

ModelArts

Entorno de desarrollo

Edición 01
Fecha 2024-09-20




Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2024. Todos los derechos reservados.

Quedan terminantemente prohibidas la reproducción y/o la divulgación totales y/o parciales del presente documento de cualquier forma y/o por cualquier medio sin la previa autorización por escrito de Huawei Cloud Computing Technologies Co., Ltd.

Marcas registradas y permisos



El logotipo  y otras marcas registradas de Huawei pertenecen a Huawei Technologies Co., Ltd. Todas las demás marcas registradas y los otros nombres comerciales mencionados en este documento son propiedad de sus respectivos titulares.

Aviso

Es posible que la totalidad o parte de los productos, las funcionalidades y/o los servicios que figuran en el presente documento no se encuentren dentro del alcance de un contrato vigente entre Huawei Cloud y el cliente. Las funcionalidades, los productos y los servicios adquiridos se limitan a los estipulados en el respectivo contrato. A menos que un contrato especifique lo contrario, ninguna de las afirmaciones, informaciones ni recomendaciones contenidas en el presente documento constituye garantía alguna, ni expresa ni implícita.

Huawei está permanentemente preocupada por la calidad de los contenidos de este documento; sin embargo, ninguna declaración, información ni recomendación aquí contenida constituye garantía alguna, ni expresa ni implícita. La información contenida en este documento se encuentra sujeta a cambios sin previo aviso.

Huawei Cloud Computing Technologies Co., Ltd.

Dirección: Huawei Cloud Data Center Jiaoxinggong Road
Avenida Qianzhong
Nuevo distrito de Gui'an
Gui Zhou, 550029
República Popular China

Sitio web: <https://www.huaweicloud.com/intl/es-us/>

Índice

1	Introducción a DevEnviron.....	1
2	Escenarios de aplicación.....	5
3	Gestión de instancias de notebook.....	6
3.1	Creación de una instancia de notebook.....	6
3.2	Acceso a una instancia de notebook.....	14
3.3	Búsqueda, inicio, detención o supresión de instancias de notebook.....	15
3.4	Cambio de una imagen de instancia de notebook.....	18
3.5	Cambio de la variante de una instancia de notebook.....	18
3.6	Selección de almacenamiento en DevEnviron.....	19
3.7	Montaje dinámico de un sistema de archivos paralelo de OBS.....	24
3.8	Ampliación dinámica de la capacidad del disco de EVS.....	25
3.9	Modificación de la configuración de SSH para una instancia de notebook.....	27
3.10	Consulta de las instancias de notebook de todos los usuarios de IAM con una cuenta de tenant.....	30
3.11	Consulta de eventos de notebook.....	31
3.12	Informes de alarmas del directorio de caché de notebook.....	36
4	JupyterLab.....	42
4.1	Proceso de operación de JupyterLab.....	42
4.2	Descripción y operaciones comunes de JupyterLab.....	43
4.3	Complemento de parametrización de código.....	51
4.4	Uso del SDK de ModelArts.....	53
4.5	Uso de complemento de Git.....	54
4.6	Entrenamiento de modelo visualizado.....	60
4.6.1	Introducción a la visualización de trabajos de entrenamiento.....	60
4.6.2	Trabajos de visualización de MindInsight.....	60
4.6.3	Trabajos de visualización de TensorBoard.....	66
4.7	Carga y descarga de datos en notebook.....	73
4.7.1	Carga de archivos a JupyterLab.....	73
4.7.1.1	Escenarios.....	73
4.7.1.2	Carga de archivos desde una ruta local a JupyterLab.....	73
4.7.1.2.1	Carga de escenarios y de entradas.....	73
4.7.1.2.2	Carga de un archivo local menos de 100 MB a JupyterLab.....	75
4.7.1.2.3	Carga de un archivo local con un tamaño que varía de 100 MB a 5 GB a JupyterLab.....	76

4.7.1.2.4 Carga de un archivo local más de 5 GB a JupyterLab.....	79
4.7.1.3 Clonación de un repositorio de código abierto de GitHub.....	81
4.7.1.4 Carga de archivos de OBS a JupyterLab.....	84
4.7.1.5 Carga de archivos remotos a JupyterLab.....	87
4.7.2 Descarga de un archivo de JupyterLab a una ruta local.....	89
5 IDE local.....	92
5.1 Proceso de operación en un IDE local.....	92
5.2 IDE local (PyCharm).....	93
5.2.1 Conexión a una instancia de notebook con PyCharm Toolkit.....	93
5.2.1.1 PyCharm Toolkit.....	93
5.2.1.2 Descarga e instalación de PyCharm Toolkit.....	94
5.2.1.3 Conexión a una instancia de notebook con PyCharm Toolkit.....	95
5.2.2 Conexión manual a una instancia de notebook con PyCharm.....	102
5.2.3 Envío de un trabajo de entrenamiento con PyCharm Toolkit.....	108
5.2.3.1 Envío de un trabajo de entrenamiento (Nueva versión).....	108
5.2.3.2 Detención de un trabajo de entrenamiento.....	112
5.2.3.3 Consulta de logs de entrenamiento.....	113
5.2.4 Carga de datos en una instancia de notebook con PyCharm.....	114
5.3 IDE local (VS Code).....	115
5.3.1 Conexión a una instancia de notebook por VS Code.....	115
5.3.2 Instalación de VS Code.....	115
5.3.3 Conexión a una instancia de notebook a través del VS Code Toolkit.....	116
5.3.4 Conexión manual a una instancia de notebook por VS Code.....	122
5.3.5 Depuración remota en VS Code.....	128
5.3.6 Carga y descarga de archivos en VS Code.....	130
5.4 IDE local (acceso con SSH).....	132
6 Referencia de comandos de ModelArts CLI.....	139
6.1 Descripción de ModelArts CLI.....	139
6.2 (Opcional) Instalación de ma-cli localmente.....	140
6.3 Completado automático para comandos de ma-cli.....	141
6.4 Autenticación de ma-cli.....	142
6.5 Comando de creación de ma-cli Image.....	145
6.5.1 Comando de creación de ma-cli Image.....	145
6.5.2 Obtención de una plantilla de creación de imágenes.....	146
6.5.3 Carga de una plantilla de creación de imágenes.....	147
6.5.4 Obtención de imágenes de ModelArts registradas.....	148
6.5.5 Creación de una imagen en notebook de ModelArts.....	150
6.5.6 Obtención de cachés de creación de imágenes en el notebook de ModelArts.....	151
6.5.7 Borrado de cachés de creación de imágenes en el notebook de ModelArts.....	152
6.5.8 Registro de imágenes de SWR con la gestión de imágenes de ModelArts.....	153
6.5.9 Cancelación del registro de una imagen en la gestión de imágenes de ModelArts.....	155
6.5.10 Depuración de una imagen de SWR en un ECS.....	156

6.6 Uso del comando ma-cli ma-job para enviar un trabajo de entrenamiento de ModelArts.....	157
6.6.1 Descripción del comando ma-cli ma-job	157
6.6.2 Obtención de trabajos de entrenamiento de ModelArts.....	158
6.6.3 Envío de un trabajo de entrenamiento de ModelArts.....	160
6.6.4 Obtención de logs de trabajos de entrenamiento de ModelArts.....	165
6.6.5 Obtención de eventos de trabajo de entrenamiento de ModelArts.....	166
6.6.6 Obtención de motores de IA de ModelArts para entrenamiento.....	167
6.6.7 Obtención de especificaciones de recursos de ModelArts para entrenamiento.....	168
6.6.8 Detención de un trabajo de entrenamiento de ModelArts.....	169
6.7 Uso del comando ma-cli dli-job para enviar un trabajo de DLI Spark.....	170
6.7.1 Descripción.....	170
6.7.2 Consulta de trabajos de DLI Spark.....	171
6.7.3 Envío de un trabajo de DLI Spark.....	173
6.7.4 Consulta de logs de ejecución de DLI Spark.....	178
6.7.5 Consulta de colas de DLI.....	179
6.7.6 Obtención de recursos del grupo de DLI.....	181
6.7.7 Carga de archivos locales o archivos de OBS en un grupo de DLI.....	183
6.7.8 Detención de un trabajo de DLI Spark.....	184
6.8 Uso de ma-cli para copiar datos de OBS.....	185

1 Introducción a DevEnviron

NOTA

Este documento describe las funciones del notebook de DevEnviron de la nueva versión.

El desarrollo de software es un proceso de reducir los costos de los desarrolladores y mejorar la experiencia de desarrollo. En el desarrollo de IA, el ModelArts se dedica a mejorar la experiencia de desarrollo de IA y a simplificar el proceso de desarrollo. ModelArts DevEnviron utiliza los recursos nativos en la nube e integra la cadena de herramientas de desarrollo para proporcionar una mejor experiencia de desarrollo de IA en la nube para el desarrollo, la exploración y la enseñanza de IA.

Notebook de ModelArts para una perfecta colaboración en la nube y en las instalaciones

- JupyterLab en la nube, IDE local y complementos de ModelArts para desarrollo y depuración remotos, adaptados a sus necesidades
- Entorno de desarrollo en la nube con recursos informáticos de IA, almacenamiento en la nube y motores de IA integrados
- Entorno de tiempo de ejecución personalizado guardado como una imagen para entrenamiento e inferencia

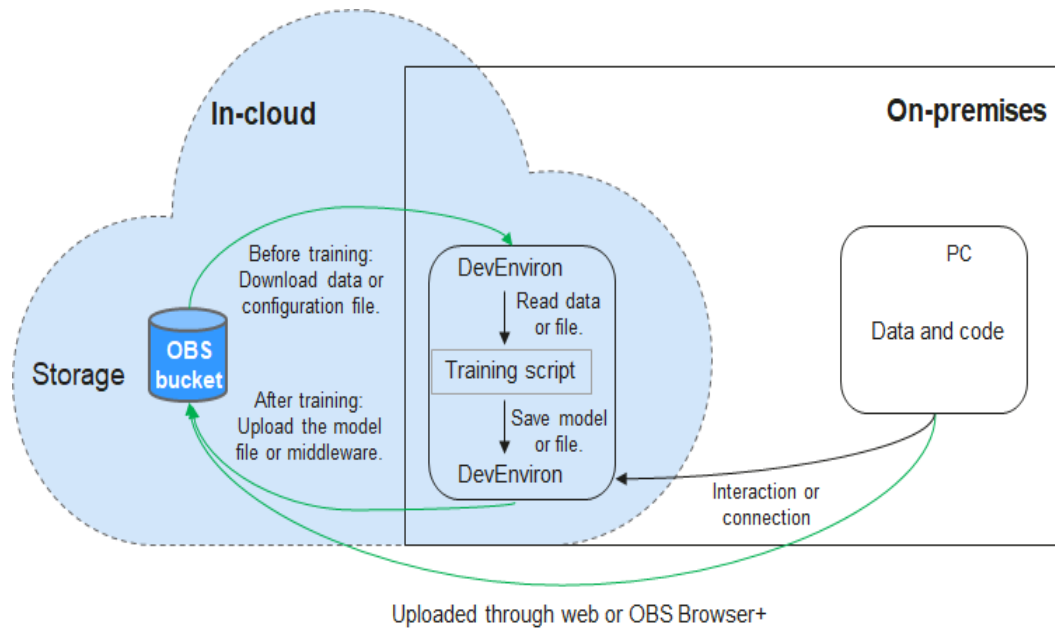
Característica 1: Desarrollo remoto, que permite el acceso remoto al notebook desde un IDE local

El notebook de la nueva versión ofrece el desarrollo remoto. Después de habilitar SSH remoto, puede acceder de forma remota al entorno de desarrollo de notebooks de ModelArts para depurar y ejecutar el código desde un IDE local.

Debido a los recursos locales limitados, los desarrolladores que usan un IDE local ejecutan y depuran código normalmente en un servidor de CPU o GPU compartido entre los miembros del equipo. La construcción y el mantenimiento del servidor de CPU o GPU son costosos.

Las instancias de notebooks de ModelArts están listas para usar con varios motores y variantes integrados para que seleccione. Puede utilizar un entorno de contenedor dedicado. Solo después de configuraciones simples, puede acceder remotamente al entorno para ejecutar y depurar código desde su IDE local.

Figura 1-1 Acceso remoto a notebook desde un IDE local



El notebook de ModelArts puede considerarse como una extensión de un entorno de desarrollo local. Las operaciones como la lectura de datos, el entrenamiento y el almacenamiento de archivos son las mismas que las realizadas en un entorno local.

El notebook de ModelArts le permite usar recursos en la nube sin cambios en los hábitos de codificación locales.

Un IDE local admite Visual Studio (VS) Code, PyCharm y SSH. Además, los complementos de PyCharm Toolkit y de VS Code Toolkit le permiten usar fácilmente recursos en la nube.

Característica 2: guardado de imágenes con un clic para guardar un entorno de desarrollo

El notebook de ModelArts de la nueva versión le permite guardar una instancia de notebook en ejecución como una imagen personalizada con un solo clic.

Cuando se guarda una imagen, se conserva el paquete de dependencia pip instalado. En el desarrollo remoto con VS Code, se conservan las aplicaciones instaladas en el servidor.

Característica 3: imágenes preestablecidas listas para usar con configuraciones optimizadas y compatibles con motores de IA estándar

Los motores de IA y las versiones preestablecidas en cada imagen son fijos. Al crear una instancia de notebook, especifique un motor y una versión de IA, incluido el tipo de chip.

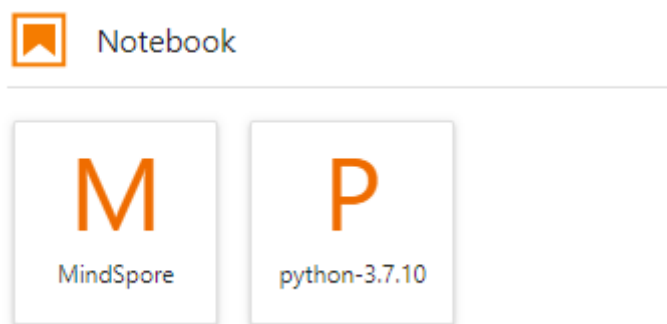
ModelArts DevEnviron proporciona un grupo de imágenes preestablecidas, incluidas las imágenes de PyTorch, de TensorFlow y de MindSpore. Puede utilizar una imagen preestablecida para iniciar la instancia del notebook. Después del desarrollo en la instancia, presenta un trabajo de entrenamiento sin ninguna adaptación.

Las versiones de imagen preestablecidas de ModelArts se determinan en función de los comentarios del usuario y la estabilidad de la versión. Si su desarrollo se puede llevar a cabo utilizando las versiones preestablecidas de ModelArts, por ejemplo, MindSpore 1.5, utilice las

imágenes preestablecidas. Estas imágenes han sido completamente verificadas y tienen muchos paquetes de instalación de uso común incorporados. Están listos para usar, lo que le libera de configurar el entorno.

Las imágenes preestablecidas en el DevEnviron de ModelArts mantiene:

- Paquetes preestablecidos comunes: motores comunes de IA como PyTorch y MindSpore basados en el estándar Conda, paquetes comunes de software de análisis de datos como Pandas y Numpy, y software de herramientas comunes como CUDA y CUDNN, que cumplen con los requisitos comunes de desarrollo de IA.
- Entornos de Conda preestablecidos: Se crean un entorno de Conda y Conda Python básico (excluyendo cualquier motor de IA) para cada imagen preestablecida. La siguiente figura muestra el entorno de Conda para una imagen MindSpore preestablecida.



Seleccione un entorno de Conda en función de si se utiliza el motor AI para la depuración.

- Notebook: una aplicación web que permite codificar en la GUI y combinar el código, las ecuaciones matemáticas y el contenido visualizado en un documento.
- Complementos de JupyterLab: permite cambiar de variante, compartir casos en AI Gallery para la comunicación y detener la instancia para mejorar la experiencia del usuario.
- SSH remoto: le permite depurar remotamente una instancia de notebook desde un PC local.
- Después de las imágenes preestablecidas en el desarrollo de soporte de ModelArts DevEnviron, los trabajos de entrenamiento se pueden ejecutar en ModelArts.

NOTA

- Para simplificar las operaciones, el notebook de ModelArts de la nueva versión admite el cambio entre motores de IA en una instancia de notebook.
- Los motores de IA varían según las regiones. Para obtener más información sobre los motores de IA disponibles en una región, consulte los motores de IA que se muestran en la consola de gestión.

Característica 4: JupyterLab, una herramienta interactiva de desarrollo y depuración en línea

ModelArts integra JupyterLab de código abierto para el desarrollo interactivo en línea y la depuración. Puede utilizar el notebook en la consola de gestión de ModelArts para compilar y depurar código y entrenar modelos basados en el código, sin importar la instalación o configuración del entorno.

JupyterLab es un entorno de desarrollo interactivo. Es el producto de última generación de Jupyter Notebook. JupyterLab le permite compilar notebook, operar terminales, editar texto Markdown, habilitar la interacción y ver archivos e imágenes de CSV.

2 Escenarios de aplicación

ModelArts ofrece entornos de desarrollo flexibles y abiertos. Seleccione un entorno de desarrollo basado en los requisitos del sitio.

- Notebook en la nube, listo para usar, que le libera de preocuparse por la instalación o configuración del entorno. Para obtener más detalles, véase [Descripción y operaciones comunes de JupyterLab](#).
- IDE local para el desarrollo de modelos. Después de habilitar SSH remoto, puede acceder de forma remota al entorno de desarrollo de notebooks de ModelArts para depurar y ejecutar el código desde un IDE local. El IDE local le permite utilizar el entorno de desarrollo de notebook en la nube sin cambios en los hábitos de codificación locales.

Un IDE local admite Visual Studio (VS) Code, PyCharm y SSH. Además, se proporciona PyCharm Toolkit y VS Code Toolkit para un cómodo acceso remoto. Para obtener más detalles, véase y [Conexión a una instancia de notebook a través del VS Code Toolkit](#).

3 Gestión de instancias de notebook

3.1 Creación de una instancia de notebook

Antes de desarrollar un modelo, cree una instancia de notebook y acceda a ella para su codificación.

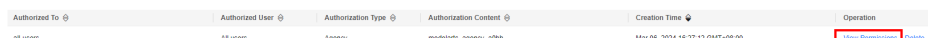
Contexto

- El notebook se factura de la siguiente manera:
 - Una instancia de notebook en ejecución se facturará en función de los recursos utilizados. Las tarifas varían dependiendo de los recursos seleccionados. Para obtener más información, consulte [Detalles del precio](#). Cuando no se utiliza una instancia de notebook, deténgala.
 - Si selecciona EVS para el almacenamiento al crear una instancia de notebook, el disco de EVS se facturará continuamente. Detenga y elimine la instancia del notebook si no es necesario.
- Cuando se crea una instancia de notebook, la detención automática está activada de forma predeterminada. La instancia del notebook se detendrá automáticamente a la hora especificada.
- Solo se puede acceder o detener a las instancias de notebook en ejecución.
- Se pueden crear un máximo de 10 instancias de notebook en una cuenta.

Procedimiento

1. Inicie sesión en la consola de gestión de ModelArts. En el panel de navegación, elija **Settings** y compruebe si se ha configurado la autorización de acceso. Si no es así, configure la autorización de acceso. Para obtener más detalles, véase [Configuración de la autorización de acceso](#).

Figura 3-1 Consulta de configuraciones de agencias



Authorized To	Authorized User	Authorization Type	Authorization Content	Creation Time	Operation
all-users	All users	Agency	modelarts_agency_000b	Mar 05, 2024 15:27:12 GMT+08:00	View Permissions Delete

2. Inicie sesión en la consola de gestión de ModelArts. En el panel de navegación de la izquierda, seleccione **DevEnviron** > **Notebook**.

3. Haga clic en **Create** en la esquina superior derecha. En la página **Create Notebook**, configure los parámetros.
 - a. Configure la información básica de la instancia del notebook, incluidos su nombre, descripción y estado de detención automática. Para obtener más detalles, véase [Tabla 3-1](#).

Figura 3-2 Información básica de una instancia de notebook

The screenshot shows a form with three main sections:

- Name:** A text input field containing 'notebook-88ed'.
- Description:** A larger text input field, currently empty, with a character count '0/256' at the bottom right.
- Auto Stop:** A toggle switch that is currently turned on (blue).

 Below the form is a light blue information banner with a question mark icon and the text: 'Enable this option to specify a time for the notebook instance to automatically stop. You will not be billed after it has stopped.' There is a close button 'X' on the right.

Tabla 3-1 Parámetros básicos

Parámetro	Descripción
Name	Nombre de la instancia de notebook, generada automáticamente por el sistema. Puede cambiarle el nombre en función de los requisitos del servicio. Un nombre tiene un máximo de 128 caracteres y no puede estar vacío. Solo puede contener dígitos, letras, guiones bajos (_), y guiones (-).
Descripción	Breve descripción de la instancia del notebook
Auto Stop	<p>Detiene automáticamente la instancia del notebook a una hora especificada. Esta función está habilitada por defecto. El valor predeterminado es 1 hour, que indica que la instancia del notebook se detiene automáticamente después de ejecutarse durante 1 hora y su facturación de recursos luego se detendrá. Las opciones son 1 hour, 2 hours, 4 hours, 6 hours y Custom. Puede seleccionar Custom para especificar cualquier entero de 1 a 24 horas.</p> <ul style="list-style-type: none"> ● Stop as scheduled: si esta opción está activada, la instancia de notebook se detiene automáticamente cuando la duración de ejecución supera la duración especificada. <p>NOTA Para proteger los trabajos en curso, una instancia de notebook no se detiene automáticamente inmediatamente en el momento de la parada automática. En su lugar, se proporciona un período de 2 a 5 minutos para que renueve el tiempo de parada automática.</p>

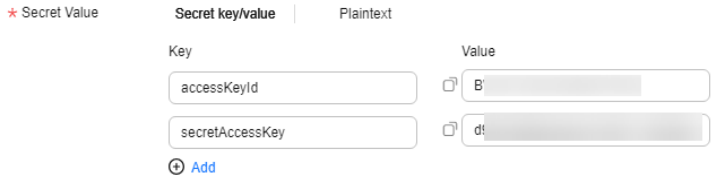
- b. Configure los parámetros del notebook, como la variante de la imagen y la instancia. Para obtener más detalles, véase [Tabla 3-2](#).

Tabla 3-2 Parámetros de instancia de notebook

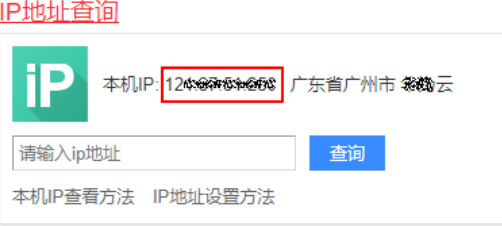
Parámetro	Descripción
Image	<p>Se admiten las imágenes públicas y privadas.</p> <ul style="list-style-type: none"> ● Las imágenes públicas son los motores de IA construidos en el ModelArts. ● Las imágenes privadas se pueden crear usando una instancia que se crea usando una imagen pública. Para obtener más información, véase Uso de imágenes personalizadas en instancias de notebook. <p>Una imagen corresponde a un motor de AI. Cuando selecciona una imagen durante la creación de una instancia, el motor de AI se especifica en consecuencia. Seleccione una imagen según sea necesaria. Introduzca una palabra clave del nombre de la imagen en el cuadro de búsqueda de la derecha para buscar rápidamente la imagen.</p> <p>Puede cambiar una imagen en una instancia de notebook detenida.</p>
Resource Type	<p>Los grupos de recursos públicos y dedicados están disponibles para que seleccione.</p> <p>Los grupos de recursos públicos se facturan en función de la duración de ejecución de las instancias de notebook.</p> <p>Seleccione un grupo de recursos dedicado creado en función de los requisitos del sitio. Si no hay los recursos dedicados disponibles, compre uno.</p>
Type	<p>Tipo de procesador, que puede ser CPU o GPU.</p> <p>Los chips varían dependiendo de la imagen seleccionada.</p> <p>Las GPU ofrecen un mejor rendimiento que las CPU, pero a un costo mayor. Seleccione un tipo de chip según sea necesario.</p>

Parámetro	Descripción
Flavor	<p>La variante de la instancia de notebook. Seleccione una variante en función de sus necesidades.</p> <ul style="list-style-type: none">● CPU 2vCPUs 8GB: Variante de CPU Intel de uso general, ideal para la exploración rápida de datos y experimentos 8vCPUs 32GB: Variante de CPU Intel de cómputo-plus general, ideal para aplicaciones de cómputo intensivo● GPU GPU: 1*Vnt1(32GB) CPU: 8vCPUs 64GB: GPU única con 32 GB de memoria, ideal para entrenamiento de algoritmos y depuración en escenarios de aprendizaje profundo GPU: 1*Tnt004(16GB) CPU: 8vCPUs* 32GB: GPU única con 16 GB de memoria, ideal para cómputo de inferencia como visión de computadora, procesamiento de video y tareas de NLP GPU: 1*Pnt1(16GB) CPU: 8vCPUs 64GB: GPU única con 16 GB de memoria, ideal para entrenamiento de algoritmos y depuración en escenarios de aprendizaje profundo

Parámetro	Descripción
Storage	<p>El valor puede ser EVS, SFS, OBS o PFS. Configure este parámetro según sus necesidades.</p> <p>NOTA OBS y PFS son funciones en la lista blanca. Si tiene requisitos de prueba, envíe un ticket de servicio para solicitar permisos.</p> <ul style="list-style-type: none"> ● EVS Establecer un tamaño de disco basado en los requisitos del servicio. El valor predeterminado es 5 GB. El tamaño máximo del disco se muestra en la GUI. El espacio en disco de EVS se carga en GB desde el momento en que se crea la instancia del notebook hasta el momento en que se elimina la instancia del notebook. ● SFS Seleccione este tipo solo para un grupo de recursos dedicado. SFS solo se aplica después de que un grupo de recursos dedicados pueda comunicarse con su VPC. Para obtener más información, véase Red de ModelArts. NOTA Para obtener detalles sobre cómo definir permisos para acceder a las carpetas de SFS Turbo, véase Gestión de permisos. <ul style="list-style-type: none"> – Scalable File Service: seleccione un sistema de archivos de SFS Turbo creado. – Cloud Mount Path: conserva el valor predeterminado /home/ma-user/work/. – Mounted Subdirectory: seleccione la ruta de almacenamiento en SFS Turbo. – Mount Method: este parámetro se muestra cuando se concede el permiso de control de carpetas al usuario. El permiso read/write o read-only se muestra según la ruta de almacenamiento en SFS Turbo. ● El valor puede ser OBS o PFS. Storage Path: configure la ruta de OBS para almacenar datos de notebook. Si desea utilizar archivos o datos existentes, cárguelos en la ruta de OBS especificada. Storage Path debe configurarse en un directorio específico en un bucket de OBS en lugar de en el directorio raíz del bucket de OBS. Secret: seleccione un secreto existente o haga clic en Create a la derecha para crear uno. En la consola de DEW que se muestra, crea un secreto. Ingrese accessKeyId y secretAccessKey en Key e ingrese las AK/SK obtenidas de My Credentials > Access Keys en Value.

Parámetro	Descripción
	<p>Figura 3-3 Configuración de los valores de secreto</p>  <p>Todas las rutas de almacenamiento de EVS y SFS se montan en el directorio /home/ma-user/work. Todas las operaciones de lectura y escritura en los archivos de la instancia del notebook se almacenan en este directorio, no en OBS.</p> <p>Puede agregar una ruta de almacenamiento de datos durante el tiempo de ejecución de una instancia de notebook consultando Montaje dinámico de un sistema de archivos paralelo de OBS.</p> <p>Los datos se conservan en /home/ma-user/work aunque la instancia del notebook se detenga o se reinicie.</p> <p>Cuando se elimina una instancia de notebook, se lanza el almacenamiento de EVS y no se conservan los datos almacenados. SFS se puede montar en una nueva instancia de notebook y se pueden conservar los datos.</p>
Extended Storage	<p>NOTA</p> <p>Este parámetro es una función en la lista blanca. Si tiene requisitos de prueba, envíe un ticket de servicio para solicitar permisos.</p> <p>Si necesita varias rutas de almacenamiento de datos, haga clic en Add Extended Storage para agregar más directorios de montaje de almacenamiento. Puede agregar un directorio de OBS, de PFS o de SFS.</p> <p>Restricciones:</p> <ul style="list-style-type: none"> ● Para cada tipo, se pueden montar un máximo de cinco directorios. ● Los directorios deben ser únicos y no se pueden montar en un directorio en la lista negra. Se permite el montaje anidado. Los directorios incluidos en la lista negra son aquellos con los siguientes prefijos: /data/, /cache/, /dev/, /etc/, /bin/, /lib/, /sbin/, /modelarts/, /train-worker1-log/, /var/, /resource_info/, /usr/, /sys/, /run/, /tmp/, /infer/ y /opt/ <p>Una vez configurado este parámetro, se muestra la página de detalles de la instancia de notebook. Haga clic en Storage > Extended Storage para ver o editar la información de almacenamiento ampliado. Si la cantidad de dispositivos de almacenamiento no alcanza el máximo, puede hacer clic en Add Extended Storage a la derecha.</p>

Parámetro	Descripción
Remote SSH	<ul style="list-style-type: none">● Después de activar esta función, puede acceder de forma remota al entorno de desarrollo de la instancia de notebook desde su entorno de desarrollo local.● Cuando se detiene una instancia de notebook, puede actualizar la configuración de SSH en la página de detalles de la instancia. <p>NOTA</p> <p>Las instancias de notebook con SSH remoto habilitado tienen complementos de VS Code (como Python y Jupyter) y el paquete de servidor de VS Code preinstalado, que ocupan aproximadamente 1 GB de espacio de almacenamiento persistente.</p>
Key Pair	<p>Establezca un par de claves después de que el SSH remoto esté habilitado.</p> <p>Seleccione un par de claves existente.</p> <p>También puede hacer clic en Create a la derecha del cuadro de texto para crear uno en la consola de DEW. Para ello, seleccione Key Pair Service > Private Key Pairs y haga clic en Create Key Pair.</p> <p>Después de crear una instancia de notebook, puede cambiar el par de claves en la página de detalles de la instancia.</p> <p>ATENCIÓN</p> <p>Descargue el par de claves creado y guárdelo correctamente. Cuando utiliza un IDE local para acceder de forma remota al entorno de desarrollo del notebook, se requiere el par de claves para la autenticación.</p>

Parámetro	Descripción
Whitelist	<p>Configure una lista blanca después de habilitar SSH remoto. Este parámetro es opcional.</p> <p>Agregue las direcciones IP para acceder de forma remota a la instancia de notebook a la lista blanca, por ejemplo, la dirección IP de su PC local o la dirección IP pública del dispositivo de origen. Un máximo de cinco direcciones IP pueden ser agregadas y separadas por comas (.). Si el parámetro se deja en blanco, se permitirán todas las direcciones IP para el acceso de SSH remoto.</p> <p>Si su dispositivo de origen y ModelArts están aislados entre sí en la red, obtenga la dirección IP pública de su dispositivo de origen mediante un motor de búsqueda convencional, por ejemplo, ingresando "IP address lookup", pero no ejecutando ipconfig ni ifconfig/ip localmente.</p> <p>Figura 3-4 IP address lookup</p>  <p>Después de crear una instancia de notebook, puede cambiar las direcciones IP en la lista blanca en la página de detalles de la instancia.</p>

- c. (Opcional) Agregue etiquetas a la instancia de notebook. Ingrese una clave y un valor de etiqueta y haga clic en **Add**.

Tabla 3-3 Adición de una etiqueta

Parámetro	Descripción
Tags	<p>ModelArts puede trabajar con Tag Management Service (TMS). Al crear tareas que consumen recursos de ModelArts, por ejemplo, trabajos de entrenamiento, configure etiquetas para estas tareas de modo que ModelArts pueda usar etiquetas para gestionar recursos por grupo.</p> <p>Para obtener más información sobre cómo utilizar etiquetas, véase ¿Cómo utiliza ModelArts las etiquetas para gestionar recursos por grupo?</p> <p>Después de agregar una etiqueta, puede verla, modificarla o eliminarla en la página de detalles de la instancia de notebook.</p>

 **NOTA**

Puede seleccionar una etiqueta de TMS predefinida de la lista desplegable de etiquetas o personalizar una etiqueta. Las etiquetas predefinidas están disponibles para todos los recursos de servicio que admiten etiquetas. Las etiquetas personalizadas solo están disponibles para los recursos de servicio del usuario que las ha creado.

4. Haga clic en **Next**.
5. Después de confirmar la configuración de los parámetros, haga clic en **Submit**.
Cambie a la lista de instancias de notebook. Se está creando la instancia de notebook. El cambio de estado a **Running** tardará varios minutos. A continuación, se crea la instancia del notebook.
6. En la lista de instancias de notebook, haga clic en el nombre de instancia. En la página de detalles de la instancia que se muestra, vea la configuración de la instancia.

Si **Remote SSH** está activado, puede hacer clic en el ícono de modificación situado a la derecha de la lista blanca para modificarlo. Puede hacer clic en el ícono de modificación situado a la derecha de **Authentication** para actualizar el par de claves de una instancia de notebook detenida.

En la ficha **Storage**, haga clic en **Mount Storage** para montar un sistema de archivos paralelo de OBS en la instancia para leer datos. Para obtener más detalles, véase [Montaje dinámico de un sistema de archivos paralelo de OBS](#).

Si se utiliza un disco de EVS, haga clic en **Expansion** a la derecha de **Storage Capacity** para ampliar dinámicamente la capacidad del disco de EVS. Para obtener más detalles, véase [Ampliación dinámica de la capacidad del disco de EVS](#).

3.2 Acceso a una instancia de notebook

Acceda a una instancia de notebook en el estado **Running** para la codificación.

Los métodos para acceder a las instancias de notebook varían según el motor de IA basado en el que se creó la instancia.

- Acceso remoto: utilice PyCharm o VS Code o SSH en el IDE local. Para obtener más detalles, véase [Conexión a una instancia de notebook a través del VS Code Toolkit](#) y .
- Acceso en línea: utilice JupyterLab. Para obtener más detalles, véase [Descripción y operaciones comunes de JupyterLab](#).

Cree una instancia y monte el almacenamiento persistente en **/home/ma-user/work**.

```
sh-4.4$pwd
/home/ma-user
sh-4.4$cd work/
sh-4.4$pwd
/home/ma-user/work
sh-4.4$
```

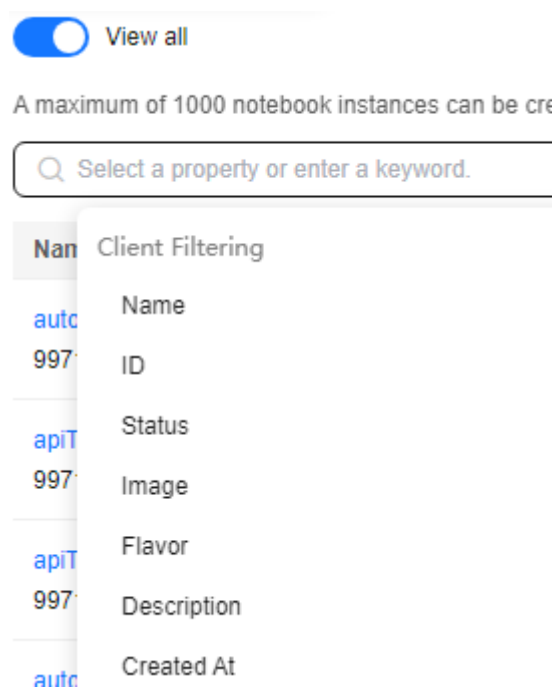
Los datos almacenados solo en el directorio **work** se conservan después de detener o reiniciar la instancia. Cuando utilice un entorno de desarrollo, almacene los datos de persistencia en **/home/ma-user/work**.

3.3 Búsqueda, inicio, detención o supresión de instancias de notebook

Búsqueda de una instancia

Todas las instancias creadas se muestran en la página del notebook. Para mostrar una instancia específica, búsquela en función de los criterios de filtro. Haga clic en el cuadro de búsqueda y seleccione uno o varios criterios de búsqueda.

Figura 3-5 Búsqueda de una instancia

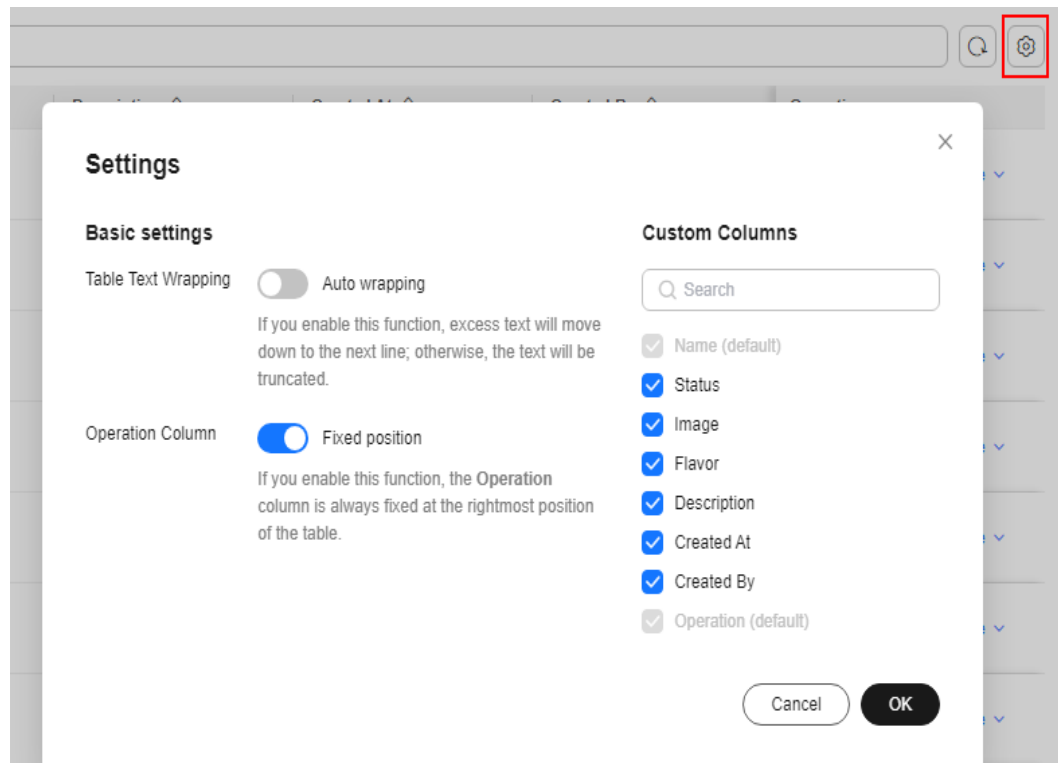


- Permita que **View all** compruebe todas las instancias de notebook creadas por todos los subusuarios del proyecto de IAM.
- Seleccione criterios de búsqueda, como nombre, ID, estado, imagen, variante, descripción y tiempo de creación.

