher score indicates a better balance between precision and recall.
BLEU_1	Matching degree between the sentence generated by the model and the actual sentence at the single-word level. A larger score indicates better model effectiveness.
BLEU_2	Matching degree between the sentence generated by the model and the actual sentence at the phrase level. A larger score indicates better model effectiveness.
BLEU_4	Weighted average accuracy of the model generation result and actual sentences. A larger score indicates better model effectiveness.

Evaluation Metric (Automated Evaluation – Custom Evaluation Set)	Description
ROUGE_1	Recall rate calculated after the model generation result and labeling result are split by 1-gram (n-gram refers to a segment consisting of n consecutive words in a sentence). A larger score indicates better model effectiveness.
ROUGE_2	Recall rate calculated after the model generation result and labeling result are split by 2-gram (n-gram refers to a segment consisting of n consecutive words in a sentence). A larger score indicates better model effectiveness.
ROUGE_L	Recall rate calculated after the model generation result and labeling result are split by longest-gram (longest-gram refers to a segment consisting of n consecutive words in a sentence). A larger score indicates better model effectiveness.

Table 2-6 Automated LLM evaluation metrics (using preset evaluation sets)

Evaluation Metric (Automated Evaluation – Evaluation Template Used)	Description
Evaluation score	The score of each dataset is the pass rate of the model in the current dataset. If there are multiple datasets in the evaluation capability items, the weighted average pass rate is calculated based on the data volume.
Comprehensive Score	The comprehensive capability is the weighted average of the pass rates of all datasets.

Table 2-7 Human LLM evaluation metrics

Evaluation Metric (Human Evaluation)	Description
Accuracy	The answer generated by the model is correct and there is no factual error.

Evaluation Metric (Human Evaluation)	Description
average	The model calculates the average score of the generated sentence and the actual sentence based on the evaluation metric.
goodcase	The model calculates the proportion of test cases whose score is 5 after the generated sentence and the actual sentence are compared based on the evaluation metric.
badcase	The model calculates the proportion of test cases whose score is less than 1 after the generated sentence and the actual sentence are compared based on the evaluation metric.
Custom metrics	Custom metrics, such as usability, logic, and security.

Table 2-8 Automated LLM evaluation metrics

Model Type	Evaluation Metric (Automated Evaluation – Rule-based – LLM-based)	Description
LLM	Score given by the judge model	Score given by the judge model for each case in the dataset.
	Average	Average score of all test cases in the dataset.
	Mid	Median score of all test cases in the dataset.
	Standard deviation	Standard deviation of all test case scores in the dataset.
	win	Number of models whose performance metrics (the definition of "good" needs to be specified in advance, for example, high accuracy rate or low error rate) are better than those of the benchmark model among all comparison models.
	lose	Number of comparison models whose performance metrics are worse than those of the benchmark model.
	tie	Number of comparison models whose performance metrics are the same as those of the benchmark model.
	quantile	(win+tie)/(lose+tie)

Model Type	Evaluation Metric (Automated Evaluation – Rule-based – LLM-based)	Description
	Quantile (excluding tie_bad)	Exclude the tie_bad score: (win + tie_good)/(lose + tie_good)
	Quantile (excluding tie_good)	Exclude the tie_good score: (win + tie_bad)/(lose + tie_bad)

2.4 Managing Model Evaluation

2.4.1 Preset Evaluation Sets and Templates

Preset Evaluation Sets

Feature

A preset evaluation set is a collection of carefully designed, labeled, and standardized data samples used to test and measure AI model performance on specific tasks.

The following table lists the preset evaluation sets supported in this version and their descriptions.

Table 2-9 Preset evaluation sets

Name	Description
MMLU-Pro	MMLU is a key benchmark for testing LLMs. It includes 57 tasks, such as math, computer science, law, and history, to evaluate the models' world knowledge and problem-solving skills.
GPQA_Diamond	GPQA_Diamond is a multiple-choice question set created and checked by experts in biology, physics, and chemistry. It has 448 very hard questions. The set tests how well AI systems handle questions that mix different subjects, especially when they are outside the system's main area of expertise.
BoolQ	BoolQ is a set of "yes/no" questions. These questions come from real-world queries and have no specific guidelines, making them complex and varied.
AGIEval	The AGIEval benchmark features high-quality tests like the LSAT, college entrance exams (including China's and the US SAT), math competitions, and bar exams. The dataset only includes objective questions, such as multiple-choice and fill-in-the-blank. These tests measure a model's cognitive abilities, knowledge, and reasoning skills using official standards.

Name	Description
C-Eval	C-Eval is a Chinese dataset with 52 subjects and four difficulty levels. It tests how well LLMs understand Chinese.
GSM8K	GSM8K is a benchmark from OpenAI that tests how well LLMs can reason with data. It includes 8,500 elementary-level math problems to assess these models' math skills.
MathBench	It evaluates the math skills of LLMs, covering both theory and problem-solving.
ARC Challenge	ARC Challenge is a dataset for testing logical reasoning and problem-solving. It covers questions from different fields to assess advanced reasoning skills.
BBH	BBH is a big dataset with 204 tasks that test LLMs in areas like linguistics, child development, common sense, social bias, and software development. It checks how well these models tackle tough tasks.
CMMLU	CMMLU is the Chinese version of MMLU, covering areas like humanities, law, engineering, and math. It tests how well models know these subjects in Chinese.
OpenFinData	OpenFinData is an open-source financial evaluation dataset jointly released by EastMoney.com and Shanghai AI Lab. This dataset represents the most realistic industrial scenario needs and is currently the most comprehensive and professional financial evaluation dataset. It provides high-quality data resources for researchers and developers in the field of financial technology based on the diverse financial services of EastMoney.com.
FinEval	FinEval, a financial industry evaluation benchmark, is grounded in quantitative methodologies. Through long-term objective research and rigorous manual screening, it features over 26,000 diverse test items highly aligned with real-world application scenarios, including multiple-choice questions, subjective/objective short-answer questions, reasoning and planning tasks, and retrieval-augmented QA. Covering financial academic knowledge, financial industry expertise, financial safety regulations, financial agents, financial multimodality, and financial rigor, FinEval is designed to comprehensively evaluate the all-around application capabilities of LLMs within the financial sector.
MedMCQA	This large-scale MCQA dataset helps answer real-world medical school entrance exam questions.

Name	Description
PubMedQA	PubMedQA is a novel biomedical QA dataset collected from PubMed abstracts. The task of PubMedQA is to answer research questions with "yes/no/maybe" using the corresponding abstracts (e.g., Do preoperative statins reduce atrial fibrillation after coronary artery bypass grafting?). Each PubMedQA instance consists of four components: (1) a question, which is either an existing research article title or derived from it; (2) a context, which is the corresponding abstract without the conclusion; (3) a long answer, which is the conclusion of the abstract that typically answers the research question; and (4) a "yes/no/maybe" answer that summarizes the conclusion. PubMedQA is the first QA dataset that requires reasoning over biomedical research texts, especially their quantitative contents, to answer questions.

2.4.2 Managing Evaluation Tasks

In the evaluation task list, the task creator can clone (copy an evaluation task), start (restart a task), and delete a task.

1. Log in to the **ModelArts console**.
2. In the navigation pane, choose **Model Evaluation** > **Evaluation Tasks**. You can perform the following operations:
   - **Clone**: Click **Clone** in the **Operation** column to copy an evaluation task in the **Completed** state.
   - **Start**: Click **Start** in the **Operation** column to restart an evaluation task in the **Stopped** state.
   - **Delete**: Click **Delete** in the **Operation** column to delete an unnecessary evaluation task in the **Completed** state.