Personalización de columnas de tabla

Haga clic en el botón de configuración para personalizar las columnas que se mostrarán en la tabla.

Figura 3-6 Ajustes



Inicio o detención de una instancia

Detenga las instancias de notebook que no son necesarias. También puede reiniciar una instancia detenida.

1. Inicie sesión en la consola de gestión de ModelArts. En el panel de navegación de la izquierda, seleccione **DevEnviron** > **Notebook**.
2. Inicie o detenga la instancia del notebook de destino.
 - Para iniciar una instancia de notebook, haga clic en **Start** en la columna **Operation** de la instancia de notebook de destino. Solo se pueden iniciar instancias de notebook detenidas.
 - Para detener una instancia de notebook, haga clic en **Stop** en la columna **Operation** de la instancia de notebook de destino. Solo se pueden detener las instancias de notebook en ejecución.

ATENCIÓN

Después de detener una instancia de notebook:

- Los datos almacenados solo en **/home/ma-user/work** se conservan. Por ejemplo, se eliminarán los paquetes de dependencias externos instalados en otros directorios del entorno de desarrollo.
- La instancia del notebook ya no se facturará. Sin embargo, si la instancia está conectada con un disco de EVS, el espacio de almacenamiento todavía se facturará.

Eliminación de una instancia

Elimine las instancias de notebook que no sean necesarias.

1. Inicie sesión en la consola de gestión de ModelArts. En el panel de navegación de la izquierda, seleccione **DevEnviron > Notebook**.
2. En la lista de notebook, localice la instancia de notebook de destino y haga clic en **Delete** en la columna **Operation**. En el cuadro de diálogo que aparece en pantalla, confirme la información, escriba **DELETE** en el cuadro de texto y haga clic en **OK**.

ATENCIÓN

Las instancias de notebook eliminadas no se pueden recuperar. Después de eliminar una instancia de notebook, se eliminarán los datos almacenados en el directorio montado.

3.4 Cambio de una imagen de instancia de notebook

ModelArts le permite cambiar imágenes en una instancia de notebook para ajustar de forma flexible su motor de IA.

Restricciones

Se detiene la instancia del notebook de destino.

Procedimiento

1. Inicie sesión en la consola de gestión de ModelArts. En el panel de navegación izquierdo, seleccione **DevEnviron > Notebook**.
2. En la lista de blocs de notas, haga clic en **More** en la columna **Operation** de la instancia de notebook de destino y seleccione **Change Image**.
3. En el cuadro de diálogo **Change Image**, seleccione una imagen nueva y haga clic en **OK**. Después de la modificación, puede ver la nueva imagen en la página de lista del notebook.

3.5 Cambio de la variante de una instancia de notebook

ModelArts permite cambiar la variante del nodo para una instancia de notebook.

Restricciones

Las especificaciones de una instancia de notebook solo se pueden modificar cuando se encuentra en estado **Stopped**, **Running** o **Startup failed**.

Procedimiento

1. Inicie sesión en la consola de gestión de ModelArts. En el panel de navegación de la izquierda, seleccione **DevEnviron > Notebook**.
2. En la lista de instancias de notebook, localice la fila que contiene la instancia de notebook de destino y seleccione **More > Modify Specifications** en la columna

Operation. En el cuadro de diálogo **Modify Specifications** que aparece, seleccione la variante deseada.

Figura 3-7 Modificar las especificaciones

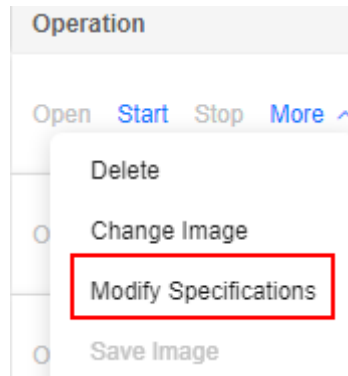
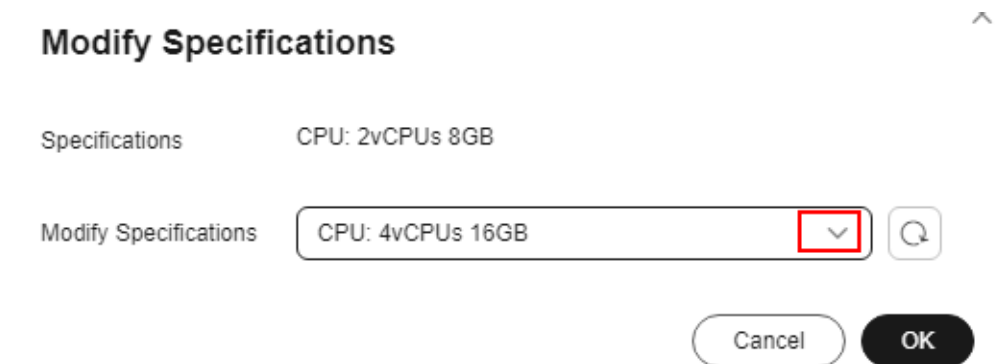


Figura 3-8 Selección de una variante



3.6 Selección de almacenamiento en DevEnviron

El almacenamiento varía según el rendimiento, la facilidad de uso y el costo. Ningún medio de almacenamiento puede cubrir todos los escenarios. Aprender sobre escenarios de aplicaciones de almacenamiento en la nube para un mejor uso.

NOTA

Solo se pueden montar sistemas de archivos paralelos (PFS) de OBS y almacenamiento de objetos en la misma región.

Tabla 3-4 Escenarios de aplicaciones de almacenamiento en la nube

Almacena miento	Escenario de aplicación	Ventaja	Desventajas
EVS	Exploración de datos y algoritmos solo en el entorno de desarrollo.	<p>Las SSD de almacenamiento en bloque ofrecen un mejor rendimiento general de E/S que NFS. La capacidad de almacenamiento se puede ampliar dinámicamente hasta 4096 GB.</p> <p>Como almacenamiento persistente, los discos de EVS se montan en /home/ma-user/work. Los datos de este directorio se conservan después de detener la instancia. La capacidad de almacenamiento se puede ampliar en línea según la demanda.</p>	Este tipo de almacenamiento solo se puede utilizar en un único entorno de desarrollo.

Almacenamiento	Escenario de aplicación	Ventaja	Desventajas
PFS	<p>NOTA PFS es una función en la lista blanca. Para utilizar esta función, póngase en contacto con el soporte técnico de Huawei.</p> <p>Buckets de PFS montados como almacenamiento persistente para el desarrollo y la exploración de IA.</p> <p>- Almacenamiento para conjuntos de datos. Los conjuntos de datos se montan directamente en los notebooks para navegar y procesar datos y se pueden utilizar directamente durante el entrenamiento. Para obtener más detalles, véase ¿Cómo puedo cargar datos en OBS?</p> <p>Una vez ejecutada la instancia, el sistema de archivos paralelo de OBS que transporta los conjuntos de datos se monta dinámicamente en los notebook. Para obtener más detalles, véase Montaje dinámico de un sistema de archivos paralelo de OBS.</p> <p>2. Almacenamiento para código. Después de la depuración en una instancia de notebook, especifique la ruta de OBS como la ruta de código para iniciar el entrenamiento, facilitando la modificación temporal.</p> <p>- Almacenamiento para el entrenamiento de</p>	<p>PFS es un sistema de archivos de almacenamiento de objetos de alto rendimiento optimizado con bajos costos de almacenamiento y gran rendimiento. Puede procesar rápidamente cargas de trabajo de cómputo de alto rendimiento (HPC). Se recomienda el montaje en PFS si se utiliza OBS.</p> <p>NOTA Empaquete o divida los datos que se van a cargar en 128 MB o 64 MB. Descargue y descomprima los datos en el almacenamiento local para obtener un mejor E/S y rendimiento.</p>	<p>Debido al rendimiento promedio en la lectura y escritura frecuente de archivos pequeños, el almacenamiento de PFS no es adecuado para el entrenamiento de modelos grandes o la descompresión de archivos.</p> <p>NOTA Antes de montar el almacenamiento de PFS en una instancia de notebook, otorgue a ModelArts permisos completos de lectura y escritura en el bucket de PFS. La política se conservará incluso después de eliminar la instancia del notebook.</p>

Almacenamiento	Escenario de aplicación	Ventaja	Desventajas
	<p>comprobación. Monte el almacenamiento en la ruta de salida de entrenamiento, como la ruta a los logs de entrenamiento. De esta manera, vea y compruebe el entrenamiento en la instancia del notebook en tiempo real. Esto es especialmente adecuado para analizar la salida de trabajos entrenados con TensorBoard o notebook.</p>		
OBS	<p>NOTA OBS es una función en la lista blanca. Para utilizar esta función, póngase en contacto con el soporte técnico de Huawei.</p> <p>Al cargar o descargar una gran cantidad de datos en el entorno de desarrollo, puede utilizar bucket de OBS para transferir datos.</p>	<p>Bajo coste de almacenamiento y alto throughput, pero rendimiento medio en la lectura y escritura de archivos pequeños. Es una buena práctica empaquetar o dividir el archivo en 128 MB o 64 MB. De este modo, puede descargar los paquetes, descomprimirlos y utilizarlos localmente.</p>	<p>La semántica del almacenamiento de objetos es diferente de la semántica de Posix y debe comprenderse mejor.</p>

Almacenamiento	Escenario de aplicación	Ventaja	Desventajas
SFS	<p>Disponible solo en los grupos de recursos dedicados. Utilice el almacenamiento de SFS en escenarios de producción informal, como exploración y experimentos. Se puede montar un dispositivo de SFS tanto en un entorno de desarrollo como en un entorno de entrenamiento. De esta manera, no es necesario descargar datos cada vez que comienza su trabajo de entrenamiento. Este tipo de almacenamiento no es adecuado para los entrenamientos intensivos de E/S en más de 32 tarjetas.</p>	<p>SFS se implementa como NFS y se puede compartir entre múltiples entornos de desarrollo y entre entornos de desarrollo y entrenamiento. Este tipo de almacenamiento se prefiere para trabajos de entrenamiento distribuidos no pesados, especialmente para aquellos que no requieren descargar datos adicionalmente cuando comienzan los trabajos de entrenamiento.</p>	<p>El rendimiento del almacenamiento de SFS no es tan bueno como el del almacenamiento de EVS.</p>
Local storage	<p>La opción primera para los trabajos de entrenamiento intensivos.</p>	<p>SSD de alto rendimiento para la VM o BMS de destino, con un alto rendimiento de E/S de archivos. Para trabajos de entrenamiento de trabajo pesado, almacene datos en el directorio de destino y, a continuación, inicie el entrenamiento.</p> <p>De forma predeterminada, el almacenamiento se monta en el directorio /cache. Para obtener detalles sobre el espacio disponible del directorio /cache, véase ¿Cuáles son los tamaños de los directorios / cache para diferentes especificaciones de notebook en DevEnviron?</p>	<p>El ciclo de vida del almacenamiento está asociado con el ciclo de vida del contenedor. Los datos deben descargarse cada vez que comienza el trabajo de entrenamiento.</p>

Uso del almacenamiento

1. ¿Cómo uso EVS en un entorno de desarrollo?

Al crear una instancia de notebook, seleccione un disco de EVS de pequeña capacidad. You can scale out the disk as needed. For details, see [Ampliación dinámica de la capacidad del disco de EVS](#).

2. ¿Cómo uso un sistema de archivos paralelo de OBS en un entorno de desarrollo?

Al entrenar datos en una instancia de notebook, puede utilizar los conjuntos de datos montados en un contenedor de notebook y un sistema de archivos paralelo de OBS. Para obtener más detalles, véase [Montaje dinámico de un sistema de archivos paralelo de OBS](#).

3.7 Montaje dinámico de un sistema de archivos paralelo de OBS

Descripción

Parallel File System (Sistema de archivos paralelo, PFS) es un sistema de archivos optimizado de alto rendimiento proporcionado por Object Storage Service (OBS). Para obtener más detalles, véase [Acerca de PFS](#).

El montaje dinámico de OBS utiliza una herramienta de montaje para convertir el protocolo de almacenamiento de objetos en el protocolo de archivo POSIX. El almacenamiento de OBS se simula como un sistema de archivos local y se monta dinámicamente en un contenedor de notebook en ejecución en ModelArts. Después del montaje, puede realizar operaciones de aplicación en los objetos de OBS del contenedor del notebook.

Escenarios de aplicación

Escenario 1: Después de montar el almacenamiento de OBS en el que se almacena el conjunto de datos de destino en la instancia de notebook, puede obtener una vista previa y realizar operaciones en el conjunto de datos, como operar un sistema de archivos local.

Escenario 2: Al entrenar datos en una instancia de notebook, puede utilizar el conjunto de datos montado en un contenedor de notebook.

Restricciones

OBS proporciona bucket de objetos y PFS para el almacenamiento.

El notebook de ModelArts de la nueva versión solo admite el montaje de un sistema de archivos paralelo de OBS a **/data/** de un contenedor de notebook.

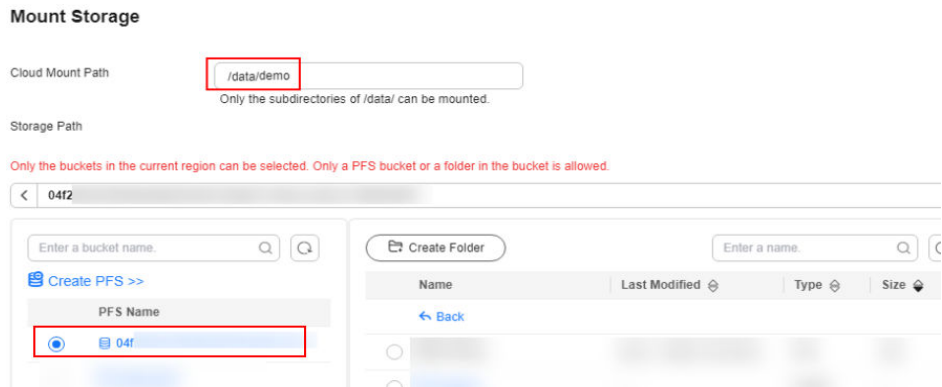
Procedimiento

Método 1: Con la consola de gestión de ModelArts

1. Inicie sesión en la consola de gestión de ModelArts. En el panel de navegación izquierdo, seleccione **DevEnviron** > **Notebook**.
2. Seleccione una instancia de notebook en ejecución y haga clic en su nombre. En la página de detalles de la instancia de notebook, haga clic en la ficha **Storage**. Desde allí, haga clic en **Mount Storage** y configure los parámetros de montaje.
 - a. Establezca un directorio de montaje local. Escriba un nombre de carpeta en **/data/**, por ejemplo, **demo**. El sistema creará automáticamente la carpeta en **/data/** del contenedor del notebook para montar el sistema de archivos de OBS.

- b. Seleccione la carpeta para almacenar el sistema de archivos paralelos de OBS y haga clic en **OK**.

Figura 3-9 Montaje dinámico de un sistema de archivos paralelo de OBS



- 3. Vea el resultado del montaje en la página de detalles de la instancia del notebook.

Figura 3-10 Montaje exitoso

Type	Status	Storage Path	Cloud Mount Path	Operation
Parallel File System	Mounted	obs://[...]	/data/demo/	Unmount Storage

3.8 Ampliación dinámica de la capacidad del disco de EVS

Descripción

Si una instancia de notebook utiliza una disco de EVS para almacenamiento, el disco se monta en **/home/ma-user/work/** del contenedor de notebook y la capacidad del disco se puede ampliar hasta 100 GB cuando la instancia se está ejecutando.

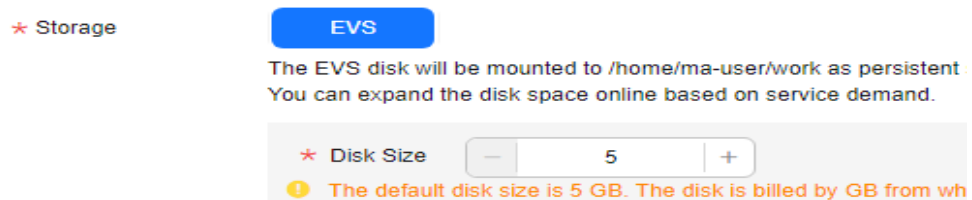
Escenarios de aplicación

Durante el desarrollo de un notebook, seleccione una pequeña capacidad de disco de EVS, por ejemplo, 5 GB, al crear una instancia de notebook, ya que los requisitos de almacenamiento son bajos en la etapa inicial. Después del desarrollo, se debe entrenar un gran volumen de datos. A continuación, amplíe la capacidad del disco para satisfacer de manera rentable sus necesidades de servicio.

Restricciones

- La instancia de notebook de destino debe usar EVS para el almacenamiento.

Figura 3-11 Selección de EVS al crear una instancia de notebook

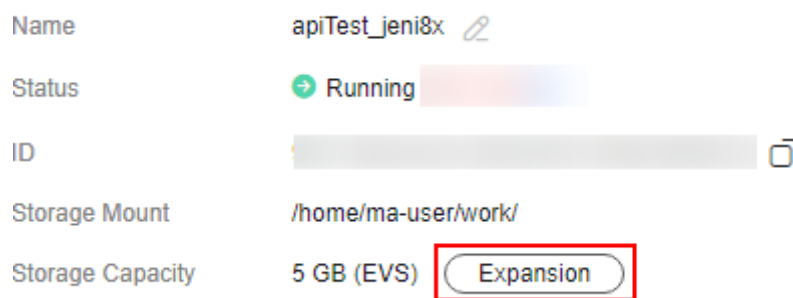


- Se pueden ampliar hasta 100 GB a la vez. Además, la capacidad total después de la expansión no puede exceder los 4096 GB.
- Si la capacidad original de un disco de EVS es 4096 GB, no se puede ampliar la capacidad del disco.
- Después de detener la instancia, la capacidad expandida todavía tiene efecto. La facturación se basa en la capacidad ampliada del disco de EVS.
- Un disco de EVS se factura mientras se utiliza. Para detener la facturación de un disco de EVS, elimine sus datos y láncelo.

Procedimiento

1. Inicie sesión en la consola de gestión de ModelArts. En el panel de navegación izquierdo, seleccione **DevEnviron** > **Notebook**.
2. Haga clic en el nombre de una instancia de notebook en ejecución. En la página de detalles de la instancia, haga clic en **Expansion**.

Figura 3-12 Página de detalles de la Instancia



3. Establezca la capacidad que se va a ampliar y haga clic en **OK**. **Expanding** muestra que la expansión de la capacidad está en curso. Después de la expansión, la capacidad de almacenamiento mostrada es la capacidad ampliada.

Figura 3-13 Expansión de capacidad

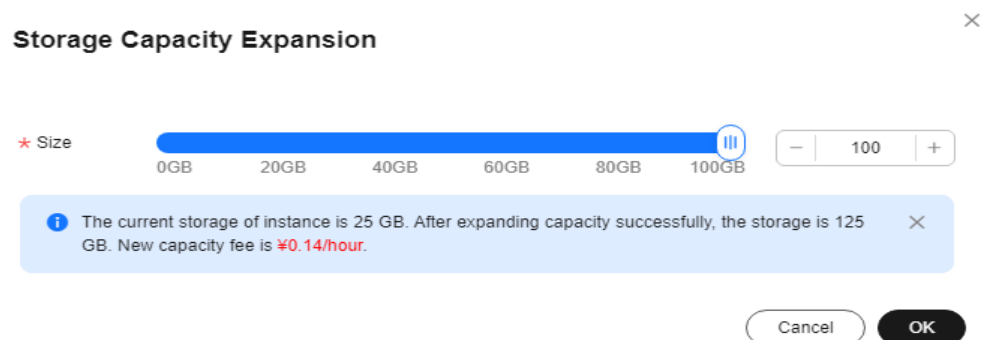



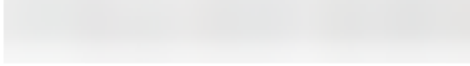


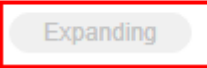


Figura 3-14 Expansión

Name	apiTest 
Status	 Running 
ID	 
Storage Mount	/home/ma-user/work/
Storage Capacity	 5 GB (EVS) 

3.9 Modificación de la configuración de SSH para una instancia de notebook

ModelArts le permite modificar la configuración de SSH para las instancias de notebook.

Si se crea una instancia de notebook con SSH remoto deshabilitado, puede habilitar SSH remoto en la página de detalles del notebook.

Durante la creación de una instancia de notebook, si establece una lista blanca para acceder a ella de forma remota, puede cambiar las direcciones IP de la lista blanca en la página de detalles de la instancia de notebook. También puede cambiar el par de claves.

Restricciones

Se debe detener la instancia del notebook de destino.

Cambio del par de claves y la dirección IP de la conexión remota

1. Inicie sesión en la consola de gestión de ModelArts. En el panel de navegación de la izquierda, seleccione **DevEnviron** > **Notebook**.
2. Haga clic en la instancia del notebook de destino. Habilite SSH remoto y cambie el par de claves y la lista blanca.

NOTA

Para ver el SSH remoto habilitado manualmente, consulte [Figura 3-15](#). Después de actualizar la configuración de SSH, la función de SSH remoto no se puede deshabilitar.

Para SSH remoto habilitado de forma predeterminada en la imagen seleccionada, consulte [Figura 3-16](#).

Figura 3-15 Actualizar la configuración de SSH

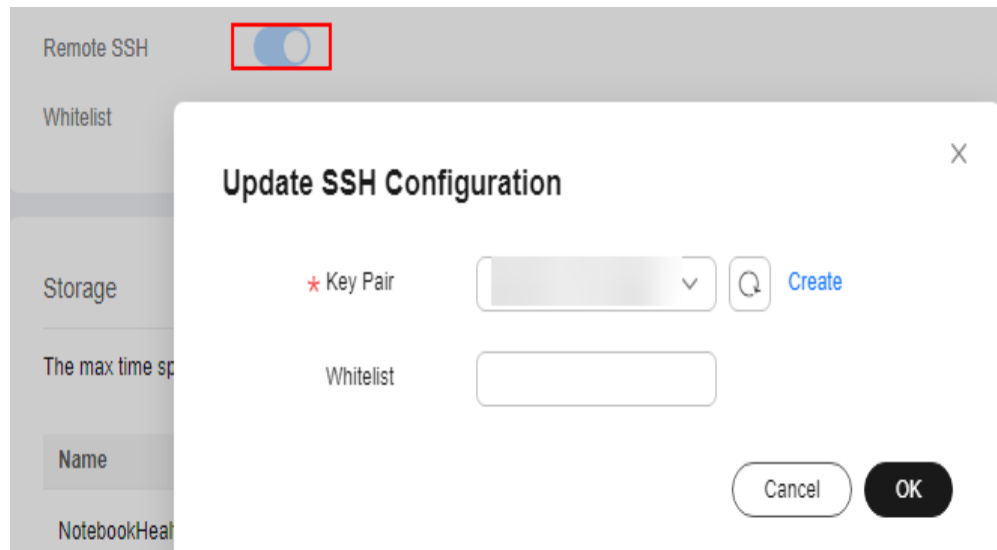
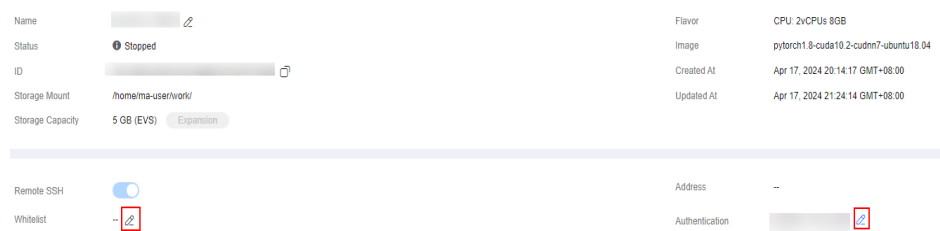



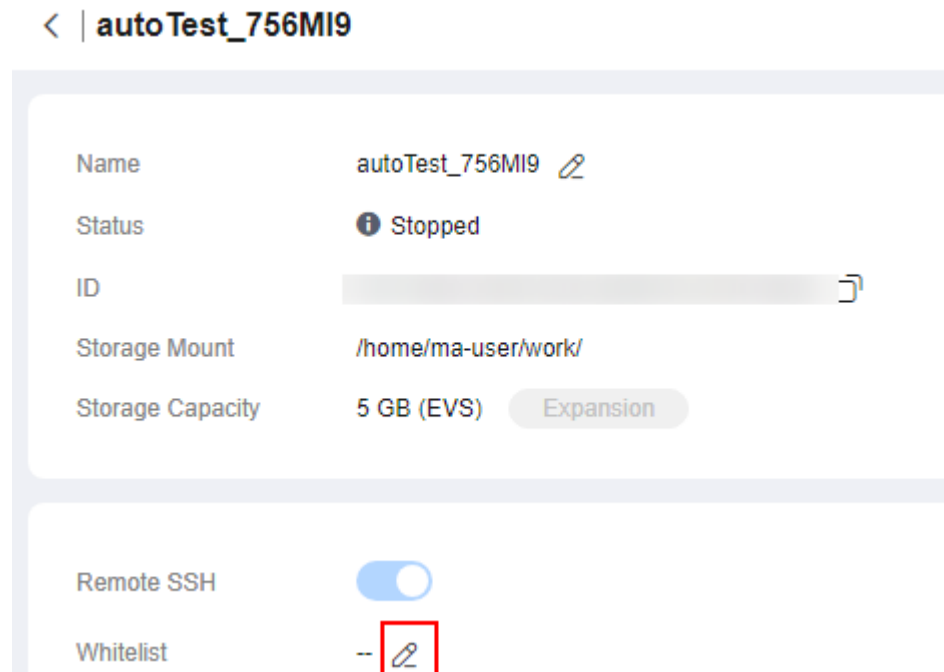
Figura 3-16 Cambio de la lista blanca y el par de claves



- Haga clic en  y elija un par de claves existente o haga clic en **Create** para crear un nuevo par de claves.
- Para obtener detalles sobre cómo configurar una lista blanca, véase [Configuración de una dirección IP para acceder de forma remota a una instancia de notebook](#). Después de cambiar las direcciones IP, los enlaces existentes siguen siendo válidos. Después de liberar los enlaces, los nuevos enlaces solo se pueden configurar desde las direcciones IP cambiadas.

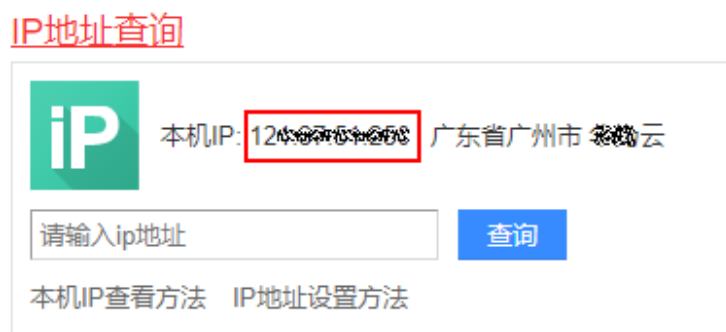
Configuración de una dirección IP para acceder de forma remota a una instancia de notebook

Figura 3-17 Configuración de una dirección IP para acceder de forma remota a una instancia de notebook



Asegúrese de que las direcciones IP públicas están configuradas. Si su dispositivo de origen y ModelArts de Huawei Cloud están aislados entre sí en la red, obtenga la dirección IP pública de su dispositivo de origen mediante un motor de búsqueda convencional, por ejemplo, ingresando "búsqueda de direcciones IP", pero no ejecutando **ipconfig** o **ifconfig/ip** localmente.

Figura 3-18 IP address lookup



3.10 Consulta de las instancias de notebook de todos los usuarios de IAM con una cuenta de tenant

Cualquier usuario de IAM que tenga los permisos **listAllNotebooks** y **listUsers** puede hacer clic en **View all** en la página de notebook para ver las instancias de todos los usuarios del proyecto de IAM actual.

📖 NOTA

Los usuarios con estos permisos también pueden acceder a OBS y SWR de todos los usuarios del proyecto IAM actual.

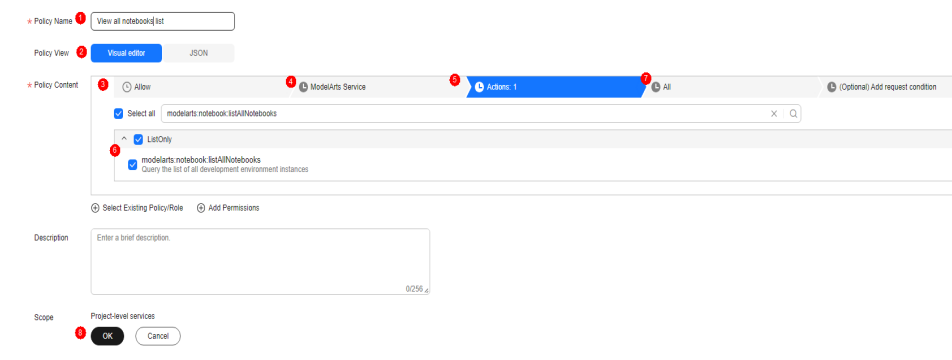
Asignación de los permisos requeridos

1. Inicie sesión en la consola de gestión de ModelArts como usuario tenant, coloque el cursor sobre su nombre de usuario en la esquina superior derecha y elija **Identity and Access Management** en la lista desplegable para cambiar a la consola de gestión de IAM.
2. En la consola de IAM, elija **Permissions > Políticas/Roles** en el panel de navegación, haga clic en **Create Custom Policy** en la esquina superior derecha y cree dos políticas.

Política 1: Cree una política que permita a los usuarios ver todas las instancias de notebook de un proyecto de IAM, como se muestra en [Figura 3-19](#).

- **Policy Name:** introduzca un nombre de política personalizado, por ejemplo, **Viewing all notebook instances**.
- **Policy View:** Seleccione **Visual editor**.
- **Policy Content:** Seleccione **Allow**, **ModelArts Service**, **modelarts:notebook:listAllNotebooks** y los recursos predeterminados.

Figura 3-19 Creación de una política personalizada



Política 2: Cree una política que permita a los usuarios ver todos los usuarios de un proyecto IAM.

- **Policy Name:** introduzca un nombre de política personalizado, por ejemplo, **Viewing all users of the current IAM project**.
- **Policy View:** Seleccione **Visual editor**.
- **Policy Content:** Seleccione **Allow**, **Identity and Access Management**, **iam:users:listUsers** y los recursos predeterminados.

- En el panel de navegación, elija **User Groups**. A continuación, haga clic en **Authorize** en la columna **Operation** del grupo de usuarios de destino. En la página **Authorize User Group**, seleccione las políticas personalizadas creadas en 2 y haga clic en **Next**. A continuación, seleccione el ámbito y haga clic en **OK**.

Después de la configuración, todos los usuarios del grupo de usuarios tienen permiso para ver todas las instancias de notebook creadas por los usuarios del grupo de usuarios.

Si no hay ningún grupo de usuarios disponible, cree un grupo de usuarios, agregue usuarios mediante la función de gestión de grupos de usuarios y configure la autorización. Si el usuario de destino no está en un grupo de usuarios, puede agregarlo a un grupo de usuarios mediante la función de gestión de grupos de usuarios.

Inicio de instancias de notebook de otros usuarios de IAM

Si un usuario de IAM desea acceder a la instancia de notebook de otro usuario de IAM a través de SSH remoto, debe actualizar el par de claves SSH a la suya. De lo contrario, se reportará el error **ModelArts.6789**. Para obtener detalles sobre cómo actualizar un par de claves, véase [Modificación de la configuración de SSH para una instancia de notebook](#).

Mensaje de error: ModelArts.6789: Failed to use SSH key pair KeyPair-xxx. Update the key pair and try again later.

3.11 Consulta de eventos de notebook

Los estados de las instancias y las operaciones clave, como crear, iniciar y detener una instancia y cambiar la variante de la instancia, se registran en el backend. Puede ver los eventos en la página de detalles de la instancia del notebook para supervisar los estados de las instancias. Puede actualizar los eventos en la parte derecha de la ficha **Event**. También puede establecer el intervalo para actualizar eventos automáticamente en 30 segundos, 1 minuto o 5 minutos.

Figura 3-20 Consulta de eventos de instancia de notebook y configuración de actualización automática

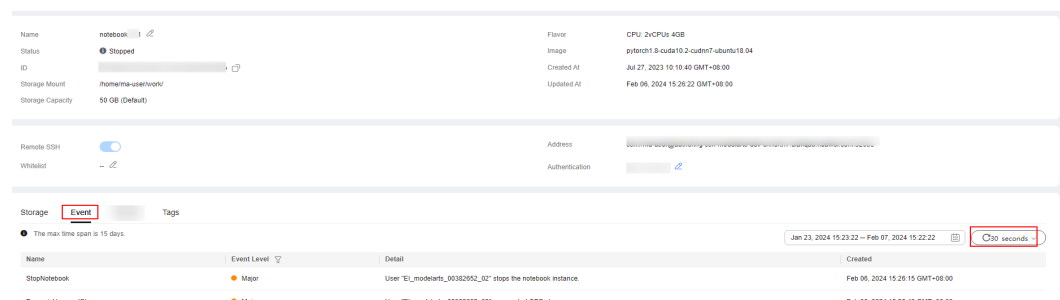


Tabla 3-5 Eventos durante la creación de instancias

Evento	Descripción	Severidad
Scheduled	Se ha programado la instancia.	Warning
PullingImage	Se está sacando la imagen.	Warning

Evento	Descripción	Severidad
PulledImage	Se ha sacado la imagen.	Warning
NotebookHealthy	La instancia está en ejecución y en buen estado.	Mayor
CreateNotebookFailed	No se pudo crear la instancia.	Critical
PullImageFailed	No se pudo sacar la imagen.	Critical

Tabla 3-6 Eventos durante el inicio de la instancia

Nombre del evento	Descripción	Severidad
Scheduled	Se ha programado la instancia.	Warning
PullingImage	Se está sacando la imagen.	Warning
PulledImage	Se ha sacado la imagen.	Warning
NotebookHealthy	La instancia está en ejecución y en buen estado.	Mayor
RunHookScript	Ejecución de un script personalizado	Warning
StartNotebookFailed	Error al iniciar la instancia.	Critical
PullImageFailed	No se pudo sacar la imagen.	Critical
CreateKernelFailed	Error al crear un Jupyter kernel porque el comando conda no está disponible. (The conda environments are not being detected and added as Jupyter kernels. Ensure that <code>{conda_env}</code> is available and the command <code>{conda_cmd}</code> env list can be run properly.)	Mayor
	Error al crear un Jupyter kernel debido a problemas de permisos. (Kernels are not showing up in Jupyter Notebook due to permission issues. Ensure that the uid <code>{ma_uid}</code> has write permissions on <code>{conda_path}</code> .)	Mayor
ConfigurationError	Error al configurar los SDK de ModelArts y las rutas de CLI en el entorno de conda debido a que el comando conda no está disponible. (The ModelArts SDK and CLI are unavailable in the conda environments due to conda environment issues. Ensure that <code>{conda_env}</code> is available and the command <code>{conda_cmd}</code> env list can be run properly.)	Mayor

Nombre del evento	Descripción	Severidad
	<p>Error al configurar los SDK de ModelArts y las rutas de CLI en el entorno de conda debido a problemas de permisos.</p> <p>(The ModelArts SDK and CLI are unavailable in the conda environments due to conda environment issues. Ensure that the uid <code>{ma_uid}</code> has write permissions on <code>{conda_path}</code>.)</p>	Mayor

Tabla 3-7 Eventos durante la detención de instancias

Evento	Descripción	Severidad
StopNotebook	Se detuvo la instancia.	Mayor
StopNotebookResourceIdle	La instancia de notebook se detendrá automáticamente o se ha detenido automáticamente porque los recursos están inactivos.	Mayor

Tabla 3-8 Eventos durante la actualización de instancias

Evento	Descripción	Severidad
UpdateName	Actualización del nombre de instancia	Warning
UpdateDescription	Actualización de la descripción de la instancia	Warning
UpdateFlavor	Actualización de la variante de instancia	Mayor
UpdateImage	Actualización de la imagen de instancia	Mayor
UpdateStorageSize	Se está actualizando el tamaño de almacenamiento de la instancia. (User %s is updating storage size from %s GB to %s GB.)	Mayor
	Se ha actualizado el tamaño de almacenamiento de la instancia. (User %s updated the storage size.)	Mayor

Evento	Descripción	Severidad
UpdateKeyPair	Se configuró el par de claves de instancia. (User %s updated the instance key pair to {%s}.)	Mayor
	Actualización del par de claves de instancia (User %s updated the instance key pair from %s to %s.)	Mayor
UpdateWhitelist	Actualización de la lista blanca de acceso a instancias	Mayor
UpdateHook	Actualización de un script personalizado	Mayor
UpdateStorageSizeFailed	Error al actualizar el tamaño del almacenamiento porque se agotaron los recursos. (EVS disks are sold out.)	Critical
	Error al actualizar el tamaño del almacenamiento debido a un error interno. (Updating the EVS disk size failed. The O&M personnel are handling the fault.)	Critical

Tabla 3-9 Eventos durante el guardado de imágenes

Evento	Descripción	Severidad
SaveImage	Se guardó la imagen.	Mayor
SavedImageFailed	No se pudo guardar la imagen debido a procesos en estado D. (There are processes in 'D' status. Check process status using 'ps -aux' and kill all the processes in 'D' status.)	Critical
	No se pudo guardar la imagen porque es demasiado grande. (The container size (%dG) is greater than the threshold (%dG).)	Critical

Evento	Descripción	Severidad
	No se pudo guardar la imagen debido al límite en el número de capas. (There are too many layers in your image.)	Critical
	No se pudo guardar la imagen debido al tiempo de espera de la tarea. (The O&M personnel are handling the fault.)	Critical
	No se pudo guardar la imagen debido a problemas con el servicio de SWR.	Critical

Tabla 3-10 Eventos durante la ejecución de la instancia

Nombre del evento	Descripción	Severidad
NotebookUnhealthy	La instancia no funciona correctamente.	Critical
OutOfMemory	La instancia no tiene memoria.	Critical
JupyterProcessKilled	Se detuvo el proceso de Jupyter.	Critical
CacheVolumeExceedQuota	El tamaño del archivo / cache ha superado el límite superior.	Critical
NotebookHealthy	Se ha restablecido el estado saludable de la instancia.	Major
EVSSoldOut	Los discos de EVS están agotados.	Critical

Tabla 3-11 Eventos para el montaje dinámico de OBS

Evento	Descripción	Severidad
DynamicMountStorage	Se monta el almacenamiento de OBS.	Major
DynamicUnmountStorage	El almacenamiento del OBS no está montado.	Major

Tabla 3-12 Eventos activados en el lado del usuario

Evento	Descripción	Severidad
RefreshCredentialsFailed	Error de autenticación.	Critical

3.12 Informes de alarmas del directorio de caché de notebook

Al crear una instancia de notebook, puede seleccionar recursos de CPU, GPU o Ascend en función del volumen de datos del servicio. Si selecciona recursos de GPU o de Ascend, ModelArts monta los discos duros en el directorio de caché. Puede utilizar este directorio para almacenar archivos temporales.

Por defecto, no se generan alarmas de capacidad para el directorio de caché de la instancia de notebook. Si se supera el límite de capacidad, se reiniciará la instancia de notebook. Después del reinicio, se restablecen varias configuraciones, lo que descarta los datos y pierde el entorno. Esto afectará su experiencia. Se recomienda habilitar el monitoreo y las alarmas para el uso del directorio de caché y reportar los datos a AOM.

Proceso de configuración

1. Ingrese la información básica de las alarmas.
2. **Configure una regla de alarmas.**
 - a. Configure las métricas de monitoreo.
 - b. Configure las condiciones de activación de alarmas.
3. **Configure las notificaciones de alarmas.**
 - a. Cree un tema, configure la política de temas y suscríbese al tema.
 - b. Cree una regla de acción de alarmas.
 - c. Seleccione la regla de acción creada.

Configuración de ajustes de alarmas

1. Inicie sesión en la consola de AOM.
2. Seleccione **Alarm Center > Alarm Rules** y haga clic en **Create Alarm Rule**.
3. Ingrese la información básica de las alarmas.

Basic Information

* Rule Name

Description

0/1,024

4. Establezca una regla de alarma.

Rule Type: seleccione **Threshold alarm**.

Monitored Object: seleccione **Select resource objects**. Haga clic en **Select Resource Object**. Aparecerá un nuevo cuadro de diálogo.

– **Add By:** seleccione **Dimension**.

– **Metric Name:** haga clic en **Custom Metrics** y seleccione las métricas de caché que desea supervisar. Ejemplo: **ma_container_notebook_cache_dir_size_bytes** (tamaño total del directorio caché) y **ma_container_notebook_cache_dir_util** (uso del directorio caché)

– **Dimension:** seleccione una dimensión métrica, por ejemplo **service_id:xxx** y haga clic en **Confirm**.

Después de configurar el objeto monitoreado, configure **Statistic** y **Statistical Period**.

Alarm Condition: configure este parámetro en función de sus necesidades.

Figura 3-21 Selección de objeto monitoreado

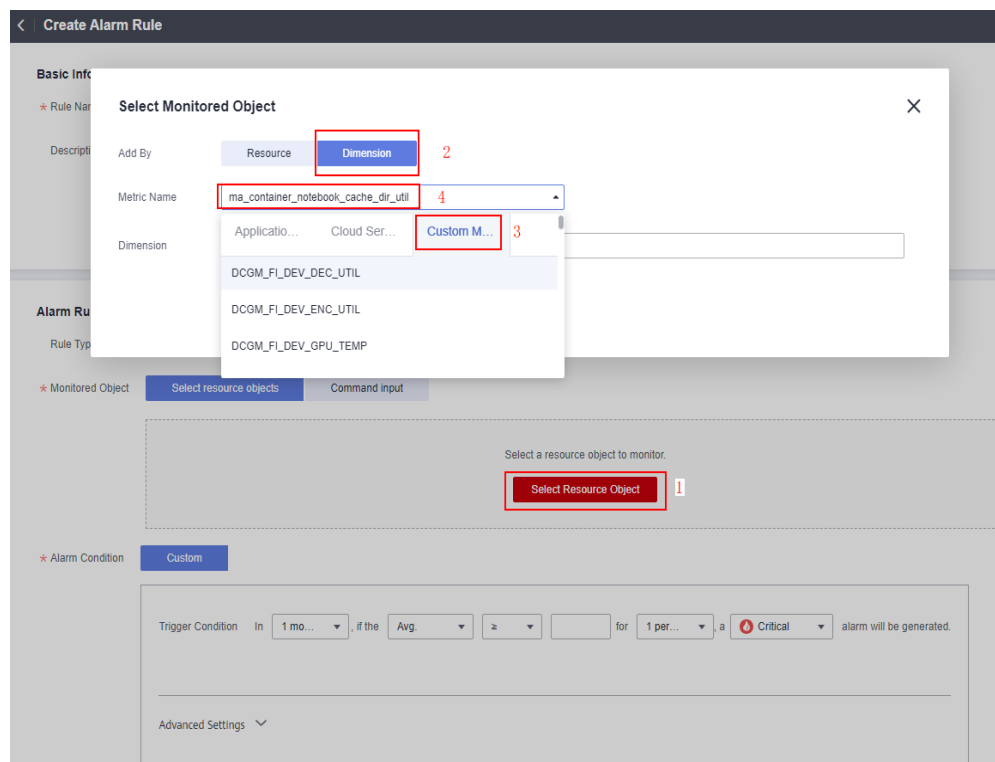


Figura 3-22 Configuración del método de estadísticas

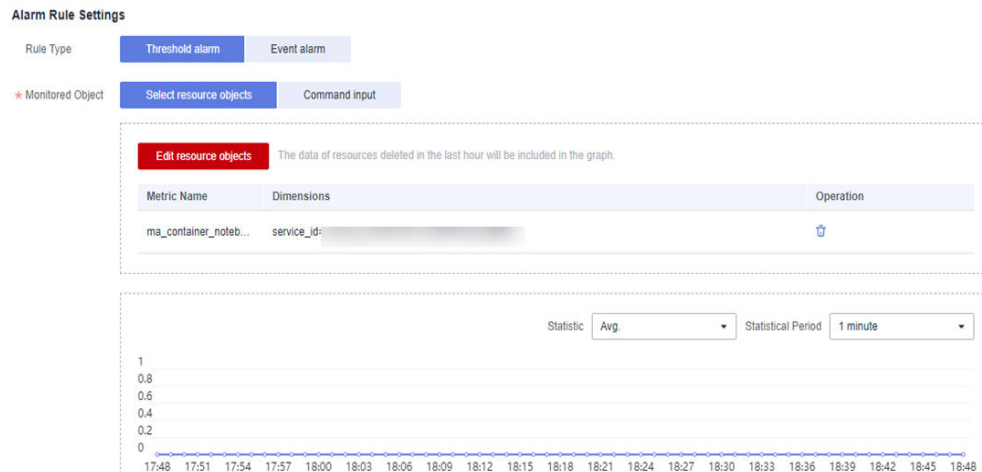
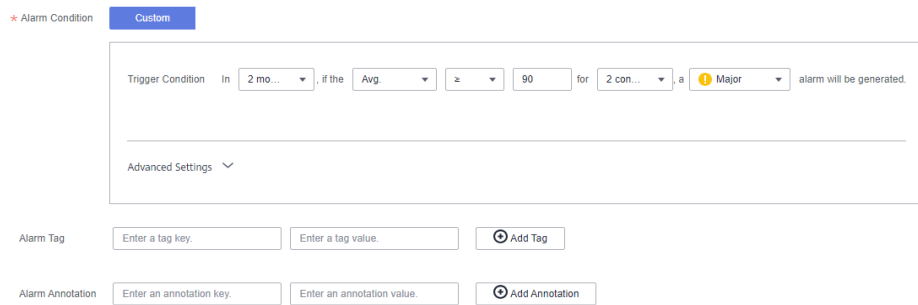


Figura 3-23 Configuración de condiciones de alarma



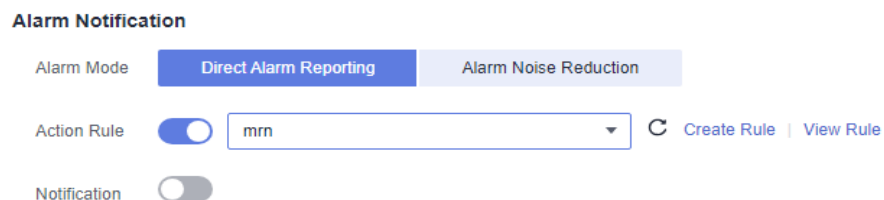
5. Configure las notificaciones de alarmas y haga clic en **Create Now**.

Alarm Mode: seleccione **Direct Alarm Reporting**.

Action Rule: actívelo y seleccione la regla de acción creada. Si las reglas de acción de alarmas existentes no pueden satisfacer sus requerimientos, haga clic en **Create Rule** para crear una regla de acción. Para obtener más detalles, véase [Creación de una regla de acción de alarma](#).

Notification: Habilite esta opción.

Figura 3-24 Configuración de notificaciones de alarma



Cree un tema en SMN para configurar las reglas de notificación de alarmas.

– **Creación de un tema**

- i. Vaya a la consola de SMN. En el panel de navegación, seleccione **Topic Management > Topics**.

- ii. Haga clic en **Create Topic**. Introduzca un nombre de tema, seleccione un proyecto empresarial y haga clic en **OK**.
- iii. Localice el tema de destino y seleccione **More > Configure Topic Policy** en la columna **Operation**.
Seleccione **APM** para permitir que las alarmas de AOM activen SMN.

Figura 3-25 Configurar política de temas

Configure Topic Policy

Topic Name test

Policy [?](#) Basic

Users who can publish messages to this topic

Topic creator

All users

Specified user accounts

Enter one or more account IDs or URNs, each on a separate line.

[Learn how to obtain an account ID.](#)

Services that can publish messages to this topic

OBS DWS APM AAD EFS VOD

MPC LTS CTS

OK Cancel

- iv. Haga clic en **Add Subscription** en la columna **Operation** del tema. Una vez que la suscripción es exitosa, se recibe una notificación una vez que se cumplen las condiciones de alarma.
Seleccione un protocolo, como correo electrónico o SMS, e introduzca los puntos de conexión, como direcciones de correo electrónico o números de teléfono móvil. Haga clic en **OK**.

Add Subscription

Basic Information

Topic Name

* Protocol

* Endpoint

[+ Add Endpoint](#)
[Batch Add Endpoints](#)

Se muestra un registro en la lista de suscripción, pero el registro se encuentra en estado **Unconfirmed**.

Subscription URN	Protocol	Endpoint	Request Header	Description	Topic Name	Status	Operation
<input type="checkbox"/> um.smn.ap-southeast-1.1b0...	SMS		-	-	test	Unconfirmed	Request Confirmation Delete

Después de recibir el correo electrónico, confirme la suscripción.

Luego, la suscripción se encuentra en estado confirmado.

– Creación de una regla de acción de alarmas

Una regla de acción específica como AOM le notifica cuando se activa una alarma. Una vez habilitada una regla de acción de alarmas, el sistema envía notificaciones basadas en el tema de SMN asociado y la plantilla de mensajes.

Introduzca el nombre de la regla de acción, seleccione el tipo de regla de acción, seleccione el tema creado en [el paso anterior](#), seleccione una plantilla de mensaje y haga clic en **Confirm**.

Figura 3-26 Crear regla de acción de alarma

Create Alarm Action Rule

* Rule Name
Enter 1 to 100 characters and do not start or end with an underscore () or hyphen (-). Only letters, digits, underscores, and hyphens ; allowed.

Description
0/1,024
Enter up to 1,024 characters. Only letters, digits, space, and special characters (_) are allowed. Do not start or end with an underscore ().

* Action Type

* Topic C
If you do not see a topic you like, create one on the SMN console.

* Message Template C [Create Template](#) | [View Template](#)

En el área **Alarm Notification** de la página **Create Alarm Rule**, configure **Action Rule** en la regla de acción de alarmas recién creada y haga clic en **Create Now**.

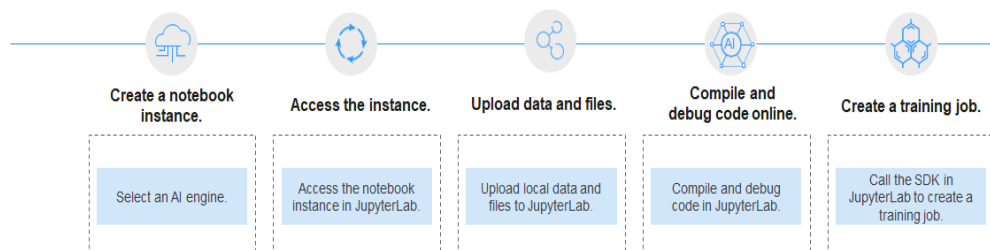
Una vez finalizada la configuración, recibirá una notificación por correo electrónico una vez que se cumplan las condiciones de alarma.

4 JupyterLab

4.1 Proceso de operación de JupyterLab

ModelArts le permite acceder a las instancias de notebook en línea usando JupyterLab y desarrollar modelos de IA basados en los motores de PyTorch, de TensorFlow o de MindSpore. La siguiente figura muestra el proceso de operación.

Figura 4-1 Uso de JupyterLab para desarrollar y depurar código en línea



1. Cree una instancia de notebook.
En la consola de gestión de ModelArts, cree una instancia de notebook con un motor de IA adecuado. Para obtener más detalles, véase [Creación de una instancia de notebook](#).
2. Utilice JupyterLab para acceder a la instancia del notebook. Para obtener más detalles, véase [Acceso a JupyterLab](#).
3. Suba datos de entrenamiento y archivos de código a JupyterLab. Para obtener más detalles, véase [Carga de archivos desde una ruta local a JupyterLab](#).
4. Compile y depure código de JupyterLab. Para obtener más detalles, véase [Descripción y operaciones comunes de JupyterLab](#).
5. En el JupyterLab, invoque al SDK de ModelArts para crear un trabajo de entrenamiento para el entrenamiento en la nube.
Para obtener más información, véase [Creación de un trabajo de entrenamiento](#).

4.2 Descripción y operaciones comunes de JupyterLab

JupyterLab es el entorno de desarrollo interactivo basado en la web de próxima generación de Jupyter Notebook, que le permite compilar notebooks, operar terminales, editar texto de Markdown, habilitar la interacción y ver archivos e imágenes de CSV.

JupyterLab es el futuro entorno de desarrollo convencional para desarrolladores. Tiene los mismos componentes que Jupyter Notebook, pero ofrece funciones más flexibles y potentes.

Acceso a JupyterLab

Para acceder a JupyterLab desde una instancia de notebook en ejecución, realice las siguientes operaciones:

1. Inicie sesión en la consola de gestión de ModelArts. In the navigation pane on the left, choose **DevEnviron** > **Notebook**.
2. Haga clic en **Open** en la columna **Operation** de una instancia de notebook en ejecución para acceder a JupyterLab.

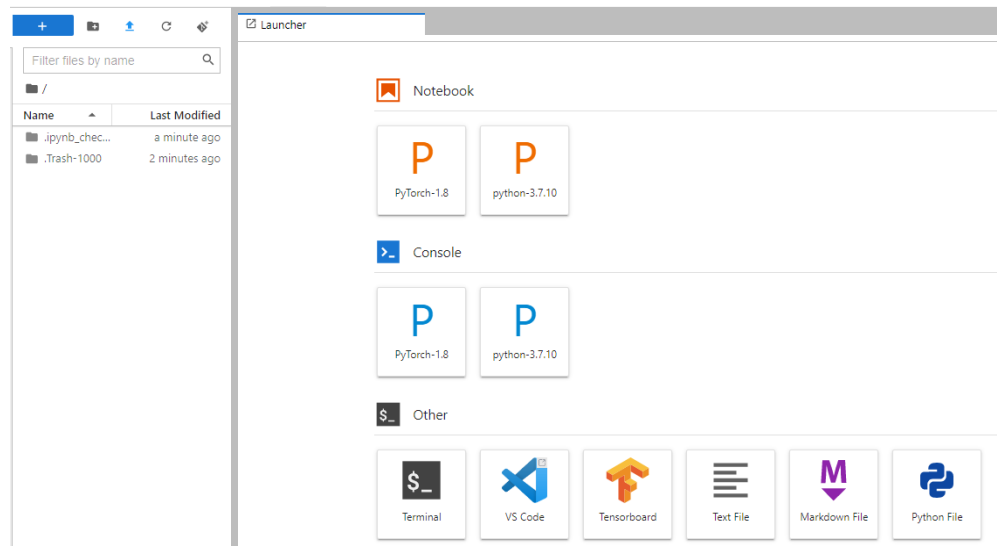
Figura 4-2 Acceso a una instancia de notebook



Name	Status	Image	Flavor	Description	Created At	Created By	Operation
notebook-f799 407607a6-746c-439c-607b-672963a77360	Running (158 minutes)	sparc3.1.1-ubuntu/18.04	CPU: 2vCPUs 8GB	--	Mar 20, 2024 11:16:32 GM...		Open Start Stop More

3. La página **Launcher** se muestra automáticamente. Realice las operaciones requeridas. Para obtener más información, consulte la [Documentación de JupyterLab](#).

Figura 4-3 Página de inicio de JupyterLab



NOTA

Los kernels y las versiones de notebook y de la consola que se muestran en la página **Launcher** varían según el motor de IA basado en el que se cree una instancia de notebook. **Figura 4-3** muestra solo un ejemplo. Obtenga los núcleos y las versiones de notebook y de la consola en la consola de gestión.

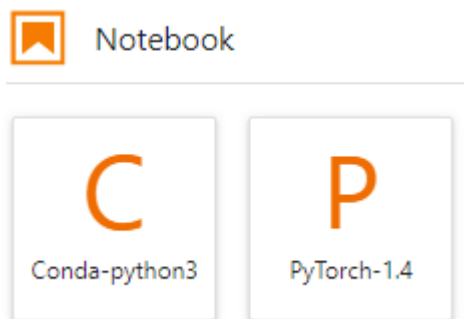
- **Notebook:** Seleccione un kernel para ejecutar notebook, por ejemplo, TensorFlow o Python.
- **Console:** Llame al terminal para el control de comandos.
- **Other:** Edite otros archivos.

Creación de un archivo de IPYNB de JupyterLab

En la página de inicio del JupyterLab, haga clic en un motor de IA adecuado en el área **Notebook** para crear un archivo de IPYNB.

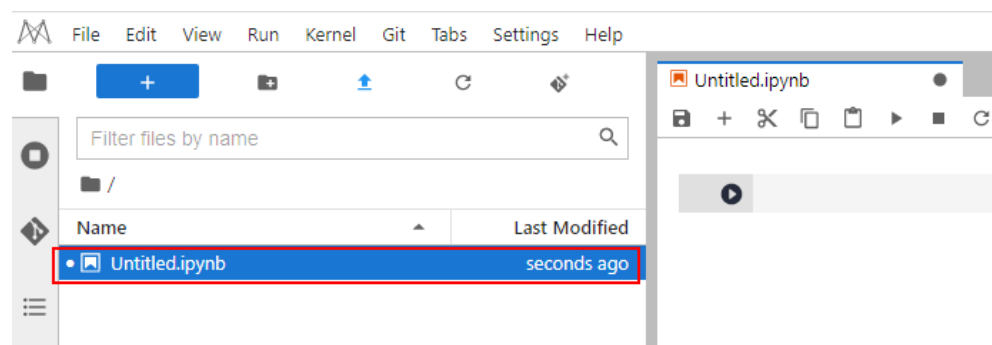
Los motores de IA compatibles con cada instancia de notebook varían según el entorno de tiempo de ejecución. La siguiente figura es solo un ejemplo. Seleccione un motor de IA basado en los requisitos del sitio.

Figura 4-4 Seleccionar un motor de AI y crear un archivo de IPYNB



El archivo de IPYNB creado se muestra en el panel de navegación de la izquierda.

Figura 4-5 Archivo de IPYNB creado



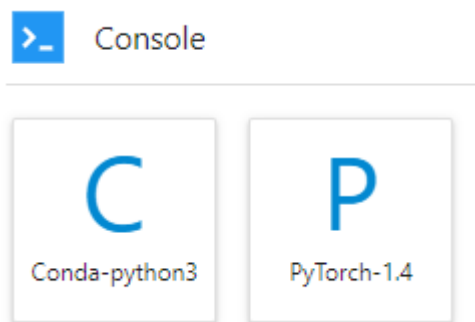
Creación de un archivo de notebook y acceso a la consola

Una consola es un terminal de Python, que es similar al IDE nativo de Python, mostrando la salida después de que se introduce una instrucción.

En la página de inicio del JupyterLab, haga clic en un motor de IA adecuado en el área **Console** para crear un archivo de notebook.

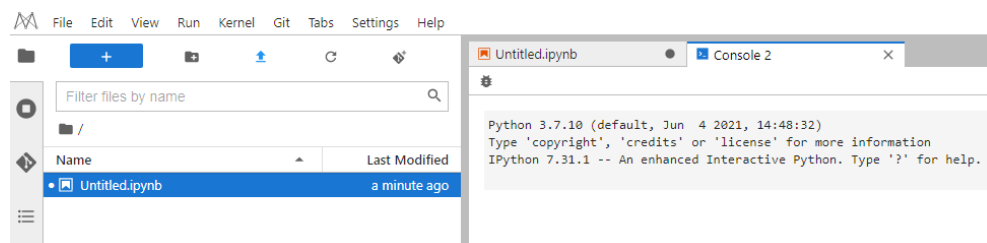
Los motores de IA admitidos por cada instancia de notebook varían en función del entorno de tiempo de ejecución. La siguiente figura es solo un ejemplo. Seleccione un motor de IA basado en los requisitos del sitio.

Figura 4-6 Selección de un motor de IA y creación de una consola



Después de crear el archivo, se muestra la página de la consola.

Figura 4-7 Creación de un archivo de notebook (consola)

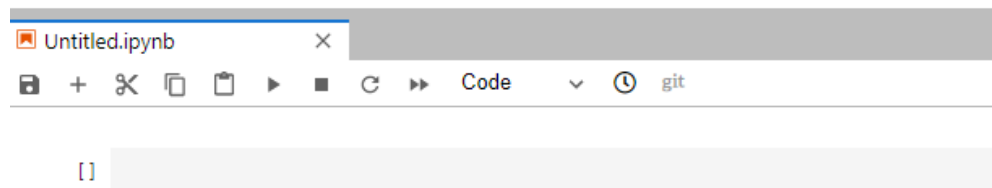
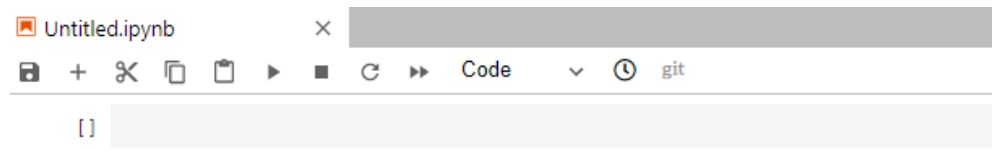


Edición de un archivo de JupyterLab

JupyterLab le permite abrir múltiples instancias o archivos de notebook (como archivos HTML, TXT y Markdown) en una ventana y los muestra en las diferentes pestañas.

Con JupyterLab puede personalizar la demostración de varios archivos. En el área de demostración de archivos de la derecha, puede arrastrar un archivo para ajustar su posición. Se pueden mostrar varios archivos simultáneamente.

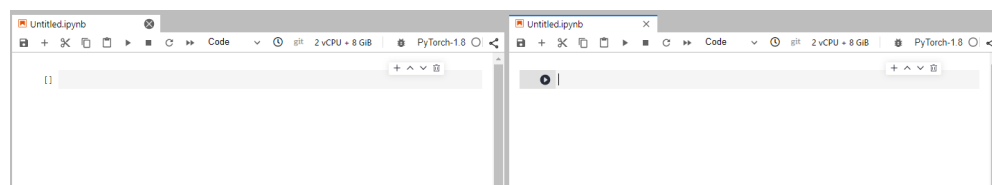
Figura 4-8 Demostración personalizada de varios archivos



Al escribir código en una instancia de notebook, puede crear varias vistas de un archivo para editarlo de forma sincrónica y ver los resultados de ejecución en tiempo real.

Para abrir varias vistas, abra un archivo de IPYNB y elija **File > New View for Notebook**.

Figura 4-9 Varias vistas de un archivo



Antes de codificar en el área de código de un archivo de IPYNB de JupyterLab, agregue un signo de exclamación (!) antes del código.

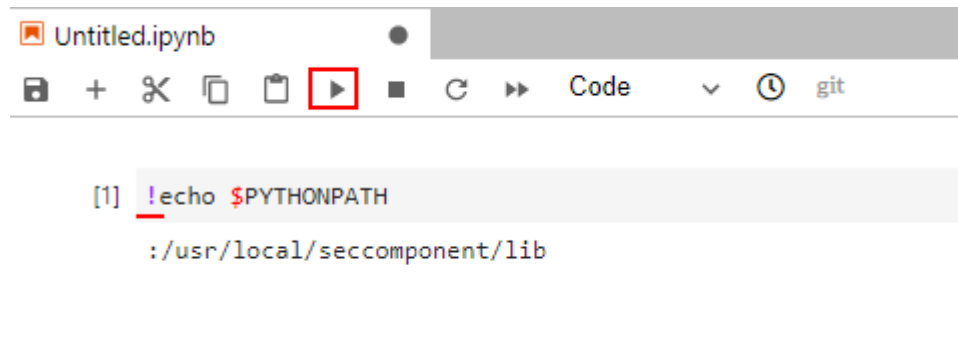
Por ejemplo, instale una biblioteca externa Shapely.

```
!pip install Shapely
```

Por ejemplo, obtenga PythonPath.

```
!echo $PYTHONPATH
```

Figura 4-10 Código de ejecución



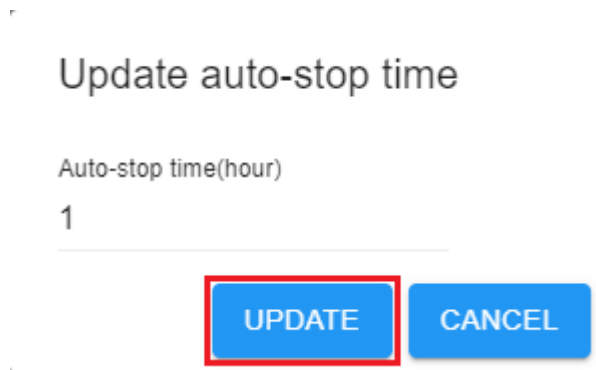
Renovación o detención automática de una instancia de notebook

Si habilita la detención automática al crear o iniciar una instancia de notebook, la duración restante para detener la instancia se muestra en la esquina superior derecha de JupyterLab. Puede hacer clic en la hora de renovación.

Figura 4-11 Período restante



Figura 4-12 Renovación de una instancia



Botones y complementos comunes de JupyterLab

Figura 4-13 Botones y complementos comunes de JupyterLab

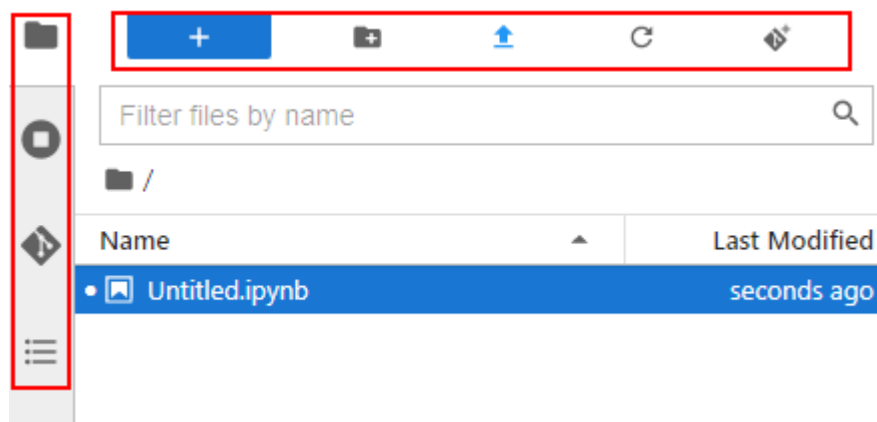


Tabla 4-1 Botones de JupyterLab

Botón	Descripción
	Abra rápidamente instancias y terminales de notebook. Abra la página Launcher , en la que puede crear rápidamente instancias de notebook, consolas u otros archivos.
	Cree una carpeta.
	Cargue los archivos.
	Actualice el directorio de archivos.
	Complemento de Git, que se puede utilizar para acceder a la biblioteca de códigos de GitHub asociada a la instancia de notebook.

Tabla 4-2 Complementos de JupyterLab

Complemento	Descripción
	Liste los archivos. Haga clic en este botón para mostrar todos los archivos de la instancia de notebook.
	Muestra los terminales y kernels que se están ejecutando en la instancia actual.
	Complemento de Git, que se puede utilizar para acceder rápidamente a la biblioteca de códigos de GitHub.



Complemento	Descripción
	Inspector de propiedades.
	Muestre la organización del documento.

Figura 4-14 Botones en la barra de navegación

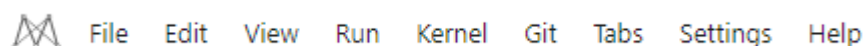












Tabla 4-3 Botones en la barra de navegación

Botón	Descripción
File	Acciones relacionadas con archivos y directorios, como crear, cerrar o guardar notebooks.
Edit	Acciones relacionadas con la edición de documentos y otras actividades en el archivo de IPYNB, como deshacer, rehacer o cortar celdas.
View	Acciones que alteran la apariencia de JupyterLab como mostrar la barra o expandir el código.
Run	Acciones para ejecutar código en diferentes actividades como notebooks y consolas de código.
Kernel	Acciones para gestionar kernels, como interrumpir, reiniciar o cerrar un kernel.
Git	Acciones en el complemento de Git, que se puede utilizar para acceder rápidamente a la biblioteca de código de GitHub.
Tabs	Una lista de los documentos y actividades abiertos en el panel de acoplamiento.
Settings	Ajustes comunes y un editor de ajustes avanzados.
Help	Una lista de JupyterLab y enlaces de ayuda del núcleo.

Figura 4-15 Botones en la barra de menús de un archivo de IPYNB



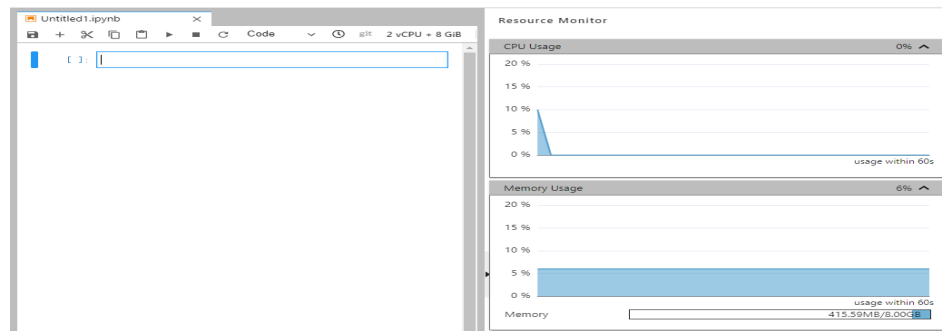
Tabla 4-4 Botones en la barra de menús de un archivo de IPYNB

Botón	Descripción
	Guardar un archivo.
+	Agregar una nueva celda.
	Cortar la celda seleccionada.
	Copiar la celda seleccionada.
	Pegar la celda seleccionada.
	Ejecutar la celda seleccionada.
	Terminar un núcleo.
	Reiniciar un kernel.
	Reiniciar un kernel y volver a ejecutar todo el código de la notebook actual.
Code ▾	Hay cuatro opciones en la lista desplegable: Code (código de Python), Markdown (código de Markdown, típicamente utilizado para comentarios), Raw (una herramienta de conversión) y - (no modificado)
	Ver versiones de código histórico.
git	Complemento de Git. El botón gris indica que el complemento no está disponible en la región actual.
2 vCPU + 4 GiB	Variante de la instancia.
PyTorch-1.4	Kernel para su selección.
	Estado de ejecución de código. ● indica que el código está siendo ejecutado.

Recursos de monitoreo

Para obtener el uso de recursos, seleccione **Resource Monitor** en el panel derecho. Se puede ver el uso de la CPU y el uso de la memoria.

Figura 4-16 Uso del recurso



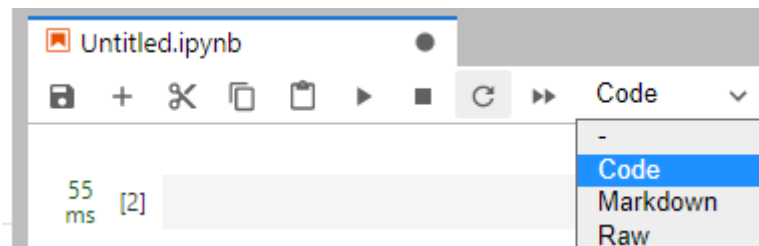
4.3 Complemento de parametrización de código

El complemento de parametrización del código simplifica los casos de notebook. Puede ajustar rápidamente los parámetros y entrenar los modelos según los casos de notebook sin el código complejo. Este complemento se puede utilizar para personalizar los casos de notebook para competiciones y aprendizaje.

Guía de uso

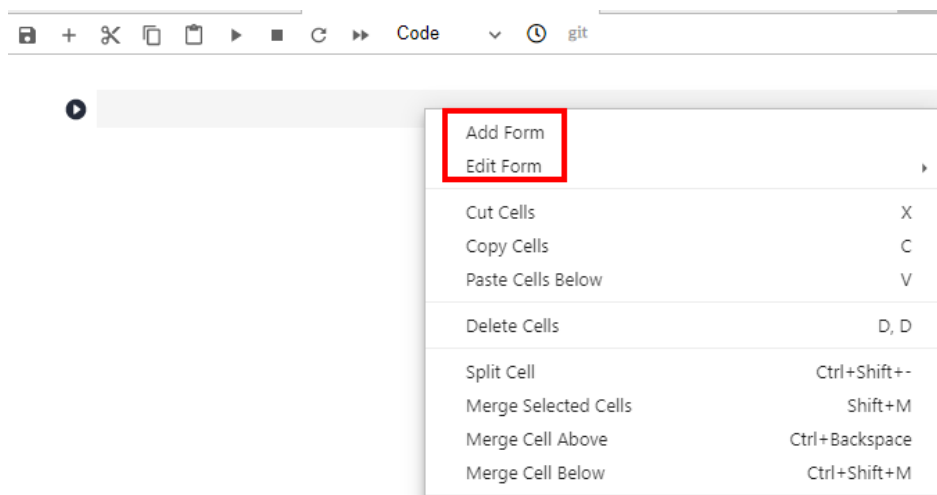
- Los botones **Add Form** y **Edit Form** solo están disponibles en el menú contextual de las celdas de código.

Figura 4-17 Consulta de una celda de código



- Después de abrir el nuevo código, agregue un formulario antes de editarlo.

Figura 4-18 Menú de acceso directo de las celdas de código



Add Form

Si hace clic en **Add Form**, se dividirá una celda de código en el área de edición de código y de formulario. Haga clic en **Edit** a la derecha del formulario para cambiar el título predeterminado.

Figura 4-19 Dos áreas de edición



Edit Form

Si hace clic en **Edit Form**, se mostrarán cuatro subopciones: **Add new form field**, **Hide code**, **Hide form** y **Show All**.

- Puede establecer el tipo de campo de formulario en **dropdown**, **input** y **slider**. Consulte [Figura 4-20](#). Cada vez que se agrega un campo, la variable correspondiente se agrega a las áreas de código y de formulario. Si se cambia un valor en el área de formulario, también se cambia la variable correspondiente en el área de código.

NOTA

Cuando cree un formulario desplegable, haga clic en **ADD Item** y agregue al menos dos elementos. Consulte [Figura 4-21](#).

Figura 4-20 Estilo de formulario de menú desplegable, entrada y control deslizante



Figura 4-21 Crear un formulario desplegable

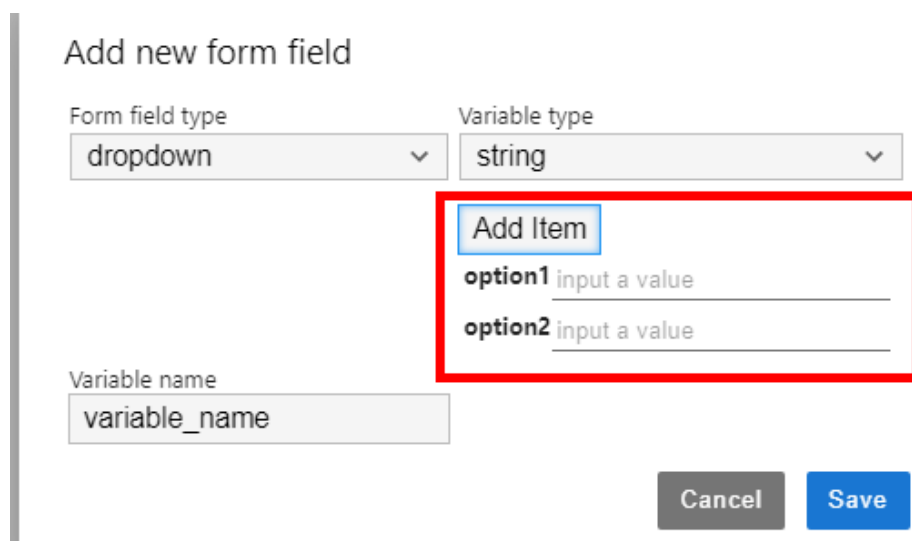


Figura 4-22 Eliminar un formulario



- Si el tipo de campo de formulario se establece en **dropdown**, los tipos de variables admitidos son **raw** y **string**.
- Si el tipo de campo de formulario se establece en **input**, los tipos de variables admitidos son **boolean**, **date**, **integer**, **number**, **raw** y **string**.
- Si el tipo de campo de formulario se establece en **slider**, se pueden establecer el valor mínimo, el valor máximo y el paso.
- Si hace clic en **Hide code**, el área de código se ocultará.
- Si hace clic en **Hide form**, el área de formulario se ocultará.
- Si hace clic en **Show All**, se mostrarán las áreas de código y de formulario.

4.4 Uso del SDK de ModelArts

Las instancias de notebook le permiten usar el SDK de ModelArts para gestionar OBS, trabajos de entrenamiento, modelos y servicios en tiempo real.

Las instancias de notebook han obtenido automáticamente su AK/SK para la autenticación y la región. Por lo tanto, las sesiones del SDK se autentican automáticamente.

Código de ejemplo

- Crear un trabajo de entrenamiento.

```
from modelarts.session import Session
from modelarts.estimator import Estimator
session = Session()
estimator = Estimator(
    modelarts_session=session,

    framework_type='PyTorch', # AI engine name
    framework_version='PyTorch-1.0.0-
python3.6', # AI engine version
    code_dir='/obs-bucket-
name/src/', # Training script directory
    boot_file='/obs-bucket-name/src/
pytorch_sentiment.py', # Training boot script directory
    log_url='/obs-bucket-
name/log/', # Training log directory
    hyperparameters=[
        {"label": "classes",
         "value": "10"},
        {"label": "lr",
         "value": "0.001"}
    ],
    output_path='/obs-bucket-name/
output/', # Training output directory

    train_instance_type='modelarts.vm.gpu.p100', # Training
environment specifications
    train_instance_count=1, # Number of
```

```
training nodes
                                job_description='pytorch-sentiment with ModelArts
SDK')      # Training job description
job_instance = estimator.fit(inputs='/obs-bucket-name/data/train/',
wait=False, job_name='my_training_job')
```

- Obtener una lista de modelos.

```
from modelarts.session import Session
from modelarts.model import Model
session = Session()
model_list_resp = Model.get_model_list(session, model_status="published",
model_name="digit", order="desc")
```

- Obtener los detalles de servicio.

```
from modelarts.session import Session
from modelarts.model import Predictor
session = Session()
predictor_instance = Predictor(session, service_id="input your service_id")
predictor_info_resp = predictor_instance.get_service_info()
```

4.5 Uso de complemento de Git

En JupyterLab puede usar el complemento de Git para clonar el repositorio de código abierto de GitHub, ver y editar datos rápidamente, y enviar los datos modificados.

Requisitos previos

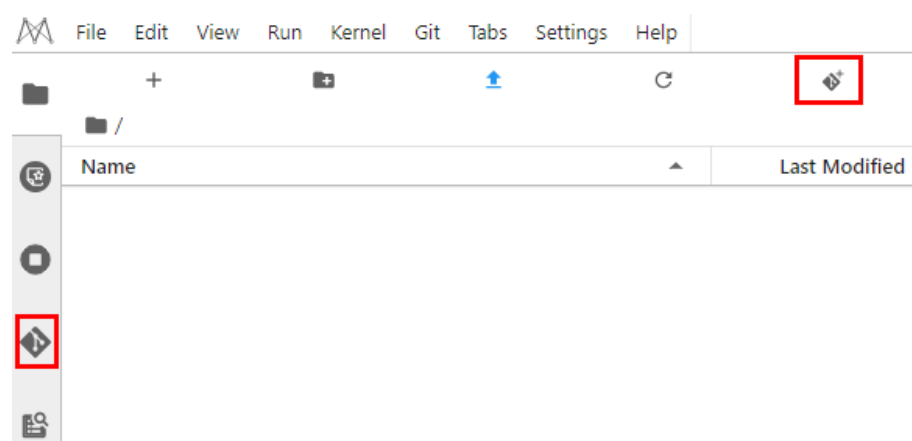
La instancia del notebook se está ejecutando.

Inicio del complemento de Git de JupyterLab


En la lista de instancias de notebook, busque la instancia de destino y haga clic en **Open** en la columna **Operation** para ir a la página de JupyterLab.

Figura 4-23 muestra el complemento de Git de JupyterLab.

Figura 4-23 Complemento de Git

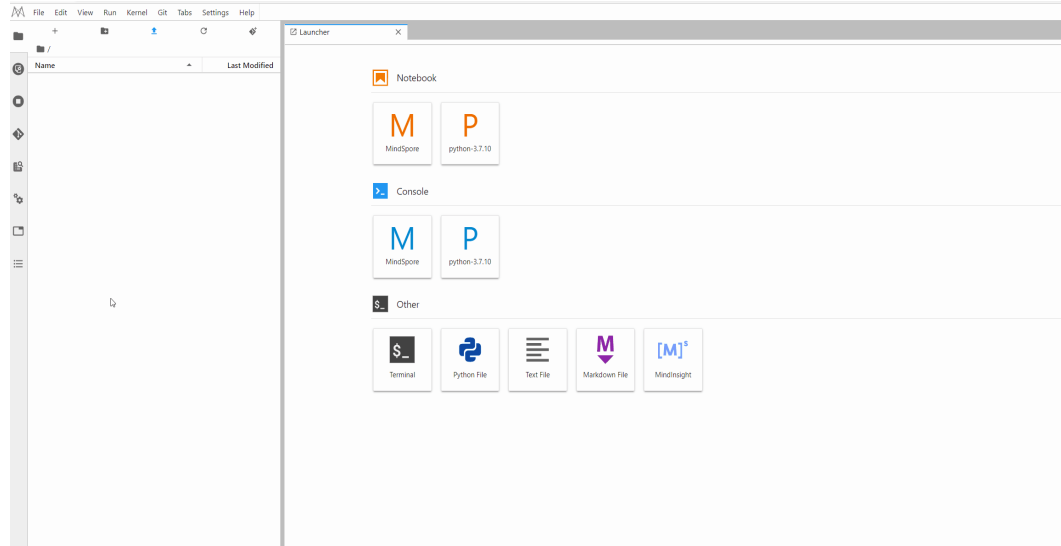


Clonación de un repositorio de código fuente abierto de GitHub

Acceda a un repositorio de código abierto de GitHub en <https://github.com/jupyterlab/extension-examples>. Haga clic en , introduzca la dirección del repositorio y haga

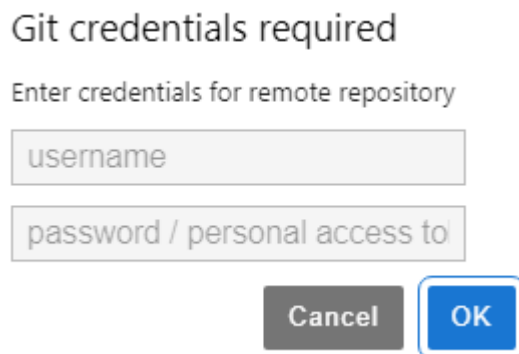
clic en **OK** para iniciar la clonación. Una vez finalizada la clonación, se muestra la carpeta de bibliotecas de códigos en el panel de navegación de JupyterLab.

Figura 4-24 Using the Git plug-in to clone a GitHub open-source code repository



Clonación de un repositorio de código privado de GitHub

Al clonar un repositorio de código privado de GitHub, aparecerá un cuadro de diálogo en el que se le pedirá que introduzca sus credenciales personales. En este caso, introduzca el token de acceso personal en GitHub.



Para obtener un token de acceso personal, realice las siguientes operaciones:

1. Inicie sesión en **GitHub** y abra la página de configuración.
2. Haga clic en **Developer settings**.
3. Elija **Personal access tokens > Generate new token**.
4. Verifique la cuenta.
5. Describa el token, seleccione permisos para acceder al repositorio privado y haga clic en **Generate token** para generar un token.
6. Copie el token generado en CodeArts Build.

AVISO

- Guarde el token de forma segura una vez generado. No estará disponible después de actualizar la página. Si no se obtiene, genere un nuevo token.
- Introduzca una descripción válida del token para poder identificarlo fácilmente. Si el token se elimina por error, la creación fallará.
- Elimine el token cuando ya no se utilice para evitar fugas de información.

Figura 4-25 Clonación de un repositorio de código privado de GitHub (solo se admite autorización mediante un token de acceso personal)

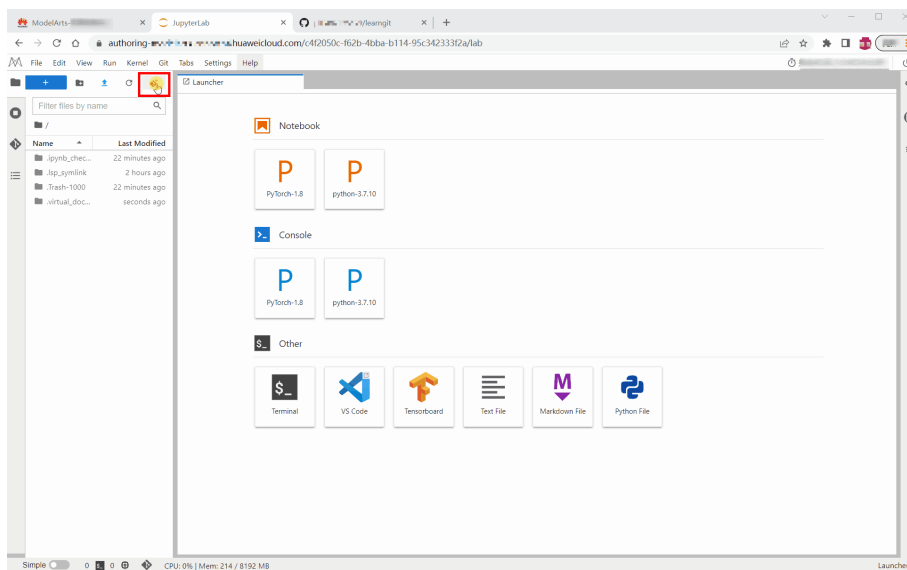
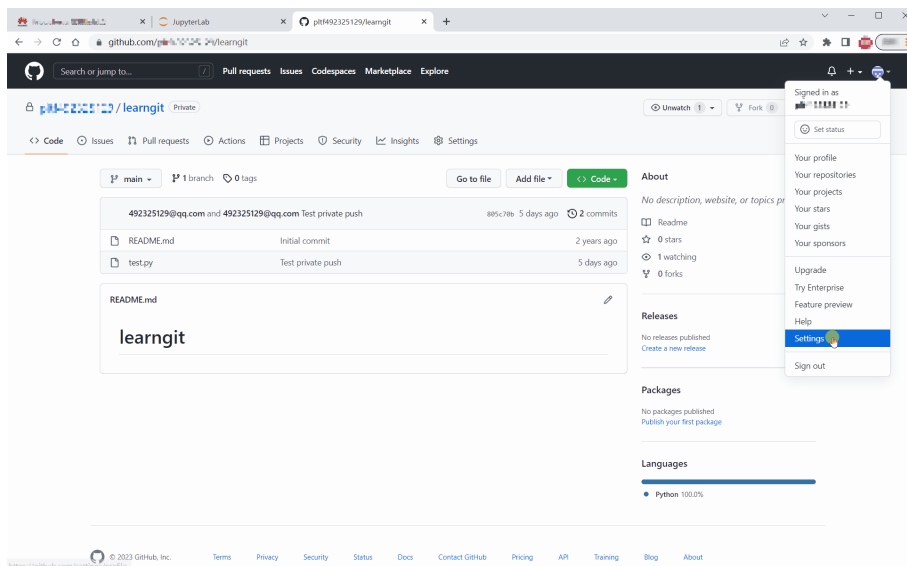


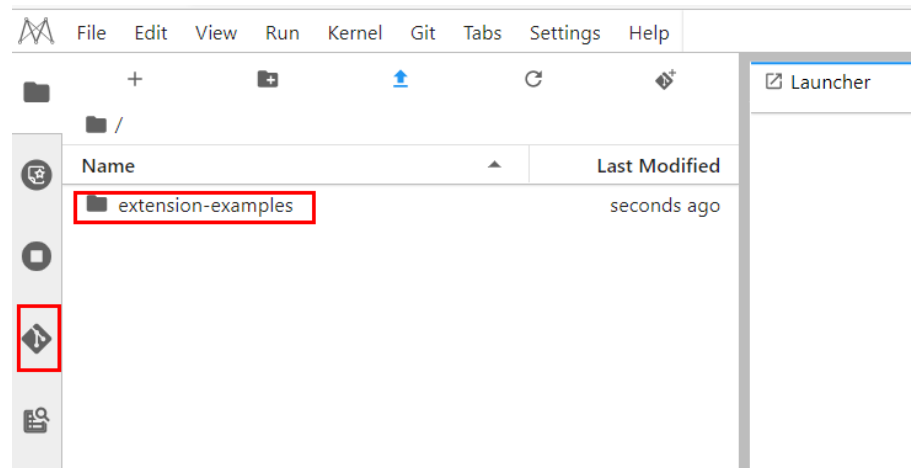
Figura 4-26 Obtención de un token de acceso personal



Consulta de un repositorio de código

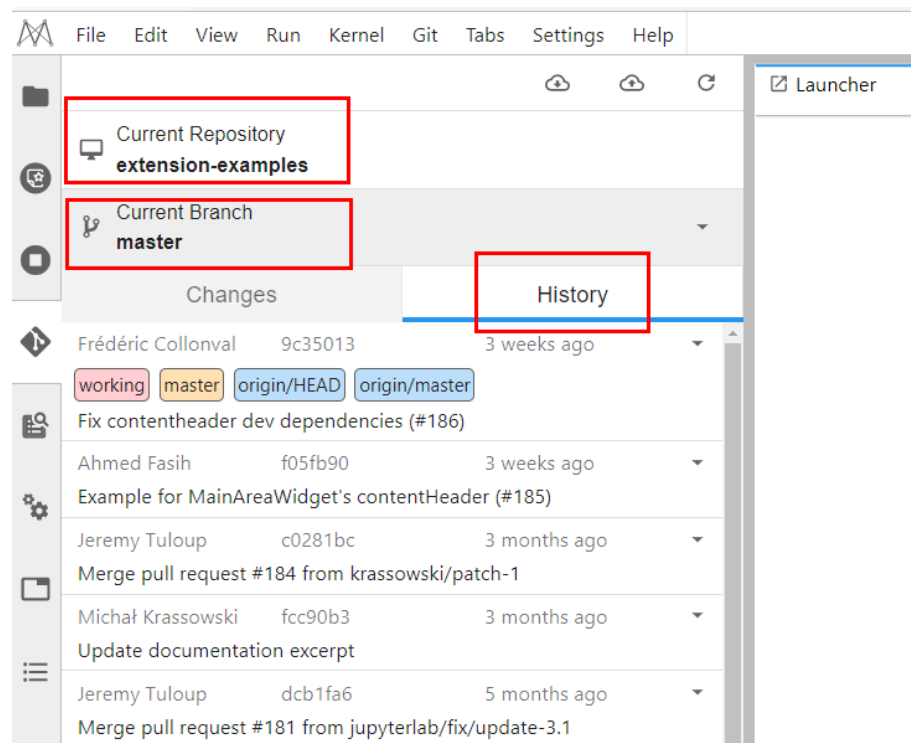
En la lista de **Name**, haga doble clic en la carpeta que quiera usar y haga clic en el icono del complemento de Git en la izquierda para acceder al repositorio de código correspondiente a la carpeta.

Figura 4-27 Abrir la carpeta e iniciar el complemento de Git



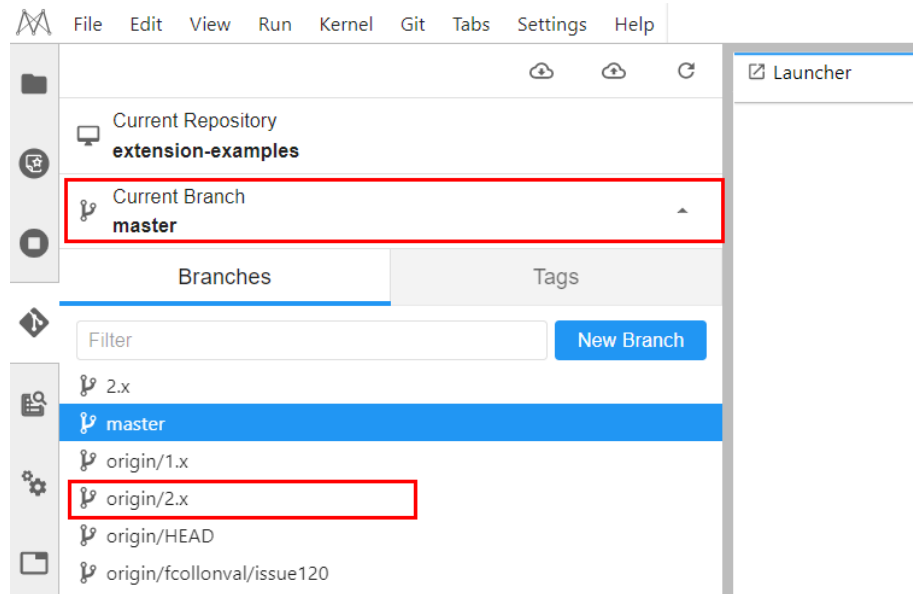
Puede ver la información del repositorio de código actual, como el nombre del repositorio, la rama y los registros de envío históricos.

Figura 4-28 Consulta de un repositorio de código



NOTA

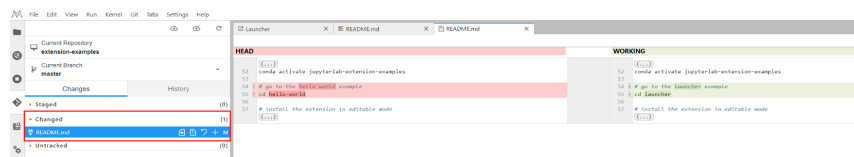
Por defecto, el complemento de Git clona la rama de master. Para cambiar otra rama, haga clic en **Current Branch** para expandir todas las ramas y haga clic en el nombre de la rama de destino.



Consulta de modificaciones

Si se ha modificado un archivo del repositorio de código, puede ver el archivo modificado en **Changed** en la ficha **Changes**. Haga clic en **Diff this file** a la derecha del nombre del archivo para ver las modificaciones.

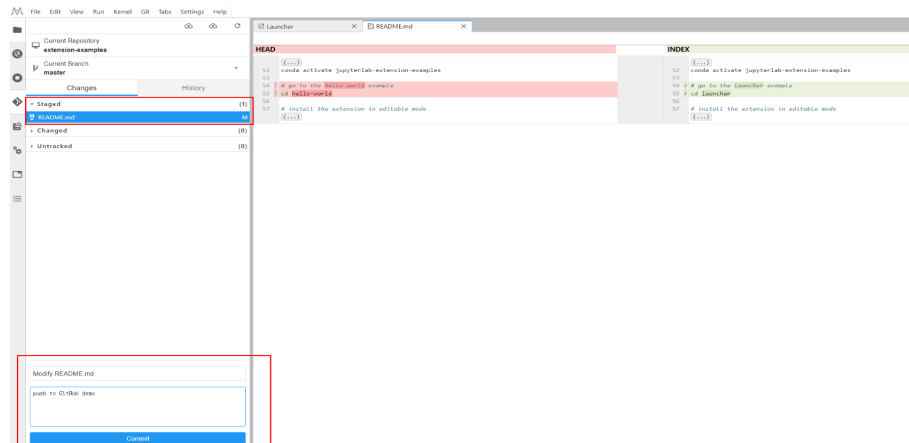
Figura 4-29 Consulta de modificaciones



Confirmación de modificaciones

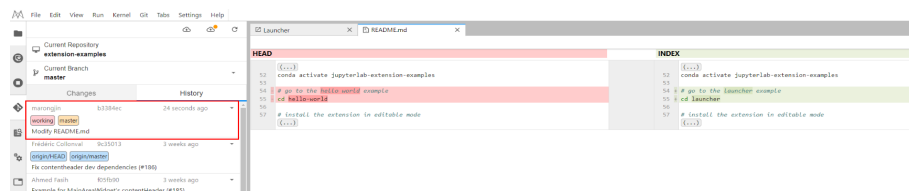
Después de confirmar que las modificaciones son correctas, haga clic en **Stage this change** a la derecha del nombre de archivo, lo que equivale a ejecutar el comando **git add**. El archivo entra en el estado **Staged**. Escriba el mensaje que desea confirmar en la esquina inferior izquierda y haga clic en **Commit** que equivale a ejecutar el comando **git commit**.

Figura 4-30 Confirmación de modificaciones



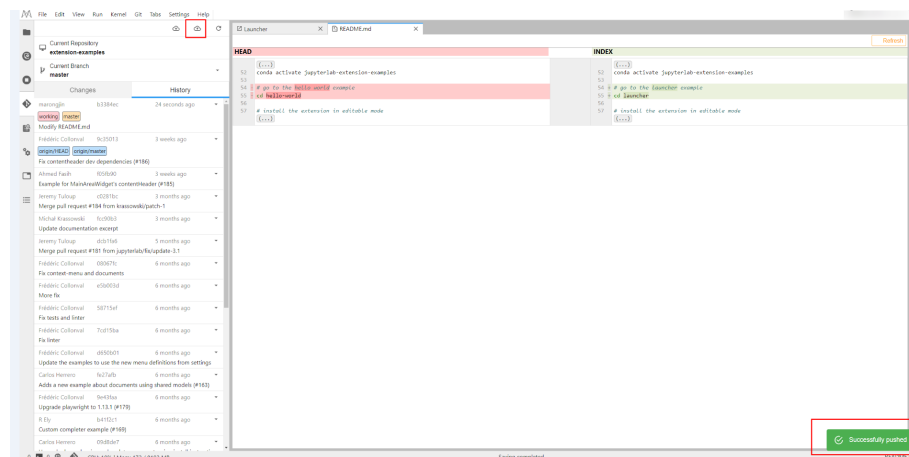
En la página de ficha **History**, vea el estado de confirmación.

Figura 4-31 Comprobar si la confirmación es correcta



Haga clic en el icono **push**, que equivale a ejecutar el comando **git push** para insertar el código en el repositorio de GitHub. Después de que la pulsación se realiza correctamente, se muestra el mensaje "Successfully completed". Si el token utilizado para la autenticación de OAuth ha caducado, aparecerá un cuadro de diálogo en el que se le pedirá que introduzca el token de usuario o la información de la cuenta. Ingrese la información según se le solicite. Esta sección describe la autorización mediante un token de acceso personal. Si utiliza una contraseña para la autorización pero no está disponible, realice las operaciones descritas en [¿Qué debo hacer si la contraseña del complemento de Git no es válida?](#)

Figura 4-32 Envío de código al repositorio de GitHub



Una vez completadas las operaciones anteriores, en la pestaña **History** de la página de complemento de Git de JupyterLab puede ver que **origin/HEAD** y **origin/master** apuntan al último impulso. Además, puede encontrar la información correspondiente en los registros de confirmación del repositorio de GitHub.

4.6 Entrenamiento de modelo visualizado

4.6.1 Introducción a la visualización de trabajos de entrenamiento

Notebook de ModelArts de la nueva versión es compatible con TensorBoard y MindInsight para visualizar los trabajos de entrenamiento. En el entorno de desarrollo, utilice pequeños conjuntos de datos para entrenar y depurar algoritmos, durante los cuales puede comprobar la convergencia del algoritmo y detectar problemas para facilitar la depuración.

Puede crear trabajos de visualización de tipos TensorBoard y MindInsight en ModelArts.

Tanto TensorBoard como MindInsight muestran eficazmente la tendencia de cambio de un trabajo de entrenamiento y los datos utilizados en el entrenamiento.

- **TensorBoard**

TensorBoard muestra eficazmente el gráfico computacional de TensorFlow en el proceso de ejecución, la tendencia de todas las métricas en el tiempo y los datos utilizados en el entrenamiento. Para obtener más información sobre TensorBoard, visite el [sitio web oficial de TensorBoard](#).

Los trabajos de entrenamiento en visualización de TensorBoard solo admiten las variantes de CPU y GPU basadas en imágenes de TensorFlow 2.1 y PyTorch 1.4 y 1.8. Seleccione las imágenes y variantes según los requisitos del sitio.

- **MindInsight**

MindInsight visualiza la información como escalares, imágenes, gráficos computacionales e hiperparámetros modelados durante el entrenamiento. También proporciona funciones como panel de entrenamiento, linaje de modelos, linaje de datos y depuración de rendimiento, lo que le ayuda a entrenar y depurar modelos de manera eficiente. MindInsight apoya los trabajos de entrenamiento de MindSpore. Para obtener más información sobre MindInsight, visite [el sitio web oficial de MindSpore](#).

A continuación se muestran las imágenes y variantes compatibles con los trabajos de entrenamiento en visualización de MindInsight y se seleccionan imágenes y variantes según los requisitos del sitio.

- MindSpore 1.2.0 (CPU o GPU)

Puede utilizar el archivo de resumen generado durante el entrenamiento del modelo para crear un trabajo de visualización en Notebook de DevEnviron.

- Para obtener detalles sobre cómo crear un trabajo de visualización de MindInsight en un entorno de desarrollo, consulte [Trabajos de visualización de MindInsight](#).
- Para obtener detalles sobre cómo crear un trabajo de visualización de TensorBoard en un entorno de desarrollo, consulte [Trabajos de visualización de TensorBoard](#).

4.6.2 Trabajos de visualización de MindInsight

Notebook de ModelArts de la nueva versión soporta los trabajos de visualización de MindInsight. En un entorno de desarrollo, utilice un conjunto de datos pequeño para entrenar

y depurar un algoritmo. Esto se utiliza para comprobar la convergencia de algoritmos y detectar problemas de entrenamiento, lo que facilita la depuración.

MindInsight visualiza la información como escalares, imágenes, gráficos computacionales e hiperparámetros modelados durante el entrenamiento. También proporciona funciones como panel de entrenamiento, linaje de modelos, linaje de datos y depuración de rendimiento, lo que le ayuda a entrenar y depurar modelos de manera eficiente. MindInsight apoya los trabajos de entrenamiento de MindSpore. Para obtener más información sobre MindInsight, visite [el sitio web oficial de MindSpore](#).

MindSpore le permite guardar los datos en el archivo de log de resumen y obtener los datos en la GUI del MindInsight.

Requisitos previos

Cuando utilice MindSpore para editar un script de entrenamiento, agregue el código para recopilar el registro de resumen al script para asegurarse de que el archivo de resumen se genera en el resultado del entrenamiento.

Para obtener más información, consulte [Recopilación de registros de resumen](#).

Nota

- Para ejecutar un trabajo de entrenamiento de MindInsight en un entorno de desarrollo, inicie MindInsight y luego el proceso de entrenamiento.
- Solo se admite el entrenamiento de un nodo único con una tarjeta única.
- Un trabajo de visualización en ejecución no se factura por separado. Cuando se detiene la instancia del notebook de destino, la facturación se detiene.
- Si el archivo de resumen se almacena en OBS, el almacenamiento del OBS se facturará por separado. Después de completar un trabajo, detenga la instancia de notebook y borre los datos de OBS para detener la facturación.

Creación de un trabajo de visualización de MindInsight en un entorno de desarrollo

[Paso 1 Crear un entorno de desarrollo y acceda a él en línea](#)

[Paso 2 Subir los datos de resumen](#)

[Paso 3 Iniciar MindInsight](#)

[Paso 4 Ver datos visualizados en el panel de entrenamiento](#)

Paso 1 Crear un entorno de desarrollo y acceda a él en línea

Inicie sesión en la consola de gestión de ModelArts, elija **DevEnviron** > **Notebook** y cree una instancia de entorno de desarrollo para el motor de MindSpore. Una vez creada la instancia, haga clic en **Open** en la columna **Operation** de la instancia para acceder a ella en línea.

Los tipos de imágenes y recursos admitidos por los trabajos de entrenamiento en visualización de MindInsight son los siguientes:

- MindSpore 1.2.0 (CPU o GPU)
- MindSpore 1.5.x o posterior (Ascend)

Paso 2 Subir los datos de resumen

Se requieren datos de resumen para la visualización de MindInsight en un entorno de desarrollo.

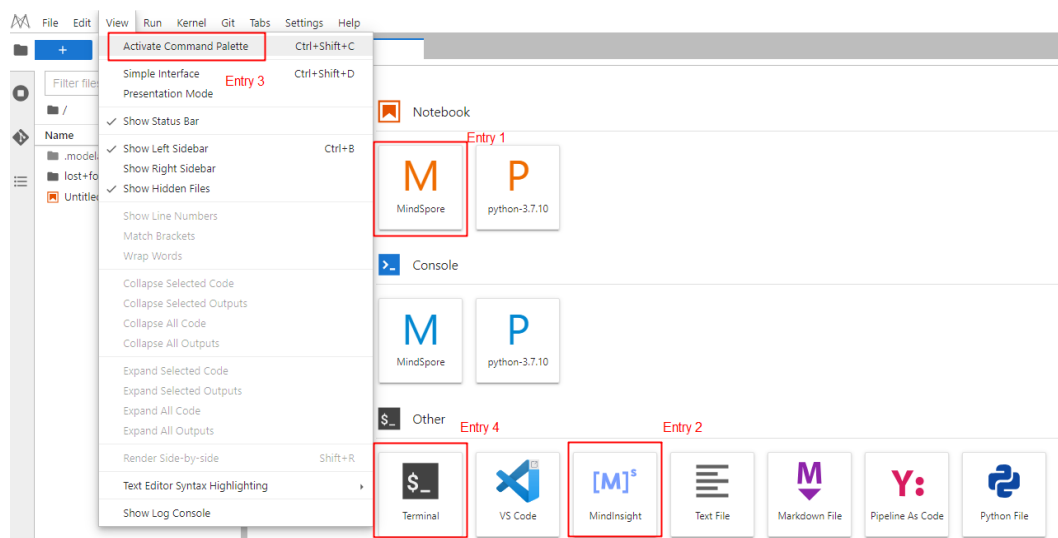
Cargue los datos de resumen en el directorio **/home/ma-user/work/** en un entorno de desarrollo o guárdelos en un sistema de archivos paralelo de OBS.

- Para obtener más detalles sobre cómo cargar los datos de resumen en **/home/ma-user/work/**, véase [Carga de archivos a JupyterLab](#).
- Para almacenar los datos de resumen en un sistema de archivos paralelo de OBS montado en una instancia de notebook, cargue el archivo de resumen generado durante el entrenamiento del modelo en el sistema de archivos paralelo de OBS y asegúrese de que el sistema de archivos paralelo de OBS y ModelArts estén en la misma región. Cuando se inicia MindInsight en una instancia de notebook, la instancia lee automáticamente los datos de resumen del sistema de archivos paralelo de OBS montado.

Paso 3 Iniciar MindInsight

Elija la forma en la que quiera iniciar MindInsight en JupyterLab.

Figura 4-33 Inicio de MindInsight en el JupyterLab



Método 1



MindSpore

1. Haga clic en **MindSpore** para ir al entorno de desarrollo JupyterLab. Se creará automáticamente un archivo de IPYNB.
2. Escriba el siguiente comando en el cuadro de diálogo:

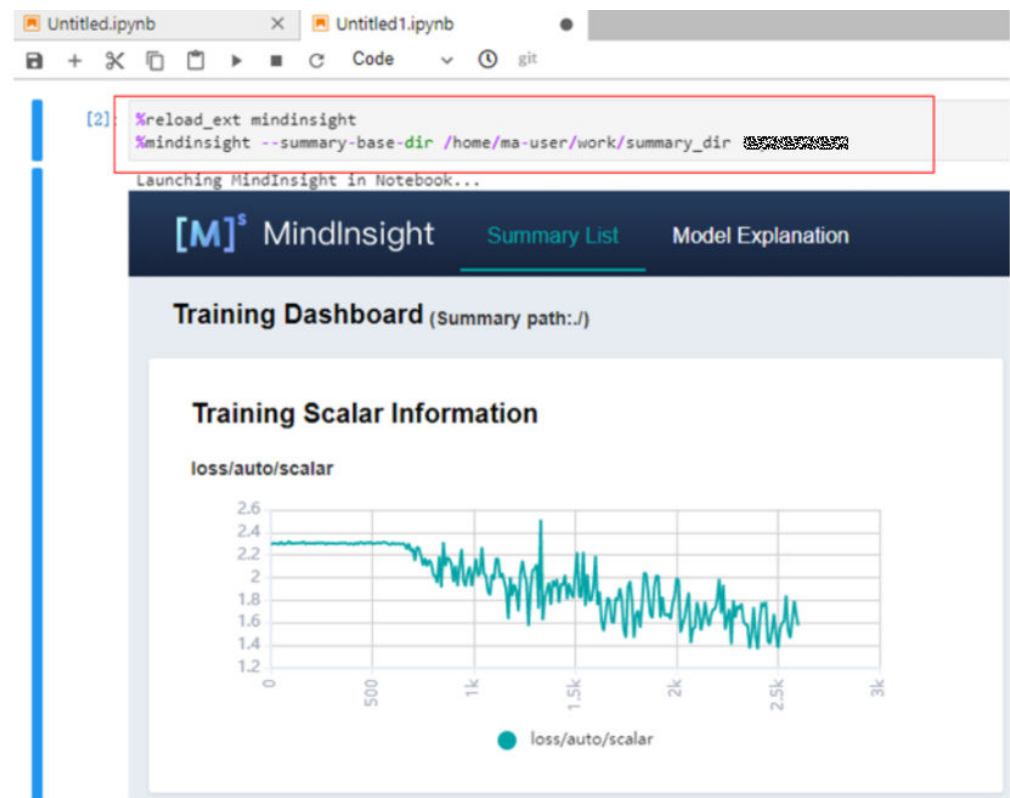
```
%reload_ext mindinsight  
%mindinsight --port {PORT} --summary-base-dir {SUMMARY_BASE_DIR}
```

Parámetros:

- **port {PORT}**: puerto de servicio web para visualización, que por defecto es **8080**. Si se ha utilizado el puerto predeterminado **8080**, especifique un puerto comprendido entre 1 y 65535.
- **summary-base-dir{SUMMARY_BASE_DIR}**: ruta de almacenamiento de datos en el entorno de desarrollo
 - Ruta local al entorno de desarrollo: **./work/xxx** (ruta relativa) o **/home/ma-user/work/xxx** (ruta absoluta)
 - Ruta al bucket del sistema de archivos paralelo de OBS:**obs://xxx/**

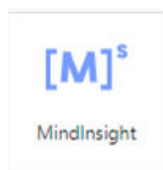
```
For example:  
# If the summary data is stored in /home/ma-user/work/ of a development  
environment, run the following command:  
%mindinsight --summary-base-dir /home/ma-user/work/xxx  
Or  
# If the summary data is stored in an OBS parallel file system, run the  
following command. Then, the development environment will automatically mount  
the storage path to the OBS parallel file system and read data from the path.  
%mindinsight --summary-base-dir obs://xxx/
```

Figura 4-34 Página de MindInsight (1)



Método 2

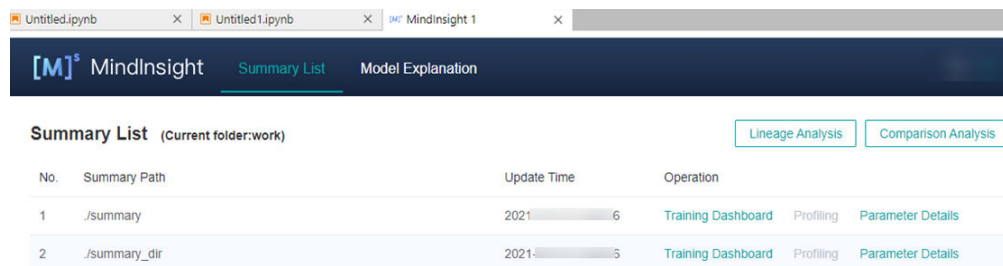


Haga clic en  para ir a la página de MindInsight.

Los datos se leen desde **/home/ma-user/work/** de forma predeterminada.

Si hay dos o más proyectos, seleccione el proyecto de destino para ver sus registros.

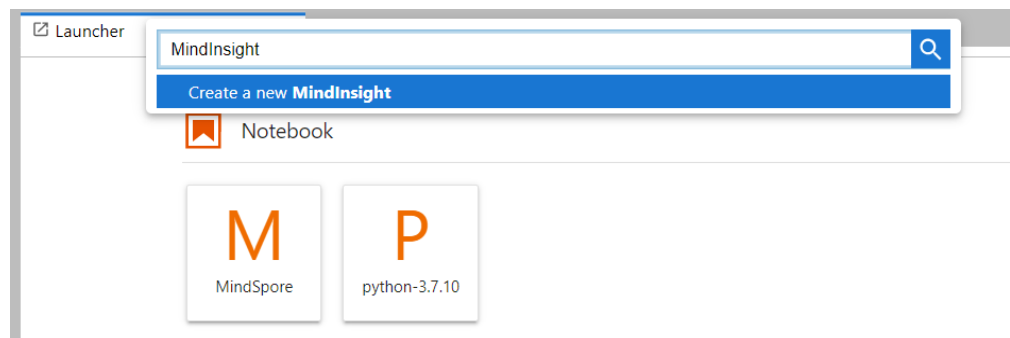
Figura 4-35 Página de MindInsight (2)



Método 3

1. Elija **View > Activate Command Palette**, escriba **MindInsight** en el cuadro de búsqueda y haga clic en **Create a new MindInsight**.

Figura 4-36 Crear un nuevo MindInsight



2. Ingrese la ruta a los datos de resumen o la ruta de almacenamiento al sistema de archivos paralelo de OBS y haga clic en **CREATE**.
 - Ruta local al entorno de desarrollo: **./summary** (ruta relativa) o **/home/ma-user/work/summary** (ruta absoluta)
 - Ruta al sistema de archivos paralelo del OBS: **obs://xxx/**

Figura 4-37 Ruta de acceso a los datos de resumen

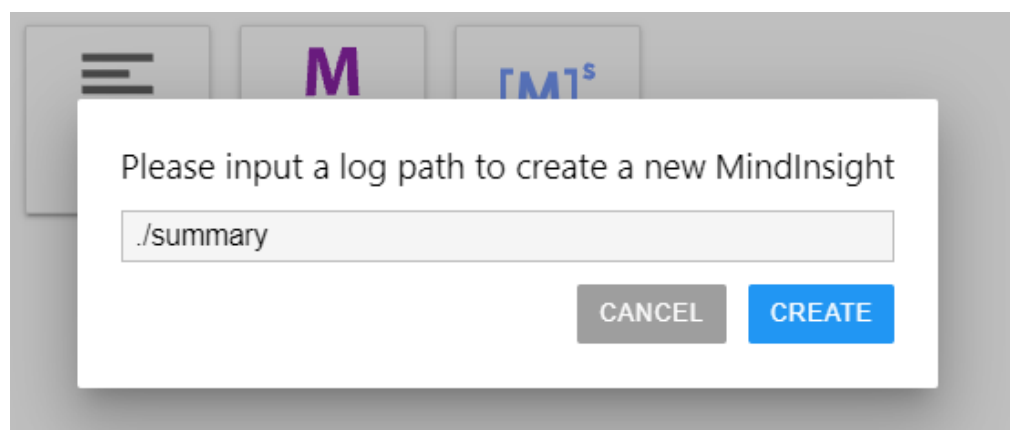
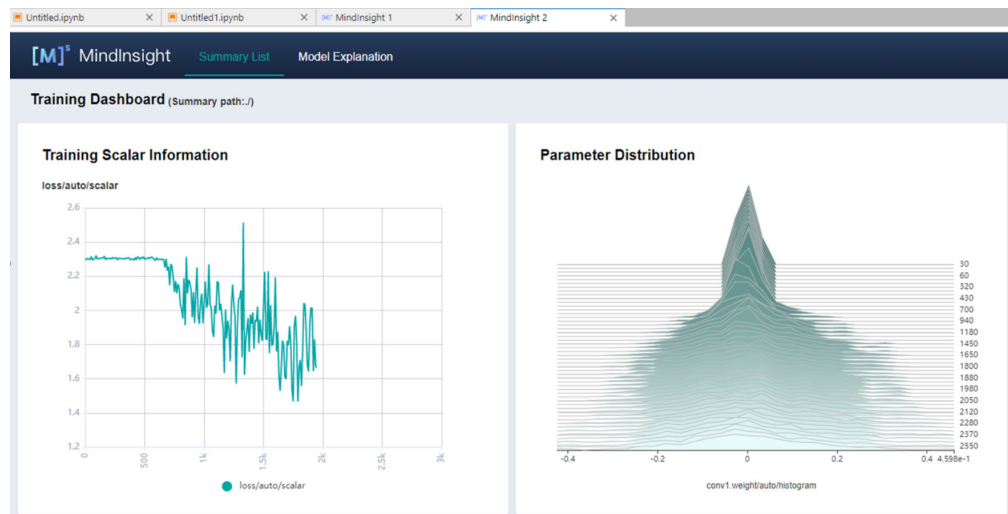


Figura 4-38 Página de MindInsight (3)



NOTA

Se puede iniciar un máximo de 10 instancias de MindInsight mediante el método 2 o 3.

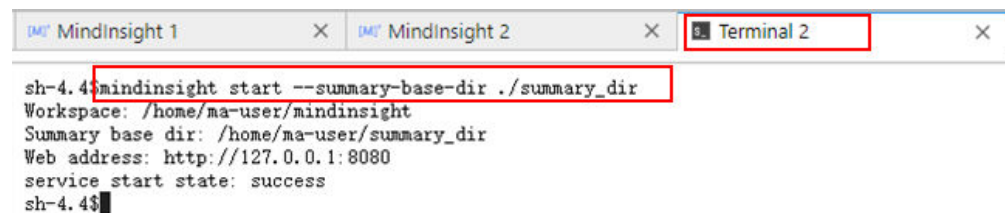
Método 4



Haga clic en **Terminal** y ejecute el siguiente comando (no se mostrará la interfaz de usuario):

```
mindinsight start --summary-base-dir ./summary_dir
```

Figura 4-39 Inicio de MindInsight por Terminal



Paso 4 Ver datos visualizados en el panel de entrenamiento

El panel de entrenamiento es importante para la visualización de MindInsight. Permite la visualización de escalares, distribución de parámetros, gráficos computacionales, gráficos de conjuntos de datos, imágenes y tensores.

Para obtener más información, véase [Consulta del panel de entrenamiento](#) en el sitio web oficial del MindSpore.

Operaciones relacionadas

Para detener una instancia de MindInsight, utilice uno de los siguientes métodos:

- Método 1: Ingrese el siguiente comando en la ventana del archivo **.ipynb** de JupyterLab en la que se configura el número de puerto en **Iniciar MindInsight (8080 por defecto)**:

```
!mindinsight stop --port 8080
```


- Método 2: Haga clic en . Se muestra la página de gestión de instancias de MindInsight, que muestra todas las instancias de MindInsight iniciadas. Haga clic en **SHUT DOWN** junto a la instancia de destino para detenerla.

Figura 4-40 Detención de una instancia




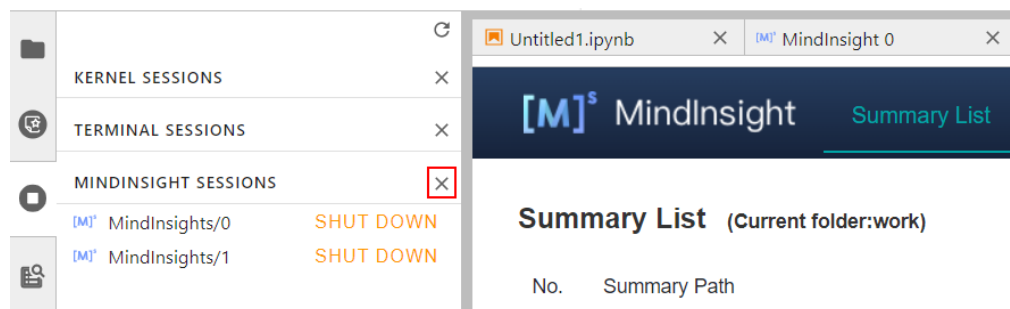
- Método 3: Haga clic en  en la siguiente figura para cerrar todas las instancias de MindInsight iniciadas.

Figura 4-41 Detener todas las instancias de MindInsight iniciadas



- Método 4 (no recomendado): Cierre la ventana de MindInsight en JupyterLab. De esta manera, solo se cierra la ventana de visualización, pero la instancia sigue ejecutándose en el backend.

4.6.3 Trabajos de visualización de TensorBoard

ModelArts es compatible con TensorBoard para visualizar los trabajos de entrenamiento. TensorBoard es un paquete de las herramientas de visualización de TensorFlow. Proporciona las funciones y herramientas de visualización necesarias para los experimentos de aprendizaje automático.

TensorBoard muestra eficazmente el gráfico computacional de TensorFlow en el proceso de ejecución, la tendencia de todas las métricas en el tiempo y los datos utilizados en el entrenamiento.

Requisitos previos

Cuando escriba un script de entrenamiento, agregue el código para recopilar el registro de resumen al script para asegurarse de que el archivo de resumen se genera en el resultado de entrenamiento.

Para obtener detalles sobre cómo agregar el código para recopilar el registro de resumen a un script de entrenamiento impulsado por TensorFlow, consulte el [sitio web oficial de TensorFlow](#).

Nota

- Un trabajo de visualización en ejecución no se factura por separado. Cuando se detiene la instancia del notebook de destino, la facturación se detiene.
- Si el archivo de resumen se almacena en OBS, se le cobrará por el almacenamiento. Después de completar un trabajo, detenga la instancia de notebook y borre los datos de OBS para detener la facturación.

Proceso de creación de un trabajo de visualización de TensorBoard en un entorno de desarrollo

[Paso 1 Crear un entorno de desarrollo y acceda a él en línea](#)

[Paso 2 Subir los datos de resumen](#)

[Paso 3 Iniciar TensorBoard](#)

[Paso 4 Ver datos visualizados en el panel de entrenamiento](#)

Paso 1 Crear un entorno de desarrollo y acceda a él en línea

En la consola de gestión de ModelArts, seleccione **DevEnviron** > **Notebook** y cree una instancia con una imagen de TensorFlow o PyTorch. Una vez creada la instancia, haga clic en **Open** en la columna **Operation** de la instancia para acceder a ella en línea.

Solo las variantes de CPU y GPU con TensorFlow2.1, PyTorch1.4 o PyTorch1.8 y posteriores pueden admitir la visualización de TensorBoard para trabajos de entrenamiento. Seleccione las imágenes y variantes según los requisitos del sitio.

Paso 2 Subir los datos de resumen

Los datos de resumen son necesarios para utilizar las funciones de visualización de TensorBoard en DevEnviron.

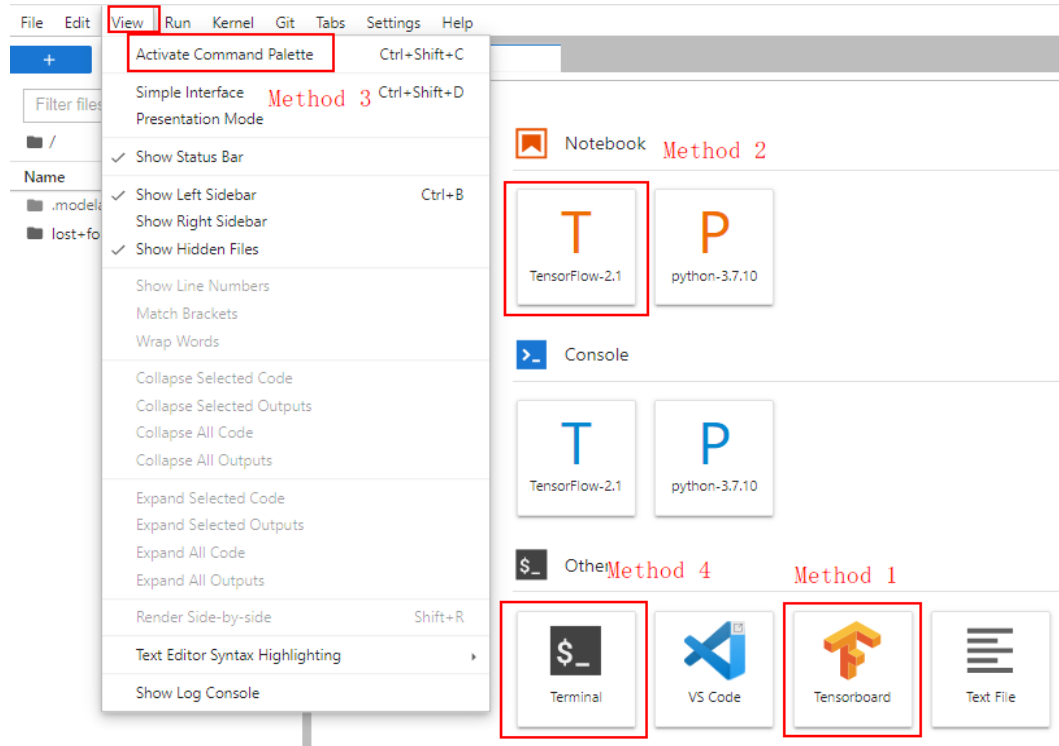
Puede cargar los datos de resumen en el directorio **/home/ma-user/work/** en el entorno de desarrollo o almacenarlos en el sistema de archivos paralelo de OBS.

- Para obtener más información sobre cómo cargar los datos de resumen en la ruta del notebook **/home/ma-user/work/**, consulte la sección [Carga de archivos a JupyterLab](#).
- Para almacenar los datos de resumen en un sistema de archivos paralelo de OBS montado en una instancia de notebook, cargue el archivo de resumen generado durante el entrenamiento del modelo en el sistema de archivos paralelo de OBS y asegúrese de que el sistema de archivos paralelo de OBS y ModelArts estén en la misma región. Cuando se inicia TensorBoard en una instancia de notebook, la instancia de notebook monta automáticamente el directorio del sistema de archivos paralelo de OBS y lee los datos de resumen.

Paso 3 Iniciar TensorBoard

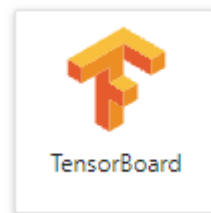
Existen varios métodos para abrir TensorBoard de JupyterLab en el entorno de desarrollo. Seleccione uno en función de sus hábitos.

Figura 4-42 TensorBoard de inicio en el JupyterLab



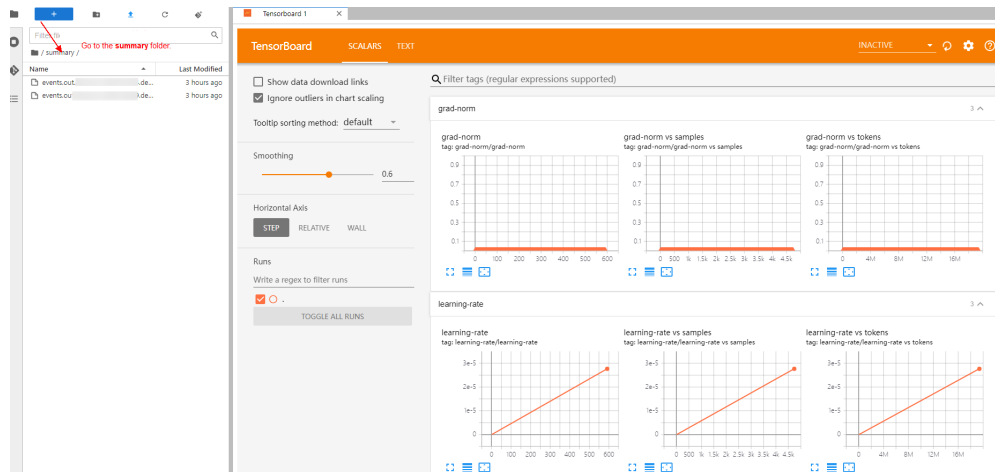
Método 1 (recomendado):

1. Abra JupyterLab en el panel de navegación de la izquierda, cree la carpeta **summary** y cargue datos en **/home/ma-user/work/summary**. El nombre de la carpeta debe ser **summary**.



2. Vaya a la carpeta **summary** y haga clic en **TensorBoard**. Consulte [Figura 4-43](#) para ir a la página de

Figura 4-43 Página de TensorBoard (1)




Método 2

AVISO

Puede actualizar TensorBoard a cualquier versión excepto 2.4.0. Después de la actualización, la nueva versión de TensorBoard solo se utiliza en el método 2. Para otros métodos, utilice TensorBoard 2.1.1.



- Haga clic en  para ir al entorno de desarrollo JupyterLab. El archivo `.ipynb` se crea automáticamente.
- Escriba el siguiente comando en el cuadro de diálogo:

```
%reload_ext ma_tensorboard
%ma_tensorboard --port {PORT} --logdir {BASE_DIR}
```

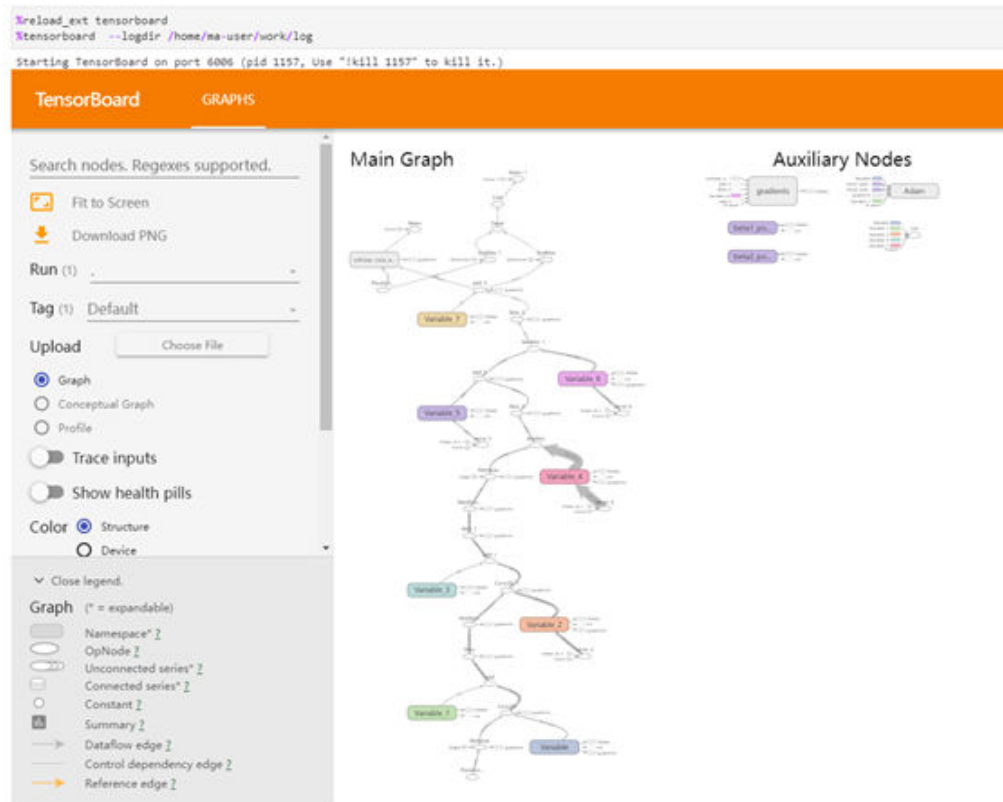
Parámetros:

- **port** *{PORT}*: puerto de servicio web para visualización, que por defecto es **8080**. Si se ha utilizado el puerto predeterminado **8080**, especifique un puerto comprendido entre 1 y 65535.
- **logdir** *{BASE_DIR}*: ruta de almacenamiento de datos en el entorno de desarrollo
 - Ruta local del entorno de desarrollo: **./work/xxx** (ruta relativa) o **/home/ma-user/work/xxx** (ruta absoluta)
 - Ruta del sistema de archivos paralelos de OBS **obs://xxx/**

Example:

```
# If the summary data is stored in /home/ma-user/work/ of the development
environment, run the following command:
%ma_tensorboard --port {PORT} --logdir /home/ma-user/work/xxx
or
# If the summary data is stored in the OBS parallel file system, run the
following command and the development environment automatically mounts the
storage path of the OBS parallel file system and reads data.
%ma_tensorboard --port {PORT} --logdir obs://xxx/
```

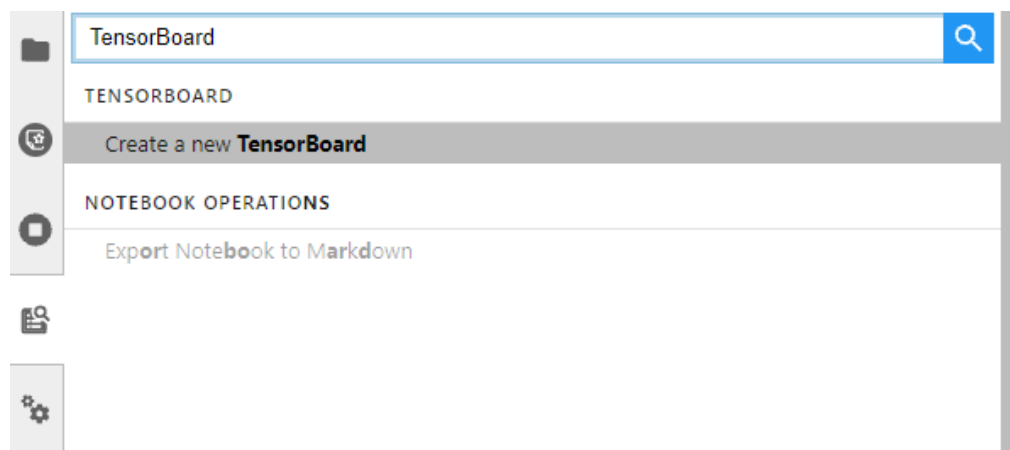
Figura 4-44 Página de TensorBoard (2)



Método 3

1. Elija **View > Activate Command Palette**, escriba **TensorBoard** en el cuadro de búsqueda y haga clic en **Create a new TensorBoard**.

Figura 4-45 Crear un nuevo TensorBoard



2. Introduzca la ruta de acceso de los datos de resumen que desea ver o la ruta de acceso de almacenamiento del sistema de archivos paralelo de OBS.
 - Ruta local del entorno de desarrollo: **./summary** (ruta relativa) o **/home/ma-user/work/summary** (ruta absoluta)

- Ruta del bucket del sistema de archivos paralelo de OBS: **obs://xxx/**

Figura 4-46 Introducción de la ruta de datos de resumen

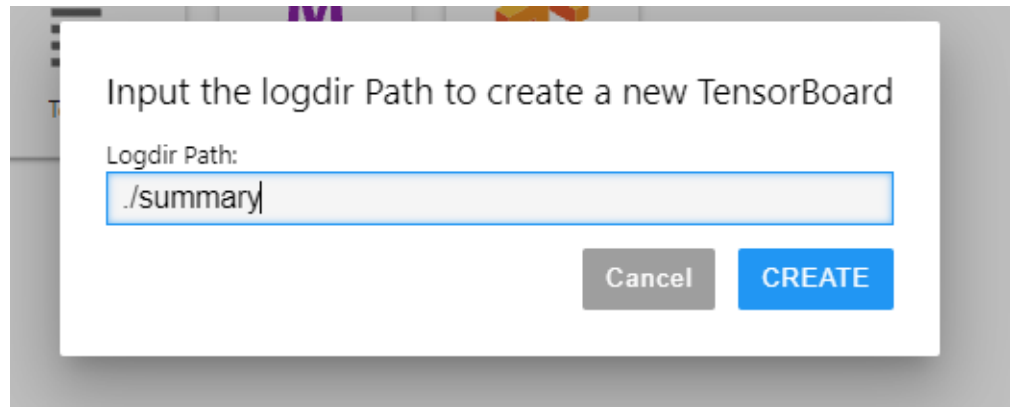
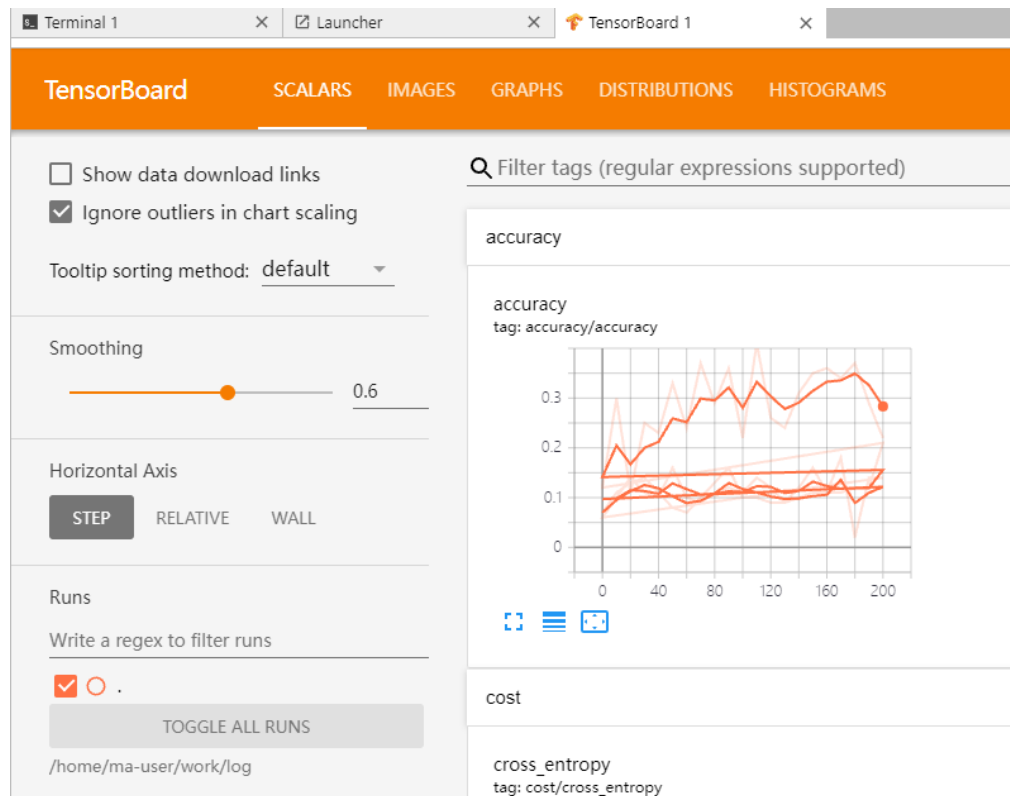


Figura 4-47 Página de TensorBoard (3)



Método 4



Terminal

Haga clic en **Terminal** y ejecute el siguiente comando. No se mostrará la interfaz de usuario.

```
tensorboard --logdir ./log
```

Figura 4-48 Inicio de TensorBoard por Terminal

```
sh-4.4@ped  
/home/aa:user  
sh-4.4$tensorboard --logdir ./log  
2021-10-18 20:34:53.506976: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libwinfer.so.6  
2021-10-18 20:34:53.589272: I tensorflow/stream_executor/platform/default/dso_loader.cc:44] Successfully opened dynamic library libwinfer_plugin.so.6  
Serving TensorBoard on localhost; to expose to the network, use a proxy or pass --bind_all  
TensorBoard 2.1.1 at http://localhost:6006/ (Press CTRL+C to quit)
```

Paso 4 Ver datos visualizados en el panel de entrenamiento

Para visualizar TensorBoard, necesita el panel de entrenamiento. Permite visualizar escalares, imágenes y gráficos computacionales.

Para obtener más funciones, consulte los [Pasos iniciales con TensorBoard](#).

Operaciones relacionadas

Para detener una instancia de TensorBoard, utilice uno de los métodos siguientes:


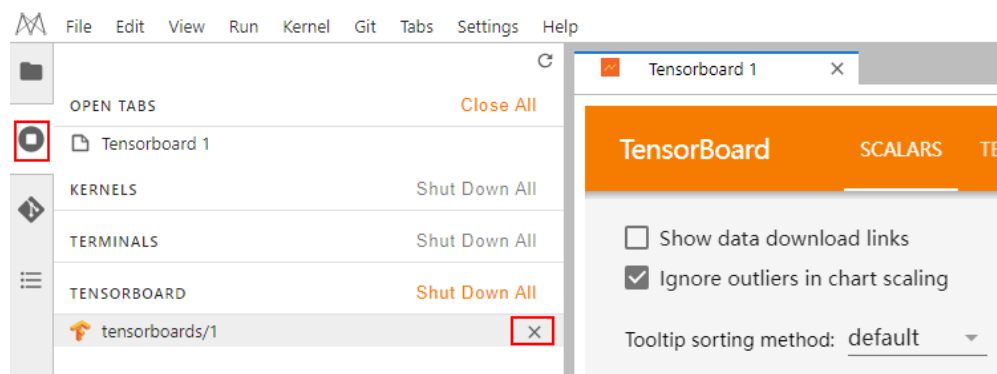
- Método 1: Haga clic en . Se muestra la página de gestión de instancias de TensorBoard, que muestra todas las instancias de TensorBoard iniciadas. Haga clic en **SHUT DOWN** junto a una instancia.

Figura 4-49 Haga clic en SHUT DOWN para detener una instancia



- Método 2: Ingrese el siguiente comando en la ventana del archivo .ipynb de JupyterLab (Obtenga PID en la pantalla de inicio o con el comando `ps -ef | grep tensorboard:`
`!kill PID`


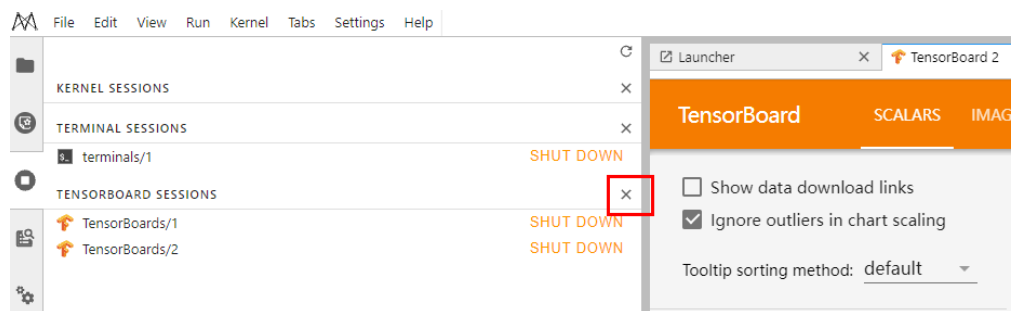
- Método 3: Haga clic en  como se muestra en la siguiente figura para detener todas las instancias de TensorBoard iniciadas.

Figura 4-50 Detener todas las instancias de TensorBoard iniciadas



- (No recomendado) Método 4: Cerrar la ventana de TensorBoard en JupyterLab. Este método cierra solo la ventana, pero la instancia sigue ejecutándose en el backend.

4.7 Carga y descarga de datos en notebook

4.7.1 Carga de archivos a JupyterLab

4.7.1.1 Escenarios

La carga de archivos fácil y rápida es un requisito común en el desarrollo de IA.

Antes de la optimización, ModelArts solo permitía cargar los archivos locales que no superaran los 100 MB directamente en una instancia de notebook. Sin embargo, los archivos que se van a cargar no son todos almacenados localmente, que puede ser de un repositorio de código abierto de GitHub, un conjunto de datos de código abierto (<https://nodejs.org/dist/v12.4.0/node-v12.4.0-linux-x64.tar.xz>) u OBS. Además, ModelArts no mostró el progreso ni la velocidad de carga del archivo.

ModelArts se ha optimizado para una mejor experiencia de carga de archivos. No solo proporciona más funciones de carga de archivos, sino que también muestra más detalles de carga de archivos.

Carga de archivos optimizada:

- Soporta los archivos locales.
- Soporta la clonación de archivos desde repositorios de código abierto de GitHub.
- Soporta los archivos de OBS.
- Soporta los archivos remotos.
- Soporta el progreso de carga visualizado.

4.7.1.2 Carga de archivos desde una ruta local a JupyterLab

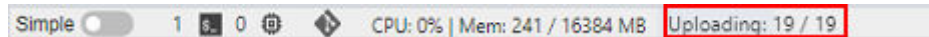
4.7.1.2.1 Carga de escenarios y de entradas

JupyterLab proporciona varios métodos para cargar archivos.

Métodos para cargar un archivo

- Para un archivo que no supere los 100 MB, cárguelo directamente y se muestran detalles como el tamaño del archivo, el progreso de la carga y la velocidad de carga.
- Para un archivo que supere los 100 MB pero no supere los 5 GB, suba el archivo a OBS (un bucket de objetos o un sistema de archivos paralelo) y, a continuación, descargue el archivo de OBS a una instancia de notebook. Una vez completada la descarga, el archivo se elimina de OBS.
- Para un archivo que supere los 5 GB, súbalo invocando al SDK de ModelArts o a MoXing.
- Para un archivo que comparta el mismo nombre con un archivo existente en el directorio actual de una instancia de notebook, sobrescriba el archivo existente o cancele la carga.

- Se puede cargar un máximo de 10 archivos a la vez. Los otros archivos están en espera de carga. No se pueden cargar carpetas. Si se requiere una carpeta, comprima en un paquete, cargue el paquete en el notebook y descomprima el paquete en Terminal.
`unzip xxx.zip # Directly decompress the package in the path where the package is stored.`
- Cuando se cargan varios archivos por lotes, el número total de archivos que se van a cargar y el número de archivos que se han cargado se muestran en la parte inferior de la ventana de JupyterLab.

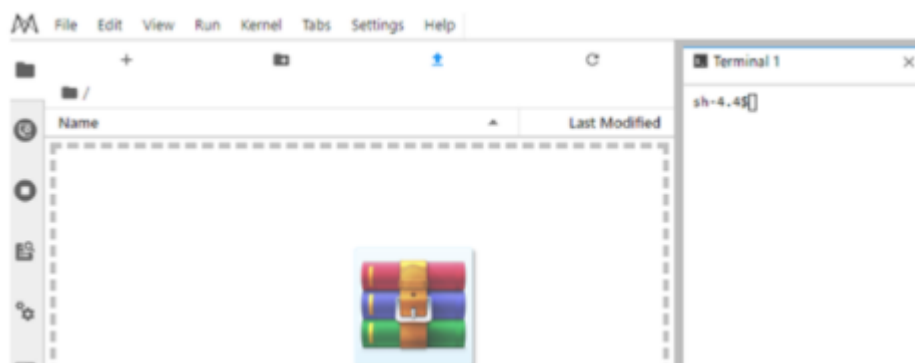


Requisitos previos

Ha utilizado JupyterLab para abrir un entorno de notebook en ejecución.

Entrada de carga 1: Arrastrar un archivo a la ventana del explorador de archivos

Arrastre el archivo al área en blanco a la izquierda de la ventana de JupyterLab y cárguelo.



Entrada de carga 2: hacer clic en el ícono de carga de archivo y cargar un archivo


Haga clic en  en la barra de navegación situada en la parte superior de la ventana. En el cuadro de diálogo que aparece, arrastre o seleccione un archivo local y cárguelo.

Figura 4-51 Ícono de carga de archivo

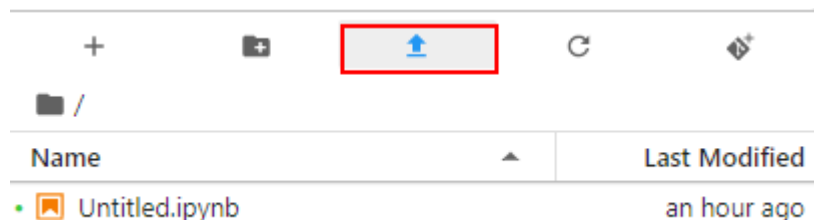
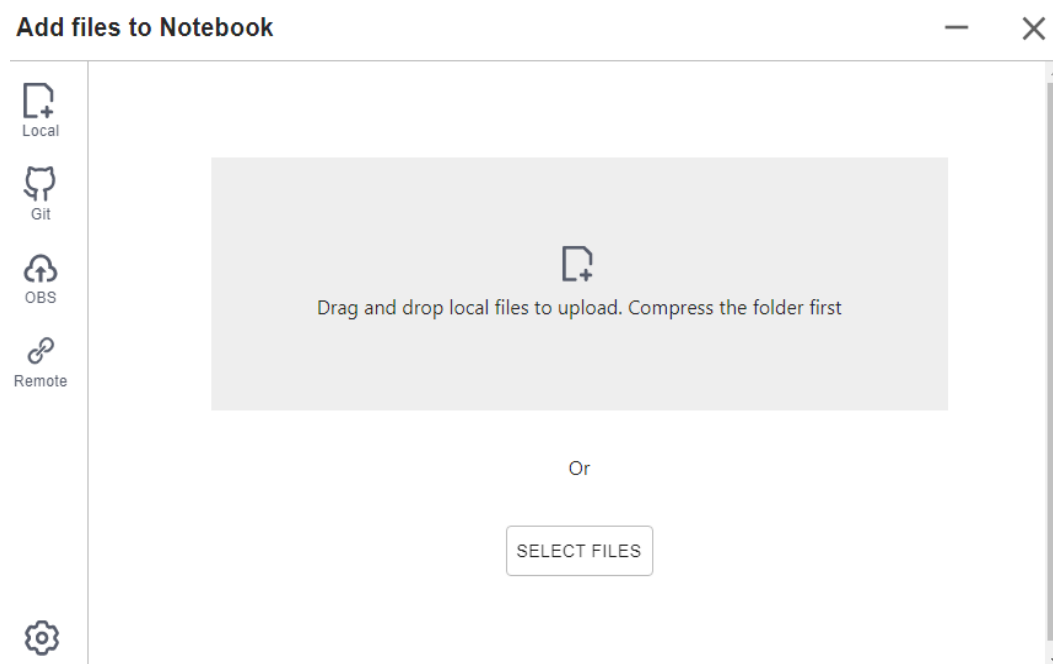


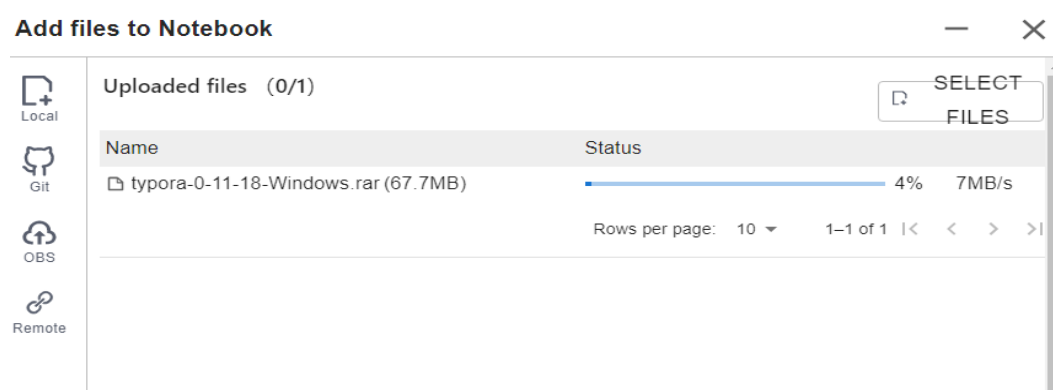
Figura 4-52 Página de carga de archivos



4.7.1.2.2 Carga de un archivo local menos de 100 MB a JupyterLab

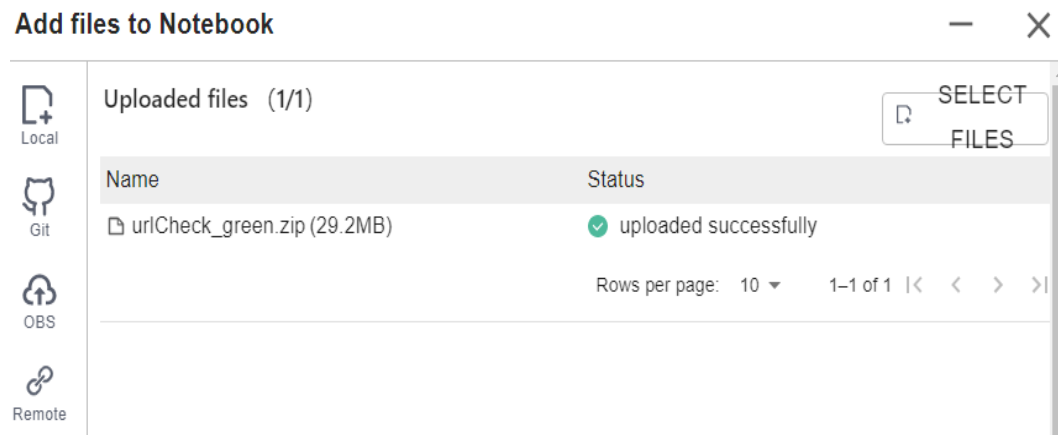
Para un archivo que no exceda los 100 MB, cárguelo directamente en la instancia del notebook de destino. Se muestra la información detallada, como el tamaño del archivo, el progreso de la carga y la velocidad de carga.

Figura 4-53 Subir un archivo de menos de 100 MB



Se muestra un mensaje después de cargar el archivo.

Figura 4-54 Fecha y hora de carga

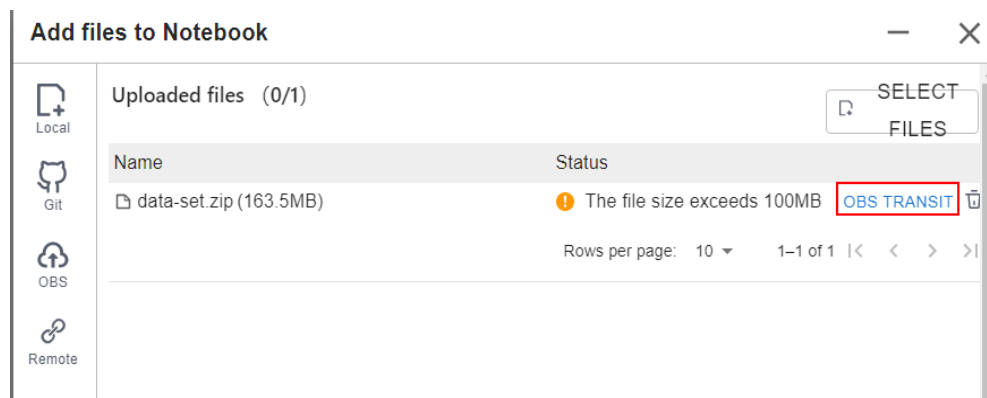


4.7.1.2.3 Carga de un archivo local con un tamaño que varía de 100 MB a 5 GB a JupyterLab

Para un archivo que supere los 100 MB pero no exceda los 5 GB, cargue el archivo en OBS (un bucket de objetos o un sistema de archivos paralelo) y, a continuación, descargue el archivo de OBS a la instancia de notebook de destino. Una vez completada la descarga, el archivo se elimina automáticamente de OBS.

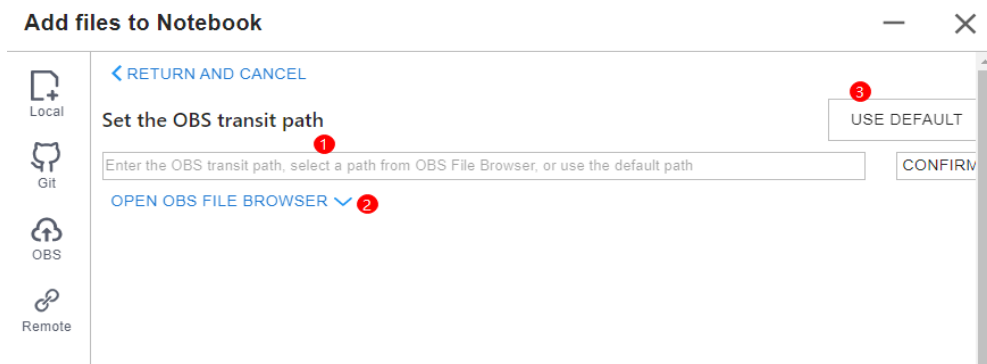
Por ejemplo, en el escenario que se muestra en la siguiente figura, cargue el archivo a través de OBS.

Figura 4-55 Subir un archivo grande a través de OBS




Para cargar un archivo grande a través de OBS, establezca una ruta de acceso de OBS.

Figura 4-56 Subir un archivo a través de OBS



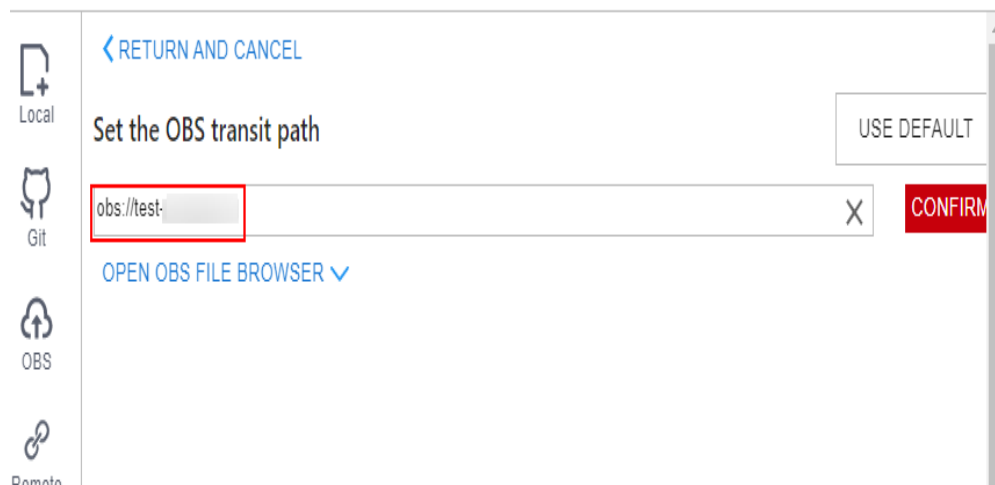
NOTA

Establezca una ruta de OBS para cargar los archivos locales en JupyterLab. Después de la configuración, esta ruta se utiliza de forma predeterminada en las operaciones de seguimiento. Para cambiar la ruta,

haga clic en  en la esquina inferior izquierda de la ventana de carga de archivos.

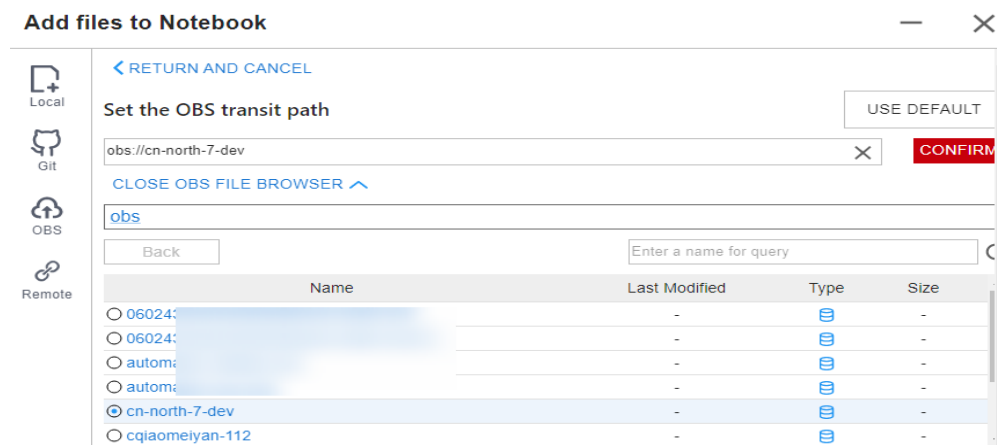
- Método 1: Ingrese una ruta de OBS válida en el cuadro de texto y haga clic en **OK**.

Figura 4-57 Configuración de una ruta de OBS



- Método 2: Seleccione una ruta de OBS de **OBS File Browser** y haga clic en **OK**.

Figura 4-58 Navegador de archivos de OBS



- Método 3: Utilice la ruta predeterminada.

Figura 4-59 Usar la ruta de acceso predeterminada para cargar un archivo

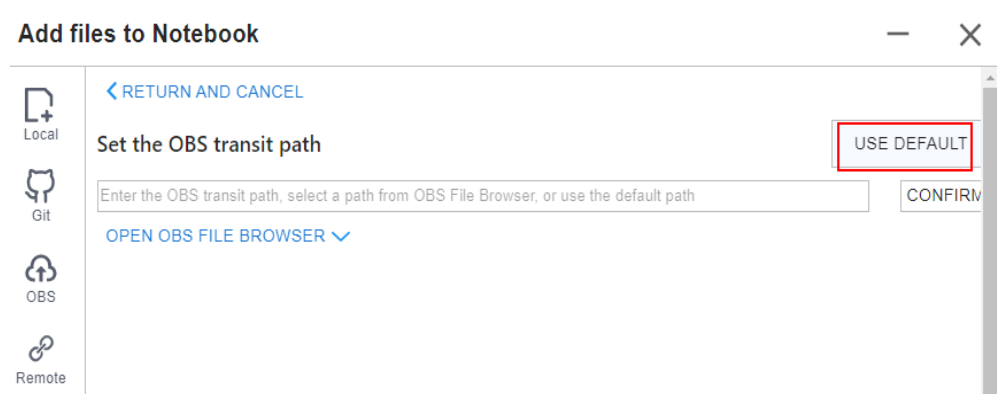
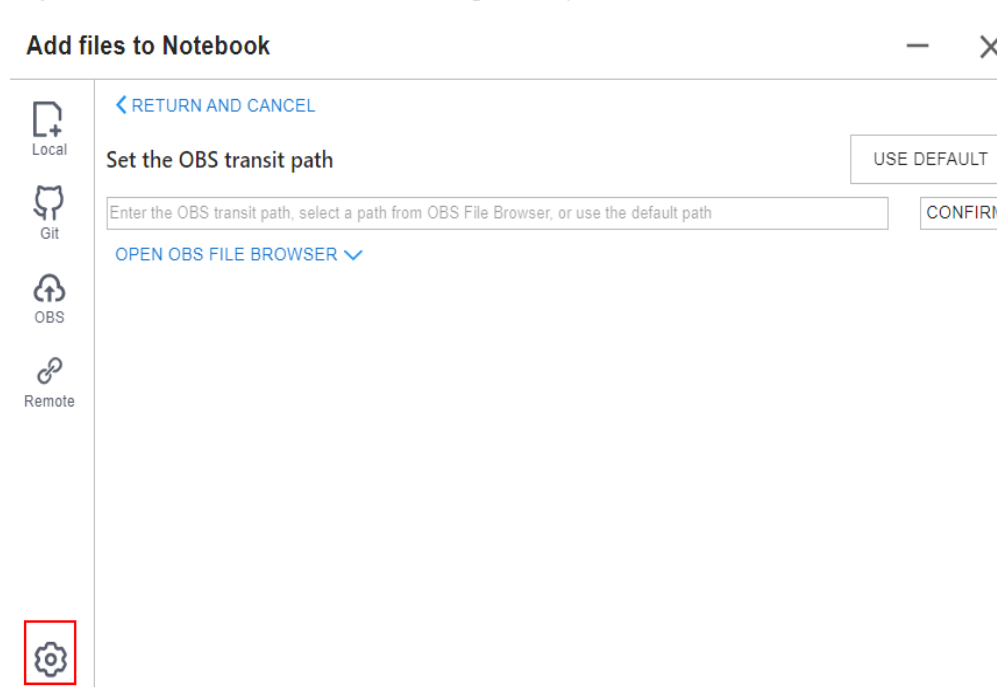
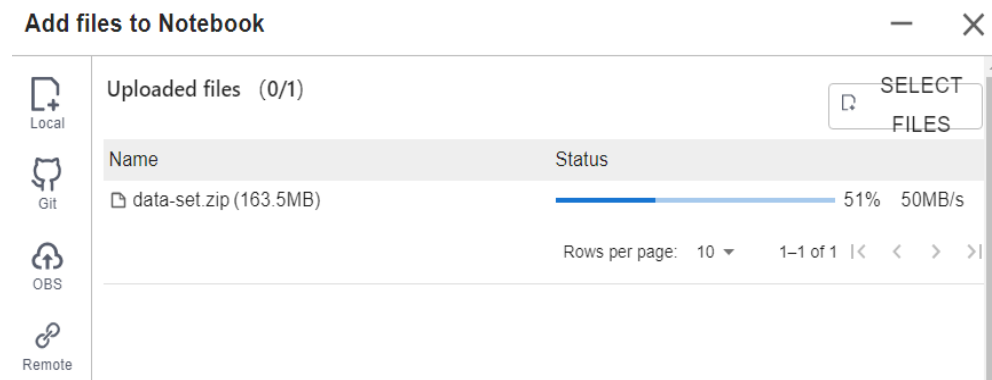


Figura 4-60 Establecer una ruta de OBS para cargar un archivo local



Después de establecer la ruta de OBS, cargue un archivo.

Figura 4-61 Carga de un archivo



Descompresión de un paquete

Después de cargar un archivo grande en Notebook JupyterLab como un paquete comprimido, puede descomprimir el paquete en Terminal.

```
unzip xxx.zip # Directly decompress the package in the path where the package is stored.
```

Para obtener más detalles, busque el comando de descompresión en los motores de búsqueda principales.

4.7.1.2.4 Carga de un archivo local más de 5 GB a JupyterLab

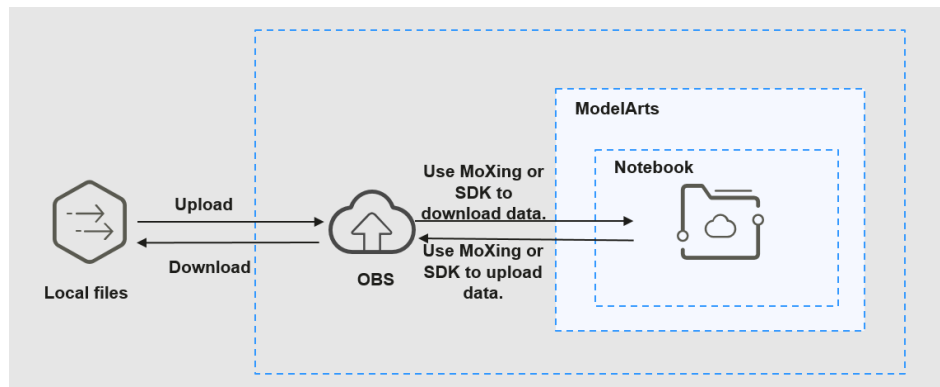
Un archivo superior a 5 GB no se puede cargar directamente en JupyterLab.

Figura 4-62 Error al cargar un archivo de más de 5 GB



Para cargar los archivos de más de 5 GB, cárguelos en OBS. A continuación, invoque a las API de MoXing o en el SDK en ModelArts en la instancia del notebook de destino para leer y escribir los archivos en OBS.

Figura 4-63 Carga y descarga de archivos de gran tamaño en una instancia de notebook



El procedimiento es el siguiente:

1. Cargue el archivo desde una ruta local a OBS. Para obtener más detalles, véase [Carga de un objeto](#).
2. Descargue el archivo desde OBS a la instancia del notebook invocando a las API en el SDK o de MoXing en ModelArts.
 - Método 1: Invocar al SDK de ModelArts para descargar un archivo de OBS.

Código de ejemplo:

```
from modelarts.session import Session
session = Session()
session.obs.copy("obs://bucket-name/obs_file.txt", "/home/ma-user/work/")
```

- Método 2: Invocar a la API de MoXing de ModelArts para leer un archivo de OBS.

```
import moxing as mox

# Download the OBS folder sub_dir_0 from OBS to a notebook instance.
mox.file.copy_parallel('obs://bucket_name/sub_dir_0', '/home/ma-user/work/sub_dir_0')
# Download the OBS file obs_file.txt from OBS to a notebook instance.
mox.file.copy('obs://bucket_name/obs_file.txt', '/home/ma-user/work/obs_file.txt')
```

Si se descarga un archivo .zip, ejecute el comando siguiente en el terminal para descomprimir el paquete:

```
unzip xxx.zip # Directly decompress the package in the path where the package is stored.
```

Una vez ejecutado el código, abra el terminal mostrado en [Figura 4-64](#) y ejecute el comando `ls /home/ma-user/work` para ver el archivo descargado en la instancia de notebook. También puede ver el archivo descargado en el panel de navegación izquierdo de Jupyter. Si no se muestra el archivo, actualice la página.

Figura 4-64 Apertura del terminal

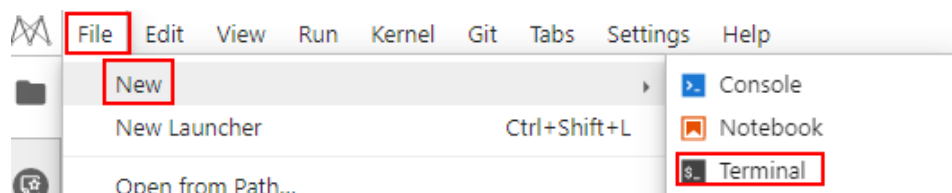
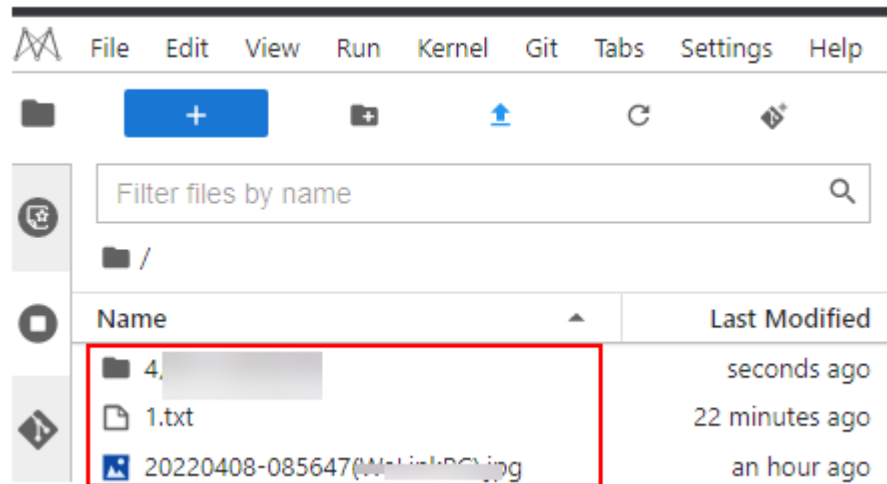


Figura 4-65 Archivo descargado a una instancia de notebook



Manejo de errores


Si descarga un archivo de OBS a la instancia de su notebook y el sistema muestra el mensaje de error "Permission denied", realice las siguientes operaciones para solucionar problemas:

- Asegúrese de que el bucket de OBS y la instancia del notebook de destino estén en la misma región. Si el bucket de OBS y la instancia de notebook están en diferentes regiones, se deniega el acceso a OBS.
- Asegúrese de que la cuenta del notebook tenga permiso para leer datos en el bucket de OBS.

Para obtener más detalles, véase [Ruta de OBS incorrecta en ModelArts](#).

4.7.1.3 Clonación de un repositorio de código abierto de GitHub

Los archivos se pueden clonar desde un repositorio de código abierto de GitHub a JupyterLab.

1. Utilice JupyterLab para abrir una instancia de notebook en ejecución.
2. Haga clic en  en la barra de navegación situada en la parte superior de la ventana de




JupyterLab. En el cuadro de diálogo mostrado, haga clic en  a la izquierda para ir a la página de clonación de archivos de un repositorio de código abierto de GitHub.

Figura 4-66 Ícono de carga de archivo

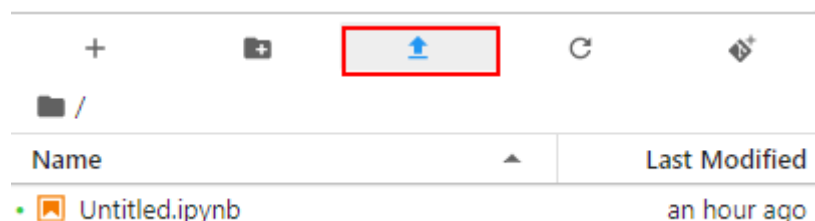
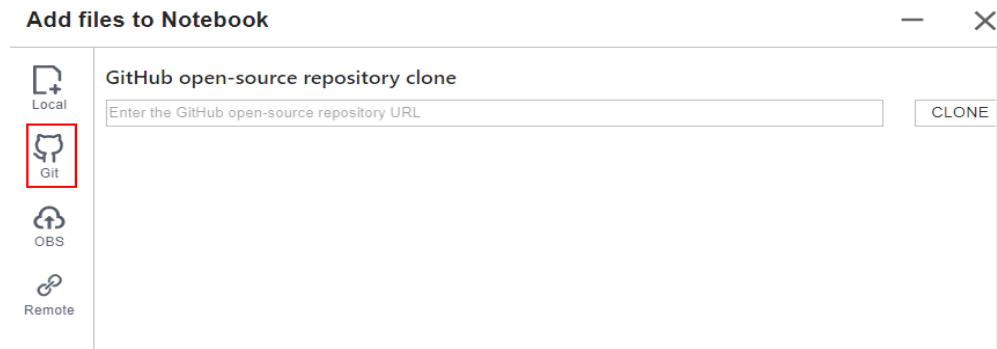


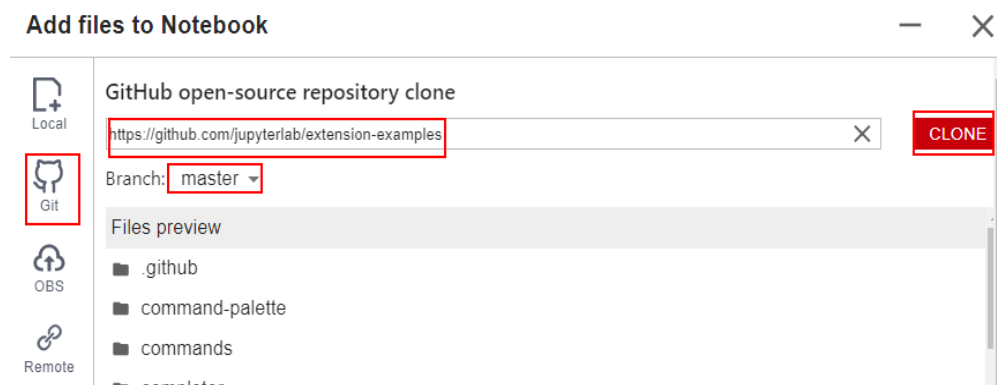
Figura 4-67 Página para clonar archivos desde un repositorio de código abierto de GitHub



3. Ingrese una dirección válida de un repositorio de código abierto de GitHub, seleccione los archivos de los archivos y carpetas mostrados y haga clic en **Clone**.

GitHub dirección de repositorio de código abierto: <https://github.com/jupyterlab/extension-examples>

Figura 4-68 Introducir una dirección válida de un repositorio de código abierto de GitHub



4. Ver el proceso de clonación.

Figura 4-69 Proceso de clonación de un repositorio



extension-examples is being cloned



Begin clone

5. Complete el clon.

Figura 4-70 Repository cloned

上传文件到Notebook



extension-examples 克隆成功

返回

Manejo de errores

- No clonar el repositorio puede deberse a problemas de red. En este caso, ejecute el comando **git clone https://github.com/jupyterlab/extension-examples.git** en la página **Terminal** para probar la conectividad de red.

Figura 4-71 Error al clonar el repositorio


上传文件到Notebook



extension-examples 仓库Clone失败


fatal: unable to access 'https://github.com/jupyterlab/extension-exa...

返回

- Si el repositorio ya existe en el directorio actual de la instancia del notebook, el sistema muestra un mensaje que indica que el nombre del repositorio ya existe. En este caso, puede sobrescribir el repositorio existente o hacer clic en  para cancelar la clonación.

4.7.1.4 Carga de archivos de OBS a JupyterLab

En JupyterLab puede descargar archivos de OBS a una instancia de notebook. Asegúrese de que el archivo no supere los 10 GB. De lo contrario, se producirá un error en la carga.

1. Utilice JupyterLab para abrir una instancia de notebook en ejecución.
2. Haga clic en  en la barra de navegación situada en la parte superior de la ventana de


JupyterLab. En la ventana mostrada, haga clic en  a la izquierda para ir a la página de carga de archivos de OBS.

Figura 4-72 Ícono de carga de archivo

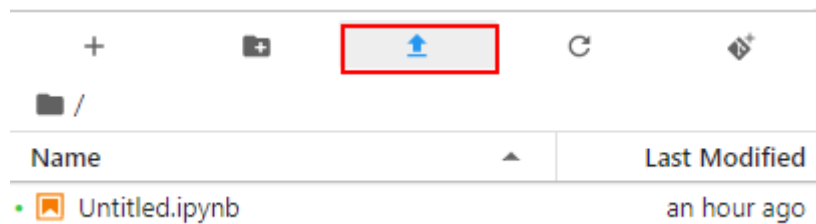
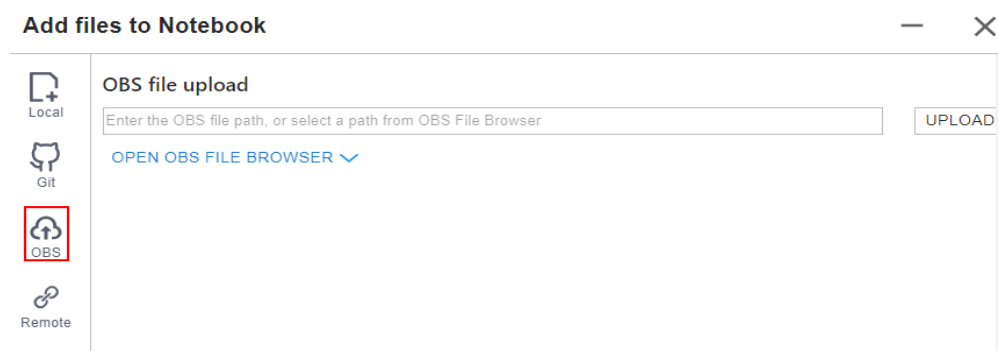
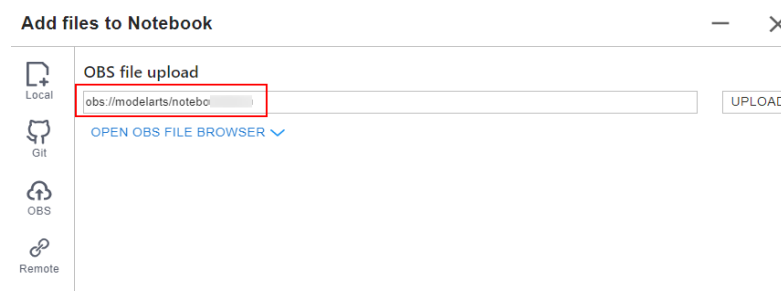


Figura 4-73 Carga de archivos de OBS



3. Establezca una ruta de archivo de OBS de cualquiera de las siguientes maneras:
 - Método 1: Ingrese una ruta de archivo de OBS válida en el cuadro de texto y haga clic en **Upload**.

Figura 4-74 Introducción de una ruta de archivo de OBS válida



NOTA

Ingrese una ruta de archivo de OBS en lugar de una ruta de carpeta. De lo contrario, la carga falla.

- Método 2: Abra **OBS File Browser** y seleccione una ruta de archivo de OBS y haga clic en **Upload**.

Figura 4-75 Carga de un archivo de OBS

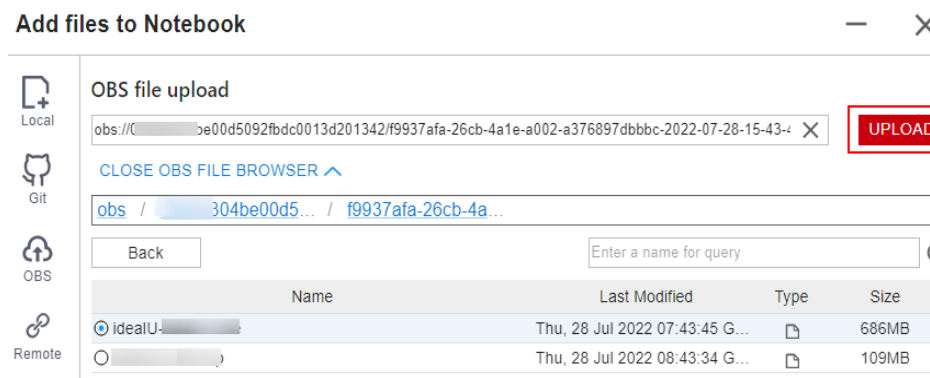
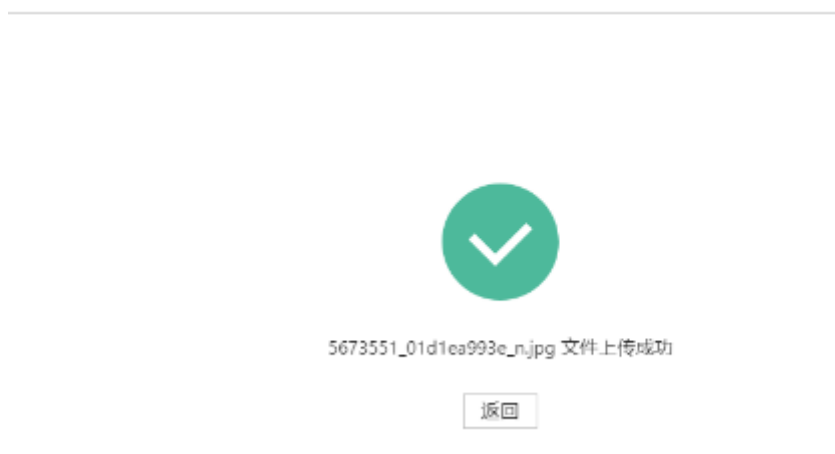


Figura 4-76 File uploaded

上传文件到Notebook

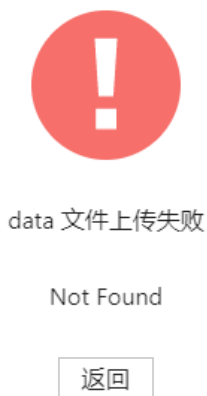


Manejo de errores

Existen tres escenarios típicos en los que no se pudo cargar un archivo.

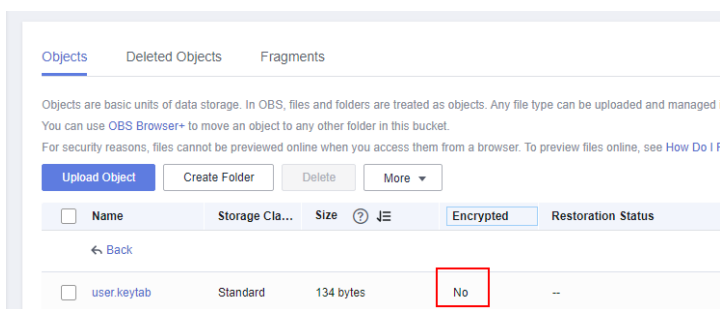
- **Escenario 1**

Figura 4-77 File uploading failure



Causas posibles:

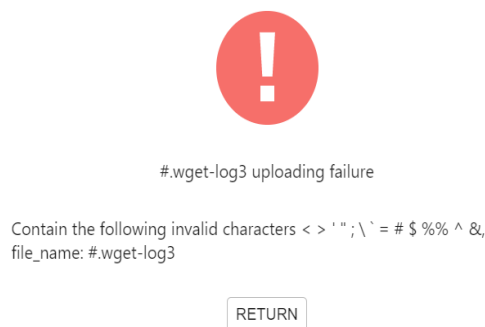
- La ruta de OBS se establece en una carpeta en lugar de en una ruta de archivo.
- El archivo de OBS está encriptado. En este caso, vaya a la consola de OBS y asegúrese de que el archivo esté encriptado.



- El bucket de OBS y la instancia de notebook no están en la misma región. Ensure that the OBS bucket to be read is in the same region as the notebook instance. You cannot access an OBS bucket in another region. Para obtener más detalles, véase [¿Cómo verifico si ModelArts y un bucket de OBS se encuentran en la misma región?](#)
- La cuenta no tiene permiso para acceder al bucket de OBS. En este caso, asegúrese de que la cuenta del notebook tenga el permiso para leer datos en el bucket de OBS. Para obtener más detalles, véase [Comprobar si tiene permiso para acceder al bucket de OBS](#).
- Se eliminó el archivo de OBS. En este caso, asegúrese de que el archivo de OBS que se va a cargar esté disponible.

- **Escenario 2**

Figura 4-78 Error al cargar archivos

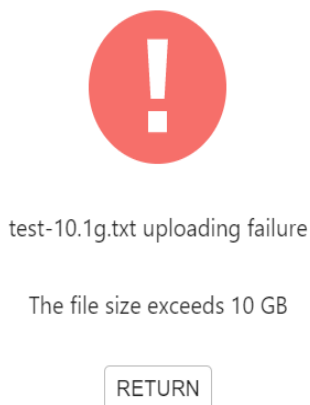


Causas posibles:

El nombre del archivo contiene caracteres especiales como < > " ; \ ` = # \$ % ^ & .

- **Escenario 3**

Figura 4-79 Error al cargar archivos



Causas posibles:

El archivo cargado superó los 10 GB.

4.7.1.5 Carga de archivos remotos a JupyterLab

Los archivos se pueden descargar a JupyterLab a través de las direcciones de archivos remotas.

Método: Introduzca el URL de un archivo remoto en el cuadro de texto de un navegador y el archivo se descargará directamente.



1. Utilice JupyterLab para abrir una instancia de notebook en ejecución.
2. Haga clic en  en la barra de navegación situada en la parte superior de la ventana de JupyterLab. En la ventana mostrada, haga clic en  a la izquierda para ir a la página de carga remota de archivos.

Figura 4-80 Ícono de carga de archivo

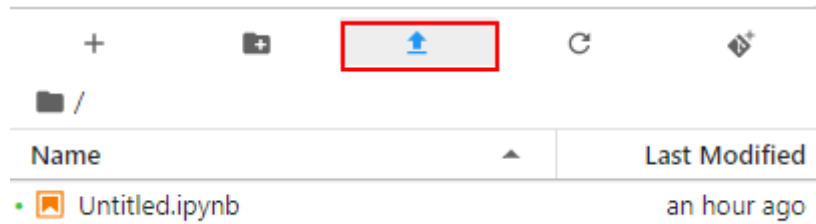
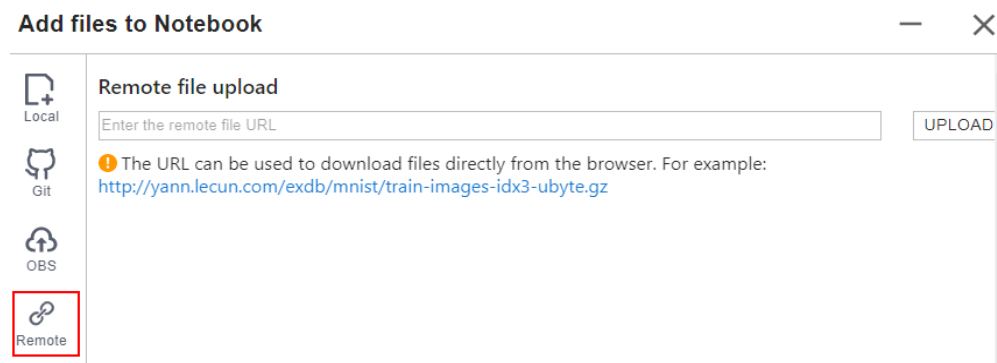


Figura 4-81 Página de carga remota de archivos



3. Introduzca una dirección de URL válida de archivo remoto y el sistema identificará automáticamente el nombre de archivo. Luego, haga clic en **Upload**.

Figura 4-82 Introducir un URL de archivo remoto válido

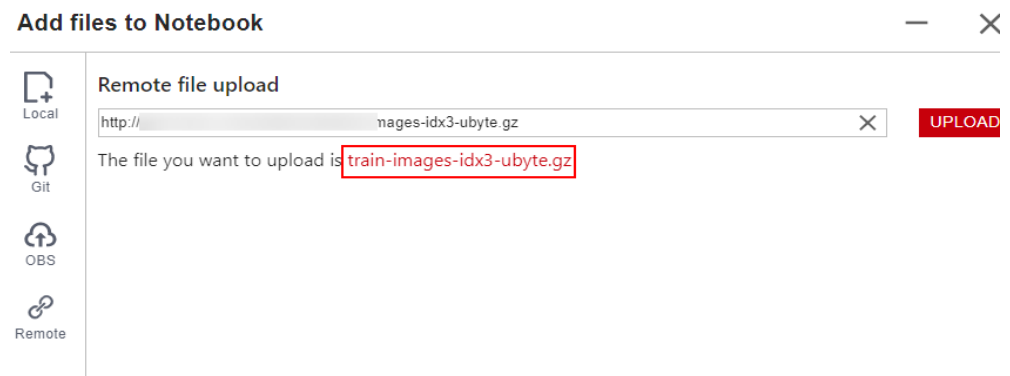


Figura 4-83 Remote file uploaded

上传文件到Notebook



Yunbao-Data-Custom.zip 文件上传成功

返回

Manejo de errores

El error al cargar el archivo remoto puede deberse a problemas de red. En este caso, introduzca el URL del archivo remoto en el cuadro de texto de un navegador para comprobar si el archivo se puede descargar.

Figura 4-84 Failed to upload the remote file

上传文件到Notebook



quick_start.ipynb文件上传失败

Proxy tunneling failed: Service UnavailableUnable to establish SSL co...

返回

4.7.2 Descarga de un archivo de JupyterLab a una ruta local

Los archivos creados en JupyterLab se pueden descargar en una ruta local.

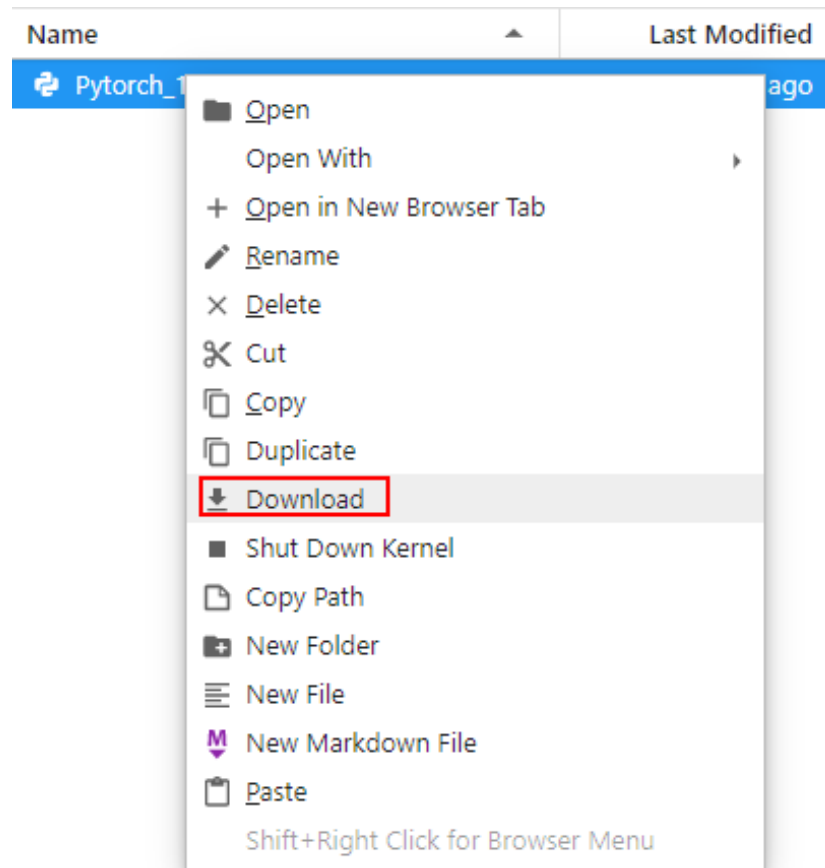
- Si un archivo es inferior o igual a 100 MB, descárguelo directamente desde JupyterLab. Para obtener más detalles, véase [Descarga de un archivo inferior o igual a 100 MB](#).

- Si un archivo tiene más de 100 MB, utilice OBS para transferirlo a su ruta local. Para obtener más detalles, véase [Descarga de un archivo de más de 100 MB](#).

Descarga de un archivo inferior o igual a 100 MB

En la lista de archivos de JupyterLab, haga clic derecho en el archivo que desea descargar y elija **Download** en el menú contextual. El archivo se descarga en la carpeta de descargas de su navegador.

Figura 4-85 Descarga de un archivo



Descarga de un archivo de más de 100 MB

Utilice OBS para transferir el archivo de la instancia del notebook de destino a la ruta local. Para ello, haga lo siguiente:

1. En la instancia de notebook, cree un archivo de IPYNB más de 100 MB y use MoXing para cargarlo en OBS. El código de ejemplo es el siguiente:

```
import moxing as mox
mox.file.copy('/home/ma-user/work/obs_file.txt', 'obs://bucket_name/
obs_file.txt')
```

/home/ma-user/work/obs_file.txt es la ruta de acceso al archivo almacenado en la instancia del notebook. **obs://bucket_name/obs_file.txt** es la ruta del archivo subido a OBS, donde **bucket_name** es el nombre del bucket creado en OBS, y **obs_file.txt** es el archivo subido.

2. Utilice el SDK de OBS o de ModelArts para descargar el archivo de OBS a la ruta local.

- Método 1: Utilice OBS para descargar el archivo.
- Descargue **obs_file.txt** desde OBS a la ruta local. Si se va a descargar una gran cantidad de datos, utilice OBS Browser+ para descargarlos. Para obtener más detalles, véase [Descarga de un objeto](#).
- Método 2: Utilice el SDK de ModelArts para descargar el archivo.
 - i. **Descargue e instale el SDK localmente.**
 - ii. **Autenticación de sesiones.**
 - iii. **Descargue el archivo desde OBS a la ruta local.** El código de ejemplo es el siguiente:

```
from modelarts.session import Session

# Hardcoded or plaintext AK/SK is risky. For security, encrypt your
AK/SK and store them in the configuration file or environment
variables.
# In this example, the AK/SK are stored in environment variables for
identity authentication. Before running this example, set
environment variables HUAWEICLOUD_SDK_AK and HUAWEICLOUD_SDK_SK.
__AK = os.environ["HUAWEICLOUD_SDK_AK"]
__SK = os.environ["HUAWEICLOUD_SDK_SK"]
# Decrypt the password if it is encrypted.
session = Session(access_key=__AK,secret_key=__SK, project_id='***',
region_name='***')

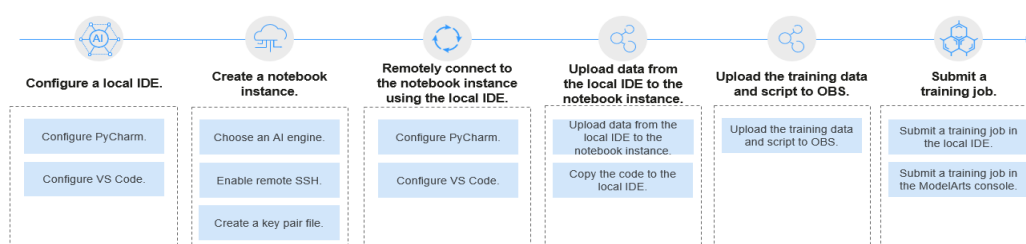
session.download_data(bucket_path="/bucket_name/obs_file.txt",path="/
home/user/obs_file.txt")
```


5 IDE local

5.1 Proceso de operación en un IDE local

ModelArts le permite acceder de forma remota a instancias de notebook desde un IDE local para desarrollar modelos de IA basados en PyTorch, TensorFlow o MindSpore. La siguiente figura muestra el proceso de operación.

Figura 5-1 Uso de un IDE local para desarrollar modelos



1. **Configure un IDE local.**
Configure un IDE local en su PC.
Puede utilizar **PyCharm**, **VS Code** o **herramientas de SSH** para acceder a una instancia de notebook desde un IDE local. PyCharm y VS Code pueden configurarse automáticamente mediante complementos o configurarse manualmente.
2. **Cree una instancia de notebook.**
En la consola de gestión de ModelArts, cree una instancia de notebook con un motor de IA adecuado y SSH remoto habilitado.
3. Utilice el IDE local para acceder a ModelArts DevEnviron de forma remota.
4. **Cargue datos y código en el entorno de desarrollo.**
 - Copie el código en el IDE local, que sincronizará automáticamente el código con el entorno de desarrollo en la nube.
 - Si los datos son menores o iguales a 500 MB, copie directamente los datos en el IDE local.
 - **Al crear un trabajo de entrenamiento**, si el volumen de datos es superior a 500 MB, cargue los datos en OBS y, a continuación, en EVS.

5. Cargue el script de entrenamiento y el conjunto de datos en el directorio de OBS.
6. Envíe un trabajo de entrenamiento.
 - Envíe un trabajo de entrenamiento en el IDE local.
Utilice los SDK de ModelArts. Para obtener más información, véase [Creación de un trabajo de entrenamiento](#).
 - Envíe un trabajo de entrenamiento en la consola de gestión de ModelArts. Para obtener más información, véase [Creación de un trabajo de entrenamiento](#).

5.2 IDE local (PyCharm)

5.2.1 Conexión a una instancia de notebook con PyCharm Toolkit

5.2.1.1 PyCharm Toolkit

Los desarrolladores de IA utilizan herramientas de PyCharm para desarrollar algoritmos o modelos. Por lo tanto, ModelArts proporciona PyCharm Toolkit para ayudar a los desarrolladores de IA a enviar rápidamente código desarrollado localmente a un entorno de entrenamiento en ModelArts. Con PyCharm Toolkit, los desarrolladores pueden cargar rápidamente código, enviar trabajos de entrenamiento y obtener logs de entrenamiento para su visualización local para que puedan centrarse mejor en el desarrollo de código local. Para obtener detalles sobre cómo descargar e instalar PyCharm Toolkit, véase [Instalación con Marketplace](#).

Restricciones

- Actualmente, solo se admite PyCharm 2019.2 o posterior, incluidas las ediciones comunitarias y profesionales.
- Solo se pueden utilizar PyCharm de la edición profesional para acceder al entorno del desarrollo de notebooks.
- Puede utilizar una edición comunitaria o profesional de PyCharm Toolkit para enviar trabajos de entrenamiento. La última versión de PyCharm Toolkit solo se puede utilizar para enviar trabajos de entrenamiento de la nueva versión.
- PyCharm Toolkit soporta PyCharm de la versión de Windows.

Funciones disponibles

Tabla 5-1 Funciones del Toolkit de la última versión

Función	Descripción	Referencia
SSH remoto	Se puede acceder al entorno del desarrollo de notebooks a través de SSH remoto.	Conexión a una instancia de notebook con PyCharm Toolkit

Función	Descripción	Referencia
Model training	El código desarrollado localmente se puede enviar rápidamente a ModelArts y se crea automáticamente un trabajo de entrenamiento de la nueva versión. Durante la ejecución del trabajo de entrenamiento, los logs de entrenamiento se pueden obtener y mostrar en un host local.	<ul style="list-style-type: none"> ● Envío de un trabajo de entrenamiento (Nueva versión) ● Detención de un trabajo de entrenamiento ● Consulta de logs de entrenamiento
OBS-based upload and download	Los archivos o carpetas locales se pueden cargar en OBS y los archivos o carpetas se pueden descargar desde OBS a un directorio local.	Carga de datos en una instancia de notebook con PyCharm

5.2.1.2 Descarga e instalación de PyCharm Toolkit

Antes de utilizar PyCharm Toolkit, instálelo y configúrelo en PyCharm siguiendo las instrucciones proporcionadas en esta sección.

Requisitos previos

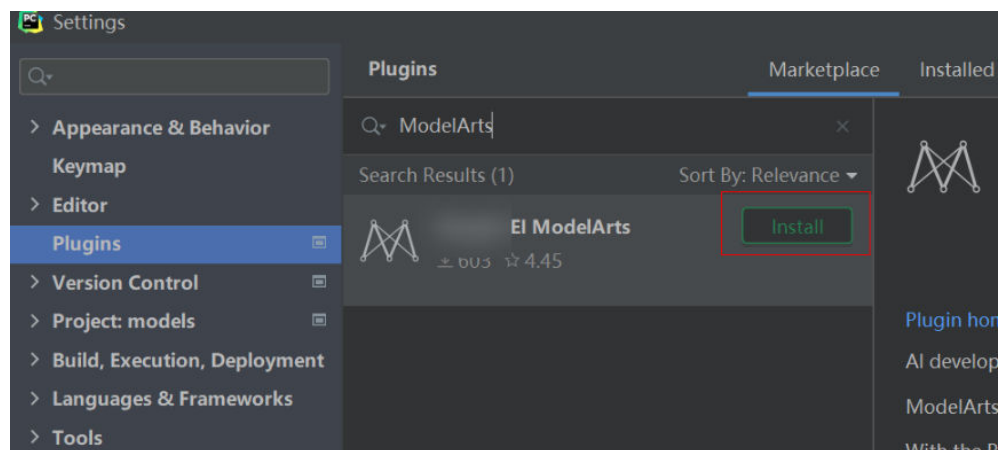
Comunidad de PyCharm o profesional 2019.2 o posterior se ha instalado localmente.

- Solo se pueden utilizar PyCharm de la edición profesional para acceder al entorno del desarrollo de notebooks.
- Puede utilizar una edición comunitaria o profesional de PyCharm Toolkit para enviar los trabajos de entrenamiento. PyCharm Toolkit 2.x se puede usar para enviar solo la versión anterior de los trabajos de entrenamiento, y la última versión de PyCharm Toolkit se puede usar para enviar solo la nueva versión de los trabajos de entrenamiento.

Instalación con Marketplace

En PyCharm, seleccione **File > Settings > Plugins**, encuentre **ModelArts** en Marketplace y haga clic en **Install**.

Figura 5-2 Instalación con Marketplace



NOTA

- La versión instalada en Marketplace es la última versión.
- Si no se encuentra ModelArts en Marketplace, es posible que su red esté restringida. Asegúrese de que puede acceder a Internet.

5.2.1.3 Conexión a una instancia de notebook con PyCharm Toolkit

ModelArts proporciona un complemento de PyCharm llamado PyCharm Toolkit para que pueda acceder de forma remota a una instancia de notebook con SSH, cargar código, enviar un trabajo de entrenamiento y obtener registros de entrenamiento para su visualización local.

Requisitos previos

PyCharm professional 2019.2 o posterior se ha instalado localmente. El SSH remoto solo se aplica a la edición profesional de PyCharm. [Descargue PyCharm](#) e instálelo.

NOTA

Descargue PyCharm Professional 2023.2 o una versión anterior. El PyCharm Toolkit no se adapta a PyCharm Professional cuya versión sea posterior a 2023.2.

Paso 1 Crear una instancia de notebook

Cree una instancia de notebook con SSH remoto habilitado y lista blanca configurada. Asegúrese de que la instancia se está ejecutando. Para obtener más detalles, véase [Creación de una instancia de notebook](#).

Paso 2 Descargar e instalar PyCharm Toolkit

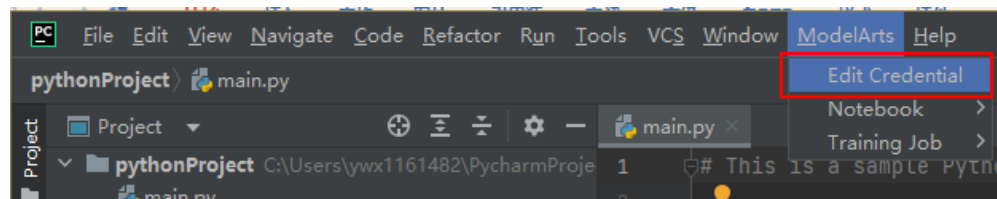
En PyCharm seleccione **File > Settings > Plugins**, busque **ModelArts** en Marketplace y haga clic en **Install**. Para obtener más detalles, véase [Descarga e instalación de PyCharm Toolkit](#).

Paso 3 Iniciar sesión en el complemento

Para utilizar el par de AK/SK para la autenticación de inicio de sesión, realice los siguientes pasos:

1. Abra PyCharm con Toolkit instalado. Elija **ModelArts > Edit Credential** en la barra de menú.

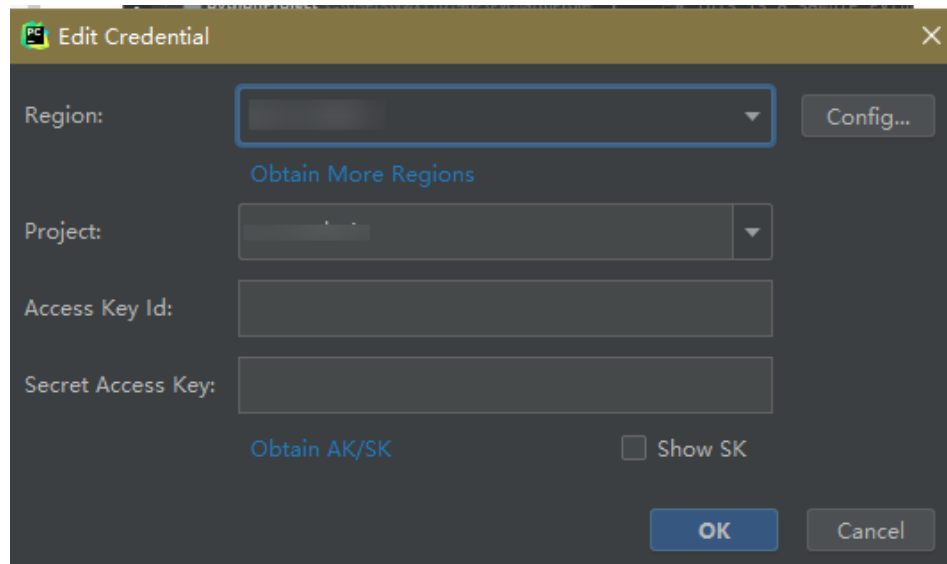
Figura 5-3 Editar Credencial



2. En el cuadro de diálogo que aparece en pantalla, seleccione la región donde se encuentra ModelArts, introduzca AK y SK y haga clic en **OK**. Para obtener detalles sobre cómo obtener AK y SK, véase [¿Cómo obtengo una clave de acceso?](#).

- **Region:** Seleccione una región de la lista desplegable. Debe coincidir con la región de la consola de ModelArts.
- **Project:** una vez seleccionada la región, el proyecto se rellena automáticamente.
- **Access Key ID:** Introduzca la AK.
- **Secret Access Key:** Introduzca la SK.

Figura 5-4 Introducir la región y las claves de acceso



3. Visualizar el resultado de la verificación.

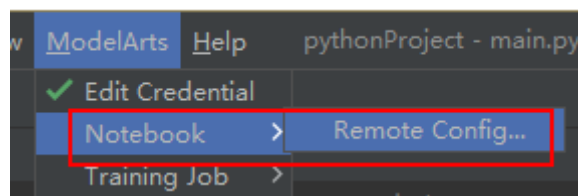
En el área **Event Log**, si se muestra la información similar a la siguiente, la clave de acceso se ha agregado correctamente:

```
16:01Validate Credential Success: The HUAWEI CLOUDcredential is valid.
```

Paso 4 Configurar automáticamente el PyCharm Toolkit

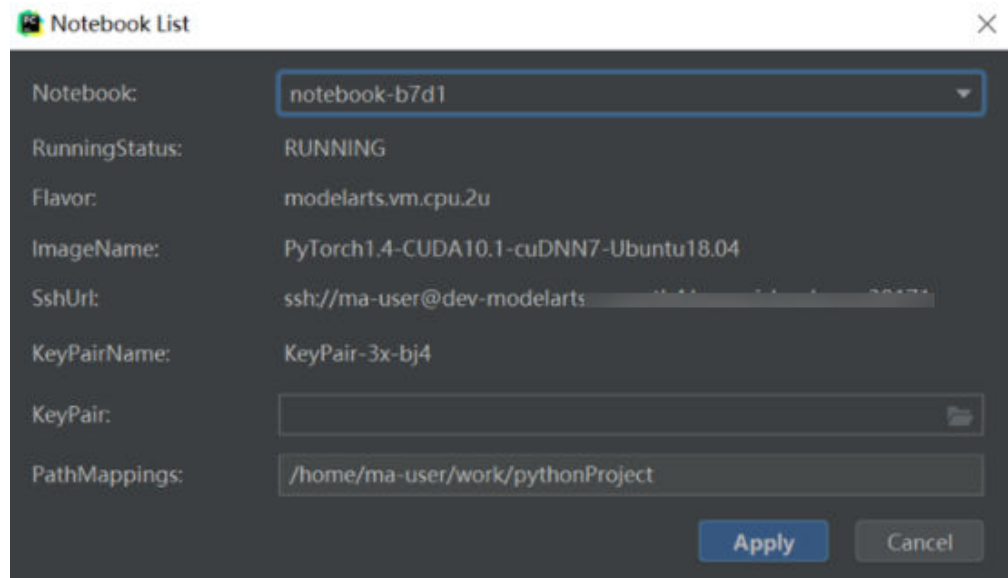
1. En el entorno de desarrollo local de PyCharm, seleccione **ModelArts > Notebook > Remote Config...** y configure el PyCharm Toolkit.

Figura 5-5 Conexión remota al PyCharm Toolkit



2. Seleccione la instancia de destino de la lista desplegable, donde se muestran todas las instancias de notebook con SSH remoto habilitado en la cuenta.

Figura 5-6 Lista de notebook



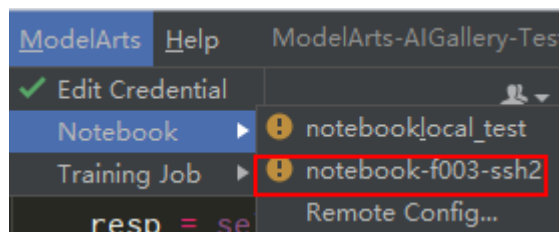
- **KeyPair:** Seleccione el par de claves almacenado localmente de la instancia del notebook para la autenticación. El par de claves creado durante la creación de la instancia del notebook se guarda en la carpeta de descargas predeterminada del navegador.
 - **PathMappings:** El directorio de sincronización para el proyecto de IDE local y el notebook, que por defecto es `/home/ma-user/work/Project name` y es ajustable.
3. Haga clic en **Apply**. Una vez completada la configuración, reinicie el IDE para que la configuración surta efecto.

Después del reinicio, se tarda unos 20 minutos en actualizar el intérprete de Python por primera vez.

Paso 5 Acceder a una instancia de notebook a través del PyCharm Toolkit

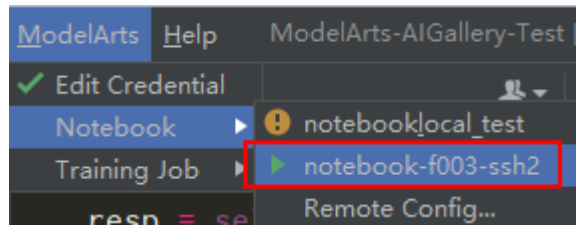
Haga clic en el nombre de la instancia del notebook y conéctelo al IDE local según se le solicite. La conexión se mantiene durante 4 horas de forma predeterminada.

Figura 5-7 Iniciar la conexión



Para interrumpir la conexión, haga clic en el nombre del notebook y desconéctelo del IDE local según se le solicite.

Figura 5-8 Interrumpir la conexión



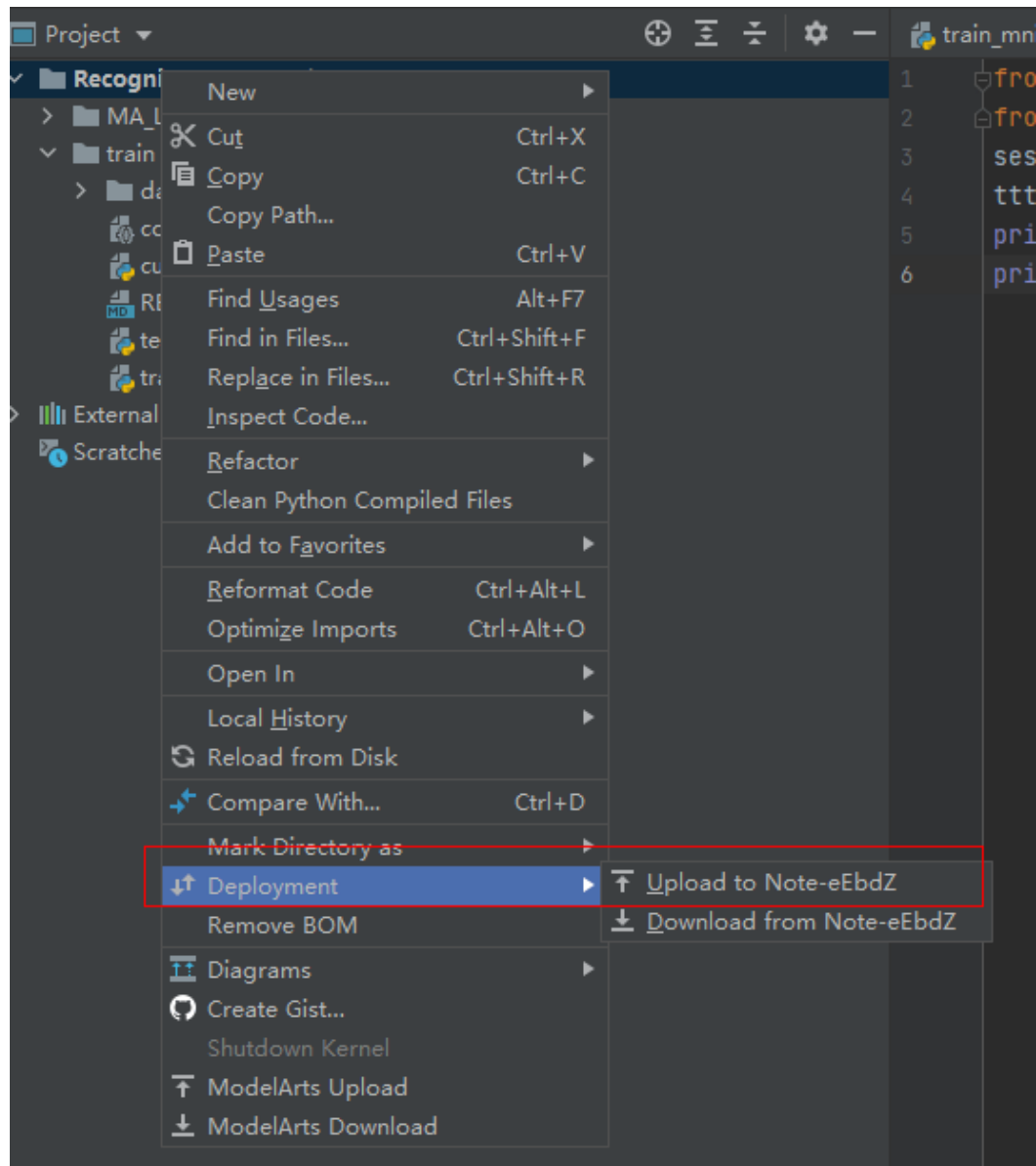
Paso 6 Cargar archivos locales en la instancia de notebook

El código de un archivo local se puede copiar al IDE local, que sincronizará automáticamente el código con el entorno de desarrollo en la nube.

Sincronización inicial

En el directorio **Project** del IDE local, haga clic derecho en **Deployment** y elija **Upload to Notebook name** en el menú contextual para cargar el archivo de proyecto local en la instancia de notebook especificada.

Figura 5-9 Sincronización de datos locales con una instancia de notebook

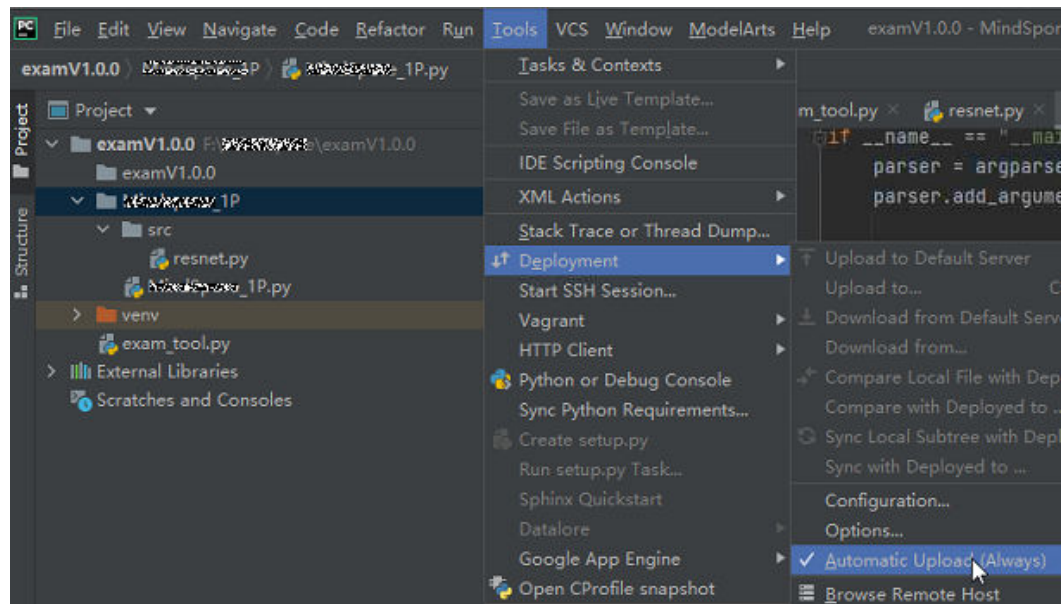


Sincronización siguientes

Después de modificar el código, presione **Ctrl+S** para guardarlo. El IDE local sincronizará automáticamente la modificación con la instancia de notebook especificada.

Después de instalar PyCharm Toolkit, se activa automáticamente **Automatic Upload** en el IDE local para cargar automáticamente los archivos en el directorio local a la instancia del notebook de destino. Si **Automatic Upload** no está habilitado, habilítelo haciendo referencia a la siguiente figura.

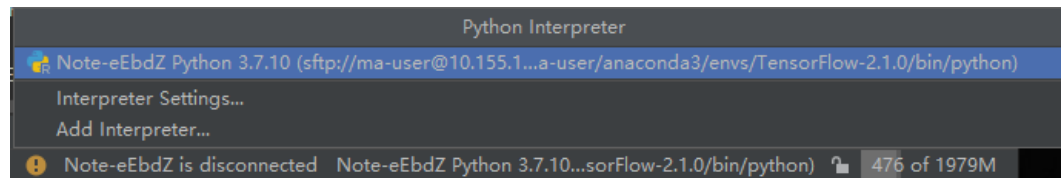
Figura 5-10 Activación de la carga automática



Paso 7 Depurar remotamente el código

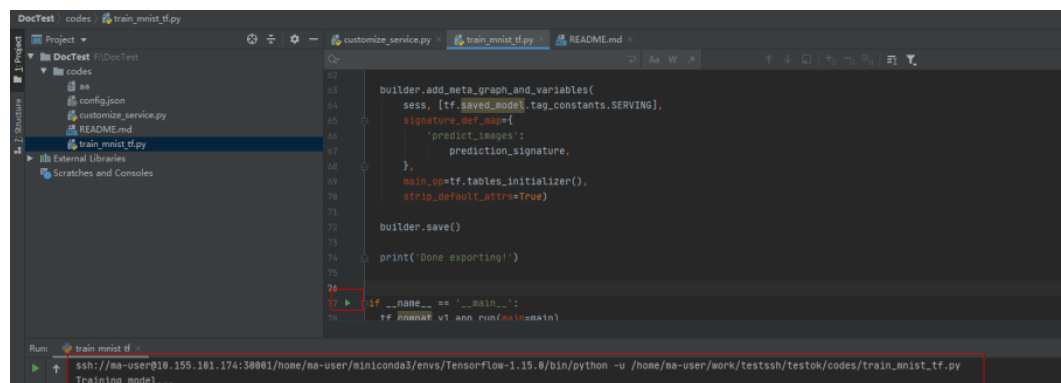
Haga clic en **Interpreter** en la esquina inferior derecha del IDE local y seleccione un intérprete de Python de notebook.

Figura 5-11 Selección de un intérprete de Python



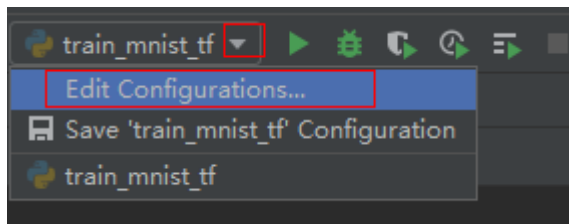
Ejecute el código en la instancia del notebook. Los logs se muestran localmente.

Figura 5-12 Logs de tiempo de ejecución



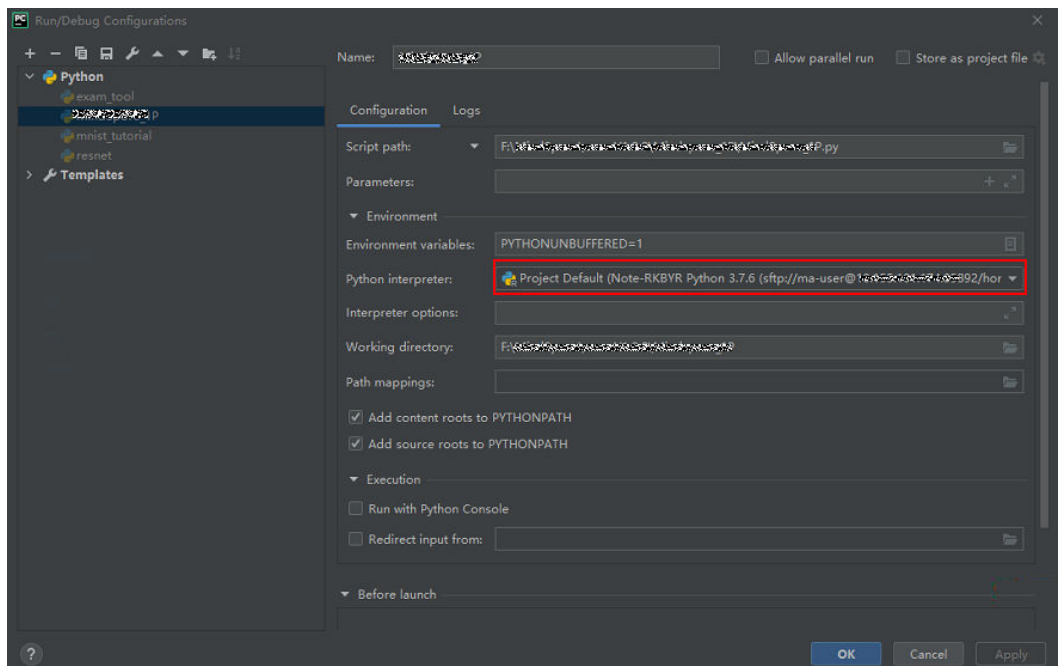
Haga clic en **Run/Debug Configurations** en la esquina superior derecha del IDE local para establecer los parámetros de tiempo de ejecución.

Figura 5-13 Configuración de parámetros de tiempo de ejecución (1)



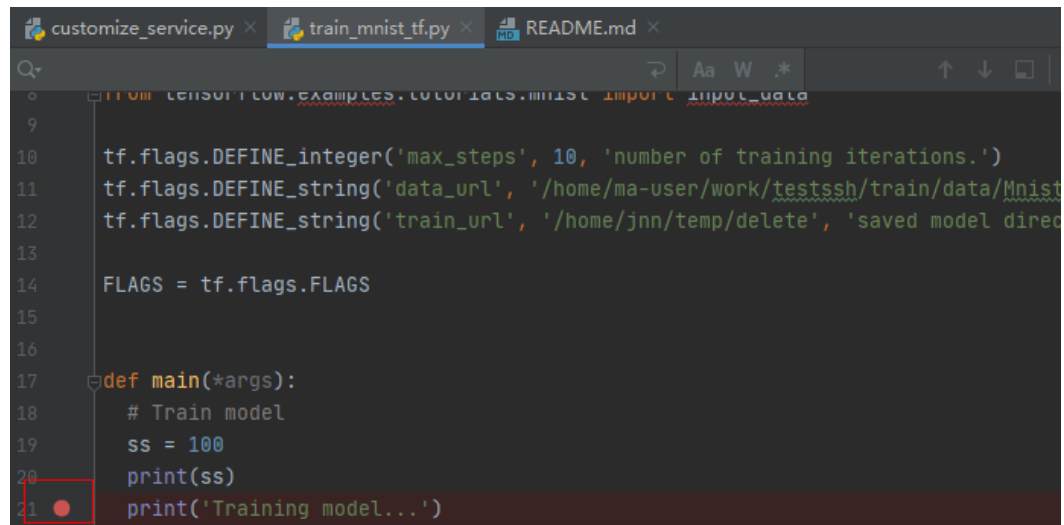
Seleccione el intérprete de Python que se conecta de forma remota a la instancia del notebook de destino.

Figura 5-14 Configuración de parámetros de tiempo de ejecución (2)



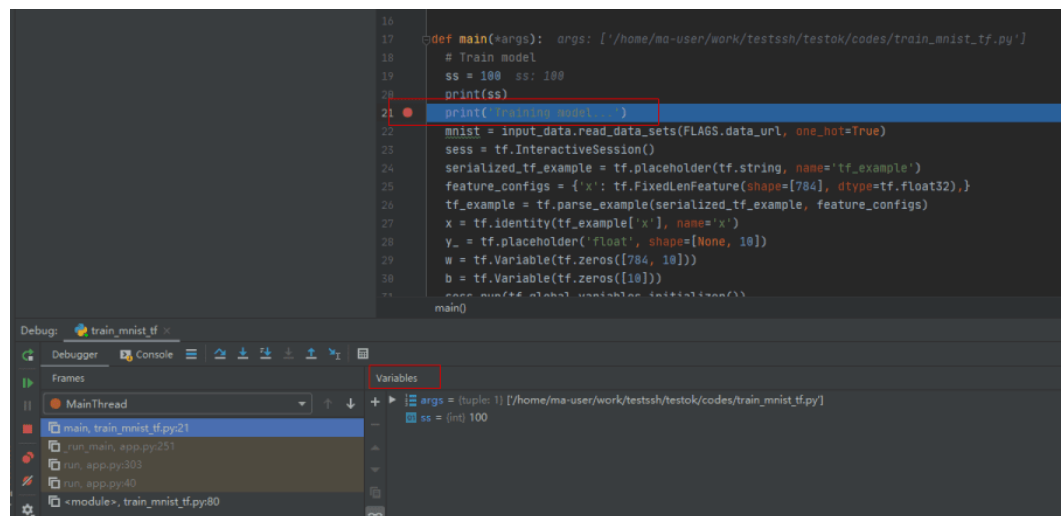
Para depurar código, establezca puntos de interrupción y ejecute el programa en modo de depuración.

Figura 5-15 Ejecución del programa en modo de depuración



En modo de depuración, la ejecución de código se suspende en la línea especificada, y puede obtener valores de variables.

Figura 5-16 Consulta de valores de variables en modo de depuración



5.2.2 Conexión manual a una instancia de notebook con PyCharm

Un IDE local admite PyCharm y VS Code. Puede utilizar VS Code o PyCharm para conectar remotamente el IDE local a la instancia de notebook de destino de ModelArts para ejecutar y depurar el código.

En esta sección se describe cómo utilizar PyCharm para acceder a una instancia de notebook.

Requisitos previos

- PyCharm profesional 2019.2 o posterior se ha instalado localmente. La edición profesional de PyCharm está disponible porque SSH remoto solo se aplica a la edición profesional.

- Se ha creado una instancia de notebook con SSH remoto habilitado. Asegúrese de que la instancia se está ejecutando. Para obtener más detalles, véase [Creación de una instancia de notebook](#).
- La dirección y el número de puerto del entorno de desarrollo están disponibles. Para obtener esta información, vaya a la página de detalles de la instancia del notebook.

Figura 5-17 Página de detalles de la Instancia

Address	ssh://ma-user@dev-modelarts- [redacted] .com	32651	
Authentication	KeyPair-9a64 	Access address of the development environment	Port number

- El par de claves está disponible.
Un par de claves se descarga automáticamente después de crearlo. Guárdelo de forma segura. Si se pierde un par de claves existente, cree uno nuevo.

Paso 1 Configurar SSH


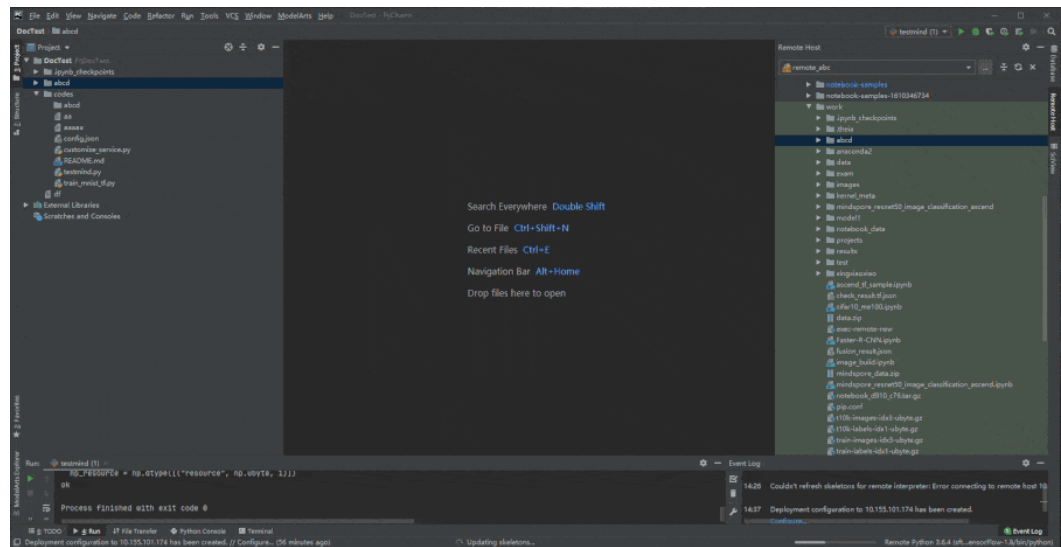
1. En su entorno de desarrollo de PyCharm local, elija **File > Settings > Tools > SSH Configurations** y haga clic en + para agregar una configuración de SSH.
 - **Host:** la dirección para acceder al entorno de desarrollo en la nube. Obtenga la dirección en la página que proporciona la información detallada de la instancia del notebook de destino.
 - **Port:** número de puertos para acceder al entorno de desarrollo en la nube. Obtenga el número de puerto en la página que proporciona la información detallada de la instancia del notebook de destino.
 - **User name:** se establece de forma consistente en **ma-user**.
 - **Authentication type:** par de claves
 - **Private key file:** el archivo de clave privada almacenado localmente del entorno de desarrollo en la nube. Es el archivo de par de claves que se descarga automáticamente al crear la instancia del notebook.
2. Haga clic en  para cambiar el nombre de la conexión. Luego, haga clic en **OK**.
3. Una vez completada la configuración, haga clic en **Test Connection** para probar la conectividad.
4. Seleccione **Yes**. Si se muestra "Successfully connected", se puede acceder a la red. Luego, haga clic en **OK**.
5. Haga clic en **OK** en la parte inferior para guardar la configuración.

Figura 5-18 Configuración de SSH

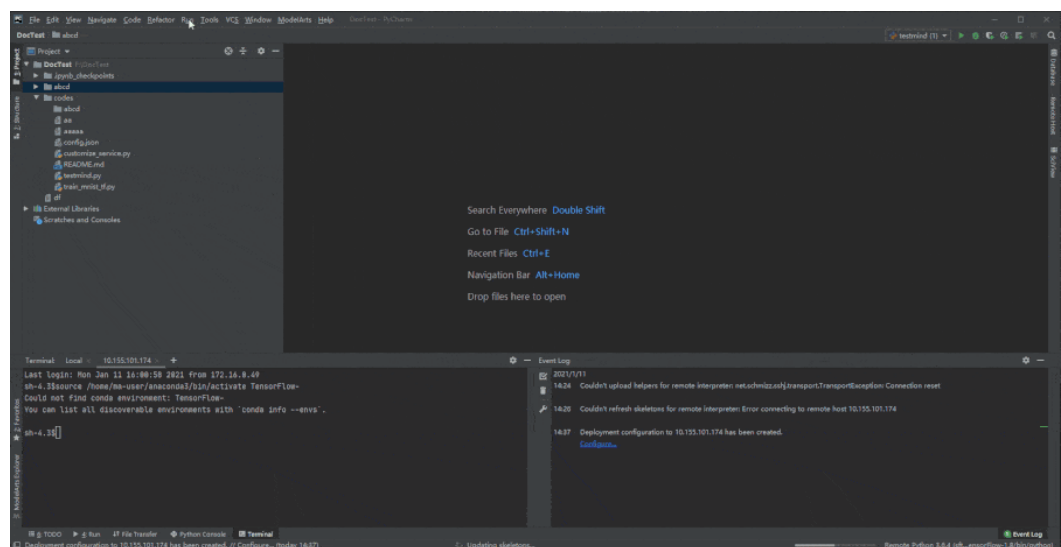


Paso 2 Obtener la ruta al entorno virtual integrado en el entorno de desarrollo


1. Elija **Tools > Start SSH Session** para acceder al entorno de desarrollo en la nube.
2. Ejecute el siguiente comando para ver los entornos virtuales de Python construidos en el entorno actual en el archivo **README** en **/home/ma-user/**:

```
cat /home/ma-user/README
```
3. Ejecute el comando **source** para cambiar a un entorno específico de Python.
4. Ejecute **which python** para obtener la ruta de Python y cópiela para configurar el intérprete de Python en la nube.

Figura 5-19 Obtención de la ruta hacia el entorno virtual integrado en el entorno de desarrollo



Paso 3 Configurar un intérprete de Python

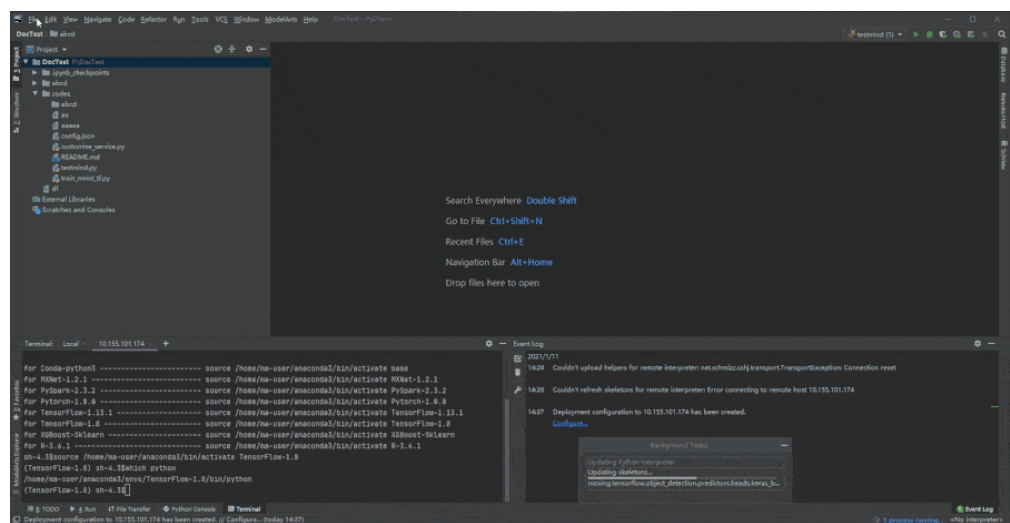
1. Elija **File > Settings > Project: Python project > Python Interpreter**. A continuación, haga clic en  y **Add** para agregar un intérprete.

2. Seleccione **Existing server configuration** en la lista desplegable, elija la configuración de SSH y haga clic en **Next**.
3. Configure el intérprete de Python.
 - **Interpreter:** introduzca la ruta de Python copiada en el paso 1, por ejemplo, **/home/ma-user/anaconda3/envs/Pytorch-1.0.0/bin/python**.
Si la ruta de acceso es **~/anaconda3/envs/Pytorch-1.0.0/bin/python**, reemplace ~ por **/home/ma-user**.
 - **Sync folders:** establezca este parámetro en un directorio en el entorno de desarrollo en la nube para sincronizar archivos de directorio de proyectos locales. Se recomienda un directorio de **/home/ma-user**, por ejemplo, el **/home/ma-user/work/projects** porque otros directorios pueden tener prohibido el acceso.
4. Haga clic en **!** a la derecha y seleccione **Automatically upload** para que el archivo modificado localmente se pueda cargar automáticamente en el contenedor.
5. Haga clic en **Finish**.

El archivo de proyecto local se ha cargado automáticamente en el entorno de nube. Cada vez que se modifica un archivo local, la modificación se sincroniza automáticamente con el entorno de nube.

En la esquina inferior derecha, el intérprete actual se muestra como un intérprete remoto.

Figura 5-20 Configuración de un intérprete de Python



Paso 4 Instalar la biblioteca dependiente para el entorno de nube

Después de acceder al entorno de desarrollo, puede utilizar diferentes entornos virtuales, como TensorFlow y PyTorch. Sin embargo, en el desarrollo real, es necesario instalar paquetes de dependencias. A continuación, puede acceder al entorno a través de la terminal para realizar operaciones.

Seleccione **Tools > Start SSH Session** y seleccione el entorno de desarrollo configurado. Ejecute el comando **pip install** para instalar los paquetes de dependencias requeridos.

```
Terminal: Local x 10.10.10.10 +
Last login: Wed Dec 30 12:46:18 2020 from 10.10.10.10
sh-4.4$cat /home/ma-user/README
Please use one of following command to start the specified framework environment.

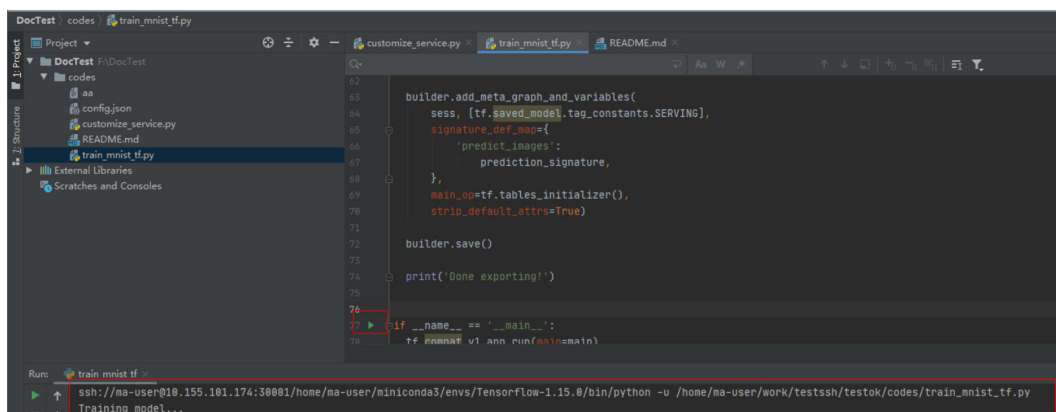
for Mindspore-1.0.1-python3.7-aarch64 ----- source /home/ma-user/miniconda3/bin/activate Mindspore-1.0.1-python3.7-aarch64
for Tensorflow-1.15.0 ----- source /home/ma-user/miniconda3/bin/activate Tensorflow-1.15.0
sh-4.4$source /home/ma-user/miniconda3/bin/activate Tensorflow-1.15.0
(Tensorflow-1.15.0) sh-4.4$which python
~/miniconda3/envs/Tensorflow-1.15.0/bin/python
(Tensorflow-1.15.0) sh-4.4$pwd
/home/ma-user
(Tensorflow-1.15.0) sh-4.4$pip install spacy
```

Paso 5 Código de depuración en el entorno de desarrollo

Ha accedido al entorno de desarrollo en la nube. Luego, puede escribir, depurar y ejecutar el código en el PyCharm local. El código se ejecuta realmente en el entorno de desarrollo en la nube y se utilizan los recursos de AI de Ascend en la nube. De esta manera, compile y modifique el código localmente y ejecute el código en la nube.

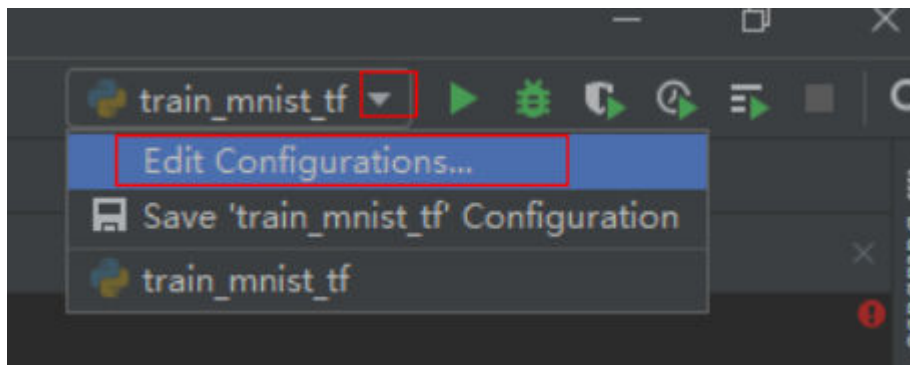
Ejecute el código en el IDE local. Los logs se pueden mostrar localmente.

Figura 5-21 Código de depuración



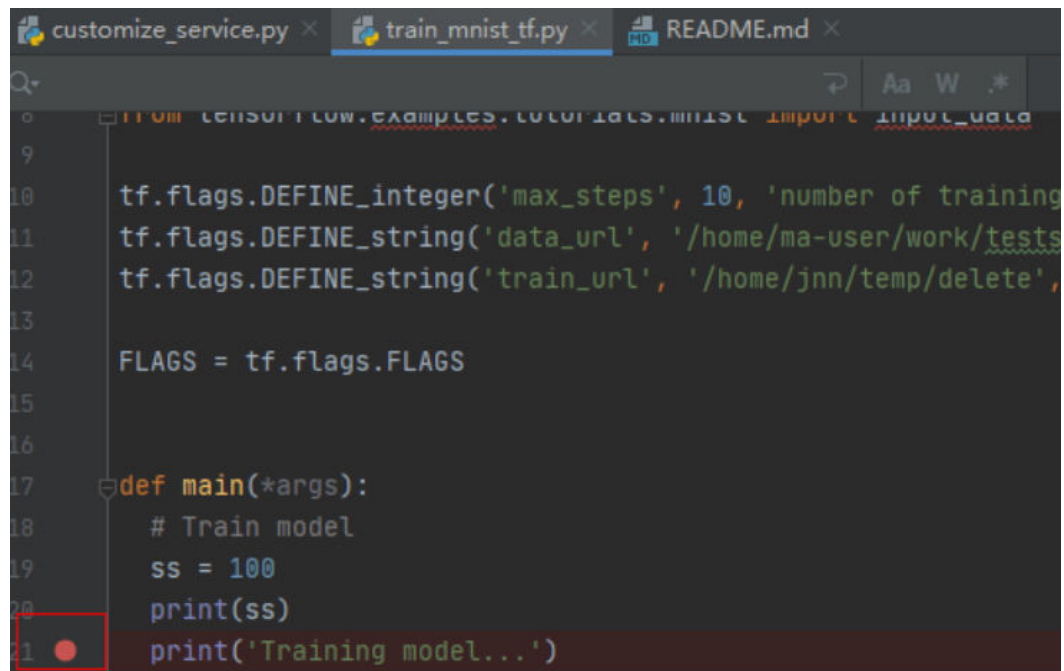
Haga clic en **Run/Debug Configurations** en la esquina superior derecha del IDE local para establecer los parámetros de tiempo de ejecución.

Figura 5-22 Configuración de parámetros de tiempo de ejecución



Para depurar código, establezca puntos de interrupción y ejecute el programa en modo de depuración.

Figura 5-23 Punto de interrupción de código



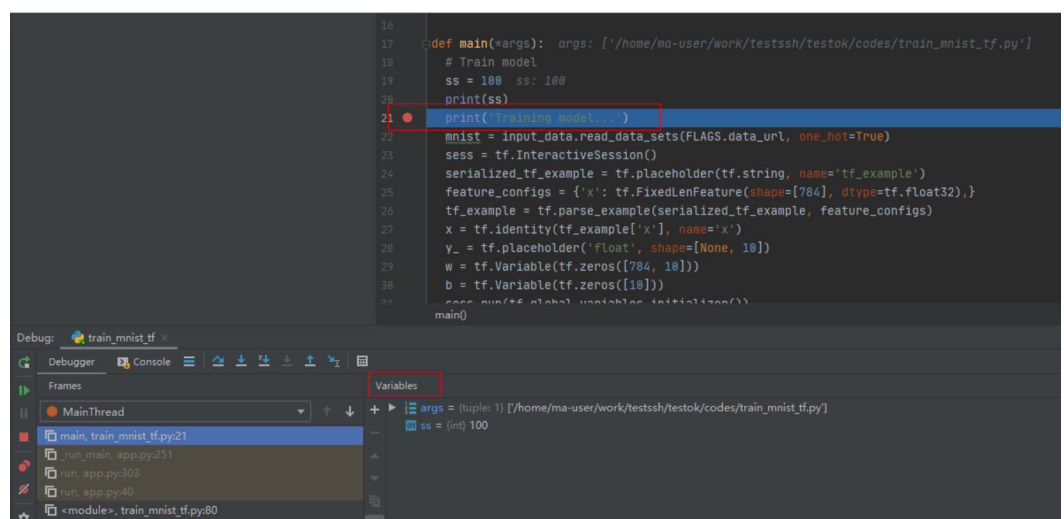
```
customize_service.py x train_mnist_tf.py x README.md x
9
10 tf.flags.DEFINE_integer('max_steps', 10, 'number of training
11 tf.flags.DEFINE_string('data_url', '/home/ma-user/work/testst
12 tf.flags.DEFINE_string('train_url', '/home/jnn/temp/delete',
13
14 FLAGS = tf.flags.FLAGS
15
16
17 def main(*args):
18     # Train model
19     ss = 100
20     print(ss)
21     print('Training model...')
```

Figura 5-24 Depuración en modo de depuración



En modo de depuración, la ejecución de código se suspende en la línea especificada, y puede obtener valores de variables.

Figura 5-25 Modo de depuración



Antes de depurar código en modo de depuración, asegúrese de que el código local es el mismo que el código de la nube. Si son diferentes, la línea donde se agregue localmente un punto de interrupción puede ser diferente de la línea del código de nube, lo que lleva a errores.

Al configurar un intérprete de Python en el entorno de desarrollo de la nube, seleccione **Automatically upload** para que cualquier modificación de archivo local se pueda cargar automáticamente en la nube. Si no selecciona **Automatically upload**, cargue manualmente el directorio o código después de modificar el código local. Para obtener más detalles, véase [Paso 6 Cargar archivos locales en la instancia de notebook](#).

5.2.3 Envío de un trabajo de entrenamiento con PyCharm Toolkit

5.2.3.1 Envío de un trabajo de entrenamiento (Nueva versión)

Puede utilizar PyCharm Toolkit de la última versión para enviar rápidamente el código de entrenamiento desarrollado localmente a ModelArts para su entrenamiento.

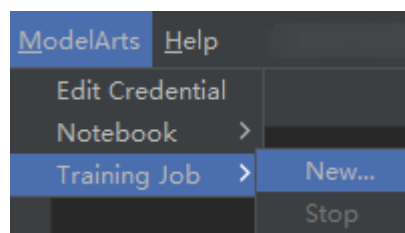
Requisitos previos

- En PyCharm local existe un proyecto de código de entrenamiento.
- Ha creado un bucket y carpetas en OBS para almacenar conjuntos de datos y modelos entrenados. Los datos utilizados por el trabajo de entrenamiento se cargaron en OBS.
- Se ha configurado la credencial. Para obtener más detalles, véase [Uso de claves de acceso para iniciar sesión](#).
- PyCharm Toolkit de la última versión solo está disponible para enviar un trabajo de entrenamiento de la nueva versión.

Configuración de parámetros del trabajo de entrenamiento

1. En PyCharm, abra el proyecto de código de entrenamiento y el archivo de inicio de entrenamiento, y elija **ModelArts > Training Job > New...** en la barra de menú.

Figura 5-26 Editar la configuración del trabajo de entrenamiento



2. En el cuadro de diálogo que aparece en pantalla, configure los parámetros del trabajo de entrenamiento. Para obtener más detalles, véase [Tabla 5-2](#).

Tabla 5-2 Parámetros de trabajo de entrenamiento

Parámetro	Descripción
Job Name	<p>Nombre de un trabajo de entrenamiento</p> <p>El sistema genera automáticamente un nombre. Puede cambiar su nombre basándose en las siguientes reglas de nombres:</p> <ul style="list-style-type: none"> ● El nombre contiene de 1 a 64 caracteres. ● Se permiten letras, dígitos, guiones medios (-) y guiones bajos (_).
Job Description	Breve descripción de un trabajo de entrenamiento
Algorithm Source	<p>Fuente del algoritmo de entrenamiento. Las opciones son Frequently-used y Custom.</p> <p>Frequently-used se refiere a los motores de IA de uso frecuente que soporta la gestión de entrenamiento de ModelArts.</p> <p>Si el motor de IA que utiliza no está en la lista compatible, se recomienda crear un trabajo de entrenamiento utilizando una imagen personalizada.</p>
AI Engine	<p>Seleccione el motor de IA y la versión utilizada en el código. Los motores de IA admitidos son los mismos que los marcos de uso frecuente soportados por trabajos de entrenamiento en la consola de gestión de ModelArts.</p>
Boot File Path	<p>Archivo de arranque de entrenamiento. El archivo de arranque seleccionado debe ser un archivo del proyecto de entrenamiento actual de PyCharm. Este parámetro se muestra si Algorithm Source se establece en Frequently-used.</p>
Code Directory	<p>Directorio de código de entrenamiento. El sistema establece automáticamente este parámetro en el directorio donde se encuentra el archivo de arranque de entrenamiento. Puede cambiar el valor del parámetro a un directorio que esté en el proyecto actual y que contenga el archivo de arranque.</p> <p>Si la fuente del algoritmo es una imagen personalizada y el código de entrenamiento se ha construido en la imagen, este parámetro se puede dejar en blanco.</p>
Ruta de acceso a la imagen (opcional)	URL de la imagen de SWR
Boot Command	<p>Comando para iniciar un trabajo de entrenamiento, por ejemplo, bash /home/work/run_train.sh python {Python boot file and parameters}. Este parámetro se muestra si Algorithm Source está establecido en Custom.</p> <p>Si el comando no contiene el parámetro --data_url o --train_url, la herramienta agrega automáticamente los dos parámetros al final del comando al enviar el trabajo de entrenamiento. Los dos parámetros corresponden a la ruta de OBS para almacenar datos de entrenamiento y la ruta de OBS para almacenar la salida de entrenamiento, respectivamente.</p>

Parámetro	Descripción
Data OBS Path	Ruta de OBS para almacenar datos de entrenamiento, por ejemplo, <code>/test-modelarts2/mnist/dataset-mnist/</code> , en la que test-modelarts2 indica un nombre de bucket.
Training OBS Path	Ruta de OBS. Un directorio se crea automáticamente en la ruta para almacenar un modelo entrenado y logs de entrenamiento.
Running Parameters	Parámetros de ejecución. Si desea agregar algunos parámetros de ejecución a su código, agréguelos aquí. Separe varios parámetros de ejecución con punto y coma (;), por ejemplo, key1=value1;key2=value2 . Este parámetro se puede dejar en blanco.
Specifications	Tipo de recursos utilizados para el entrenamiento. Actualmente, se apoyan los pools de recursos públicos y pools de recursos dedicados. Dedicated Resource Pool identifica las especificaciones del grupo de recursos dedicados. Las especificaciones del grupo de recursos dedicados se muestran solo para los usuarios que han comprado grupos de recursos dedicados.
Compute Nodes	Cantidad de nodos de cómputo. Si este parámetro se establece en 1 , el sistema se ejecuta en modo independiente. Si este parámetro se establece en un valor mayor que 1, el modo de cómputo distribuido se utiliza en segundo plano.
Available/Total Nodes	Cuando Specifications se establece en un grupo de recursos dedicado, se muestra el número de nodos disponibles y el número total de nodos. El valor de Compute Nodes no puede exceder el número de nodos disponibles.

Figura 5-27 Configuración del parámetro de trabajo de entrenamiento (grupo de recursos públicos)

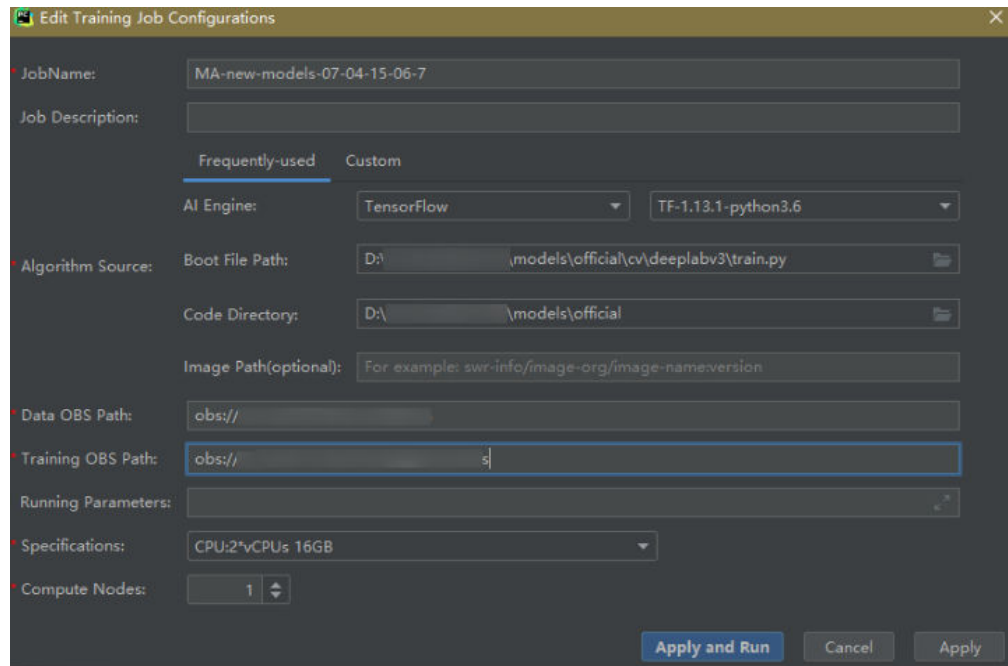


Figura 5-28 Configuración del parámetro de trabajo de entrenamiento (grupo de recursos dedicados)

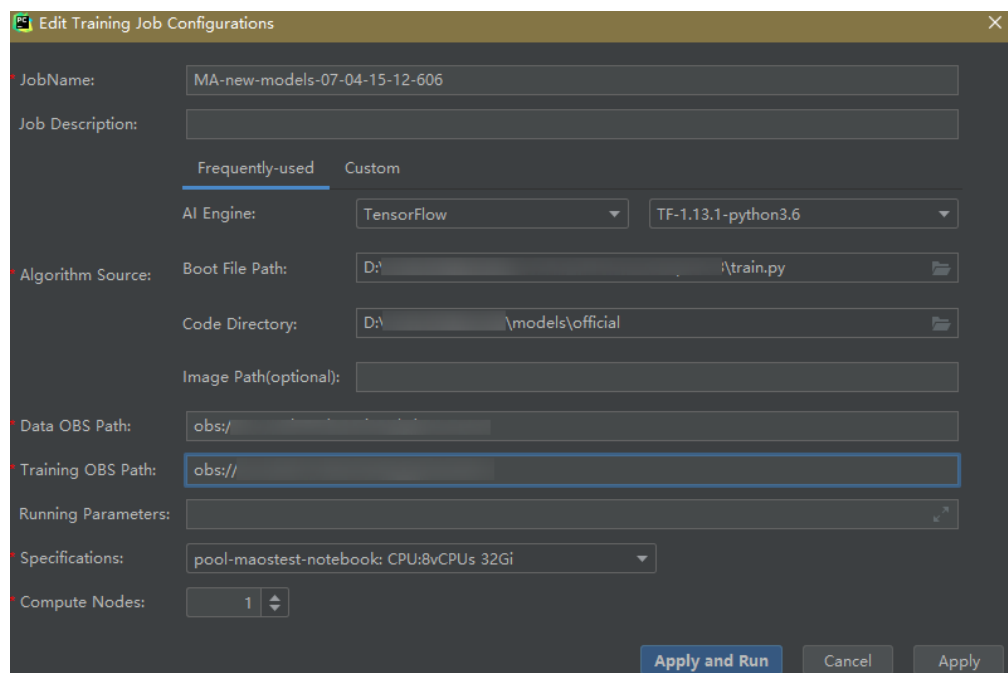
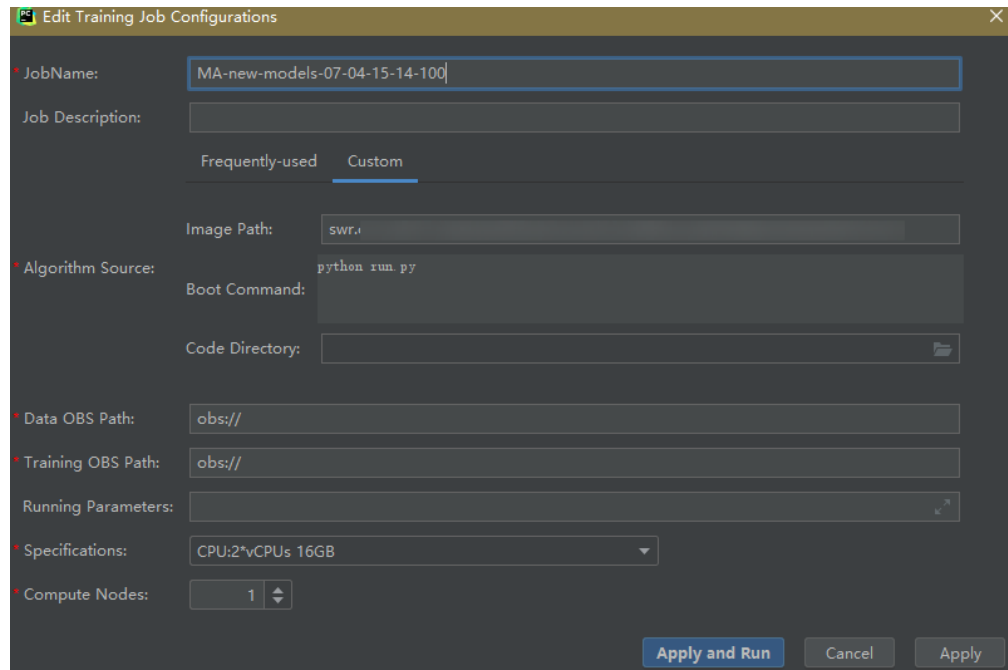


Figura 5-29 Configuración del parámetro de trabajo de entrenamiento (imagen personalizada)

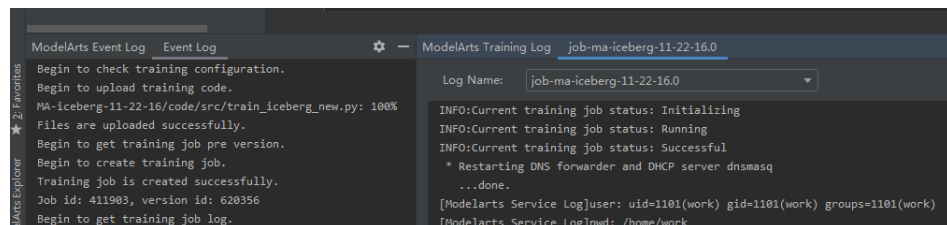


3. Después de establecer los parámetros, haga clic en **Apply and Run**. Luego, el código local se carga automáticamente en la nube y se inicia el entrenamiento. El estado de ejecución del trabajo de entrenamiento se muestra en el área **Training Log** en tiempo real. Si se muestra información similar a **Current training job status: Successful** en el log del entrenamiento, el trabajo de entrenamiento se ha ejecutado correctamente.

NOTA

- Después de hacer clic en **Apply and Run**, el sistema ejecuta automáticamente el trabajo de entrenamiento. Para detener el trabajo de entrenamiento, elija **ModelArts > Training Job > Stop** en la barra de menú.
- Si hace clic en **Apply**, el trabajo no se inicia directamente y, en su lugar, se guarda la configuración del trabajo de entrenamiento. Para iniciar el trabajo, haga clic en **Apply and Run**.

Figura 5-30 Ejemplo de log de entrenamiento



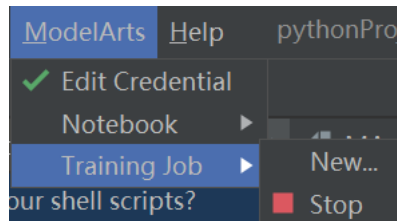
5.2.3.2 Detención de un trabajo de entrenamiento

Puede detener un trabajo de entrenamiento en ejecución.

Detener un trabajo

Cuando se está ejecutando un trabajo de entrenamiento, elija **ModelArts > Training Job > Stop** en la barra de menú de PyCharm para detener el trabajo.

Figura 5-31 Detener un trabajo



5.2.3.3 Consulta de logs de entrenamiento

En esta sección se describe cómo ver los logs de trabajos de entrenamiento.

Consulta de los logs de entrenamiento en OBS

Cuando envía un trabajo de entrenamiento, el sistema crea automáticamente una carpeta con el mismo nombre que el trabajo de entrenamiento en la ruta de acceso de OBS configurada para almacenar el modelo, los logs y el código generados una vez finalizado el entrenamiento.

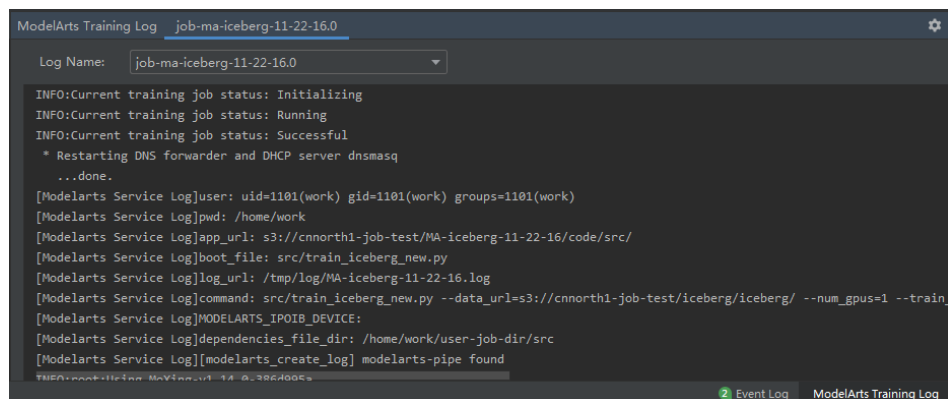
Por ejemplo, cuando se envía el trabajo **train-job-01**, se crea una carpeta llamada **train-job-01** en el bucket **test-modelarts2**. En esta carpeta, se crean tres subcarpetas (**output**, **log** y **code**) para almacenar el modelo, los logs y el código de entrenamiento, respectivamente. Las subcarpetas se crearán en la carpeta **output** según la versión del trabajo de entrenamiento. A continuación se muestra un ejemplo de la estructura de carpetas:

```
test-modelarts2
|---train-job-01
|   |---output
|   |---log
|   |---code
```

Consulta de los logs de entrenamiento en Toolkit

En PyCharm, haga clic en **ModelArts Training Log** en la esquina inferior derecha de la página. Se muestran los logs de entrenamiento.

Figura 5-32 Consulta de logs de entrenamiento



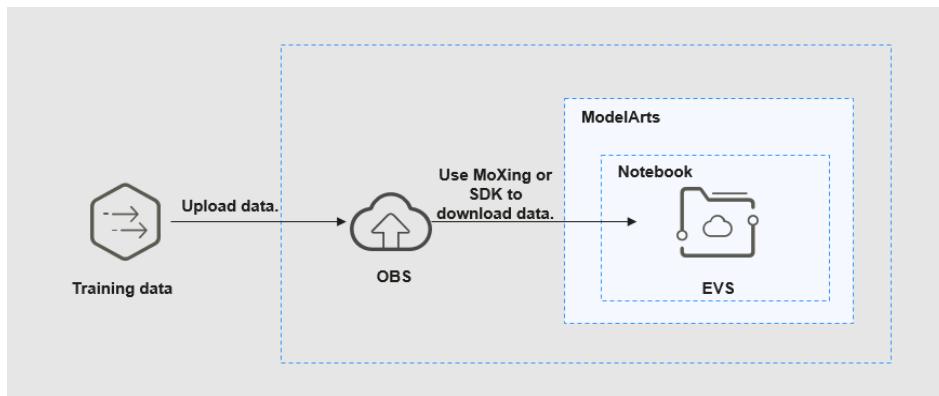
5.2.4 Carga de datos en una instancia de notebook con PyCharm

Si los datos son menores o iguales a 500 MB, copie directamente los datos en el IDE local.

Si los datos superan los 500 MB, cargue el código en OBS y luego en la instancia de notebook.

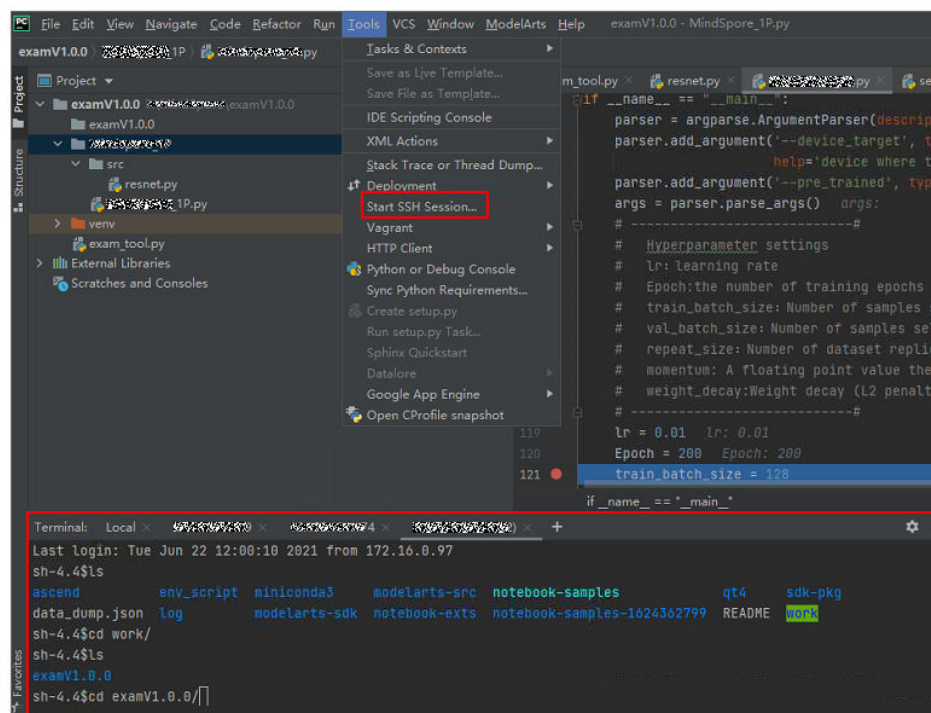
1. Cargue los datos en OBS. Para obtener más detalles, véase [Carga de un objeto](#).
2. Llame a la API de MoXing `mox.file.copy_parallel` provista por ModelArts en el terminal del IDE local para transferir datos desde OBS a la instancia del notebook.

Figura 5-33 Carga de datos a una instancia de notebook a través de OBS



A continuación se muestra cómo habilitar el terminal en PyCharm (las operaciones en VS Code son similares).

Figura 5-34 Habilitación del terminal en PyCharm



A continuación se muestra cómo utilizar MoXing en el terminal del IDE local para descargar archivos desde OBS a un entorno de desarrollo:

```
# Manually access the development environment.
cat /home/ma-user/README
# Select the source environment.
source /home/ma-user/miniconda3/bin/activate MindSpore-python3.7-aarch64
# Enter python and press Enter to enter the Python environment.
python
# Use MoXing for access.
import moxing as mox
# Download a folder from OBS to EVS.
mox.file.copy_parallel('obs://bucket_name/sub_dir_0', '/tmp/sub_dir_0')
```

5.3 IDE local (VS Code)

5.3.1 Conexión a una instancia de notebook por VS Code

Después de crear una instancia de notebook con SSH remoto habilitado, puede usar VS Code para acceder al entorno de desarrollo de cualquiera de las siguientes:

- **Conexión a una instancia de notebook a través del VS Code Toolkit** (recomendado)
Bajo este modo, inicie sesión en el complemento de VS Code Toolkit de ModelArts y utilícelo para conectarse a una instancia.
- En este modo, utilice el complemento de VS Code Remote-SSH para configurar la información de conexión y conectarse a una instancia.

5.3.2 Instalación de VS Code

URL para descarga:

- Dirección de descarga: https://code.visualstudio.com/updates/v1_85

Figura 5-35 URL de descarga de VS Code

November 2023 (version 1.85)

Update 1.85.1: The update addresses these [issues](#).

Update 1.85.2: The update addresses these [issues](#).

Downloads: Windows: [x64](#) [Arm64](#) | Mac: [Universal Intel silicon](#) | Linux: [deb](#) [rpm](#) [tarball](#) [Arm snap](#)

Requerimientos de versión de VS Code:

Se recomienda utilizar VS Code 1.85.2 o la versión más reciente para la conexión remota.

Guía de instalación de VS Code:

En Windows, haga doble clic en el paquete de instalación para completar la instalación.

En Linux, ejecute el comando `sudo dpkg -i code_1.85.2-1705561292_amd64.deb` para instalar VS Code.

 **NOTA**

Los usuarios del sistema de Linux deben instalar VS Code como usuario no root.

5.3.3 Conexión a una instancia de notebook a través del VS Code Toolkit

Esta sección describe cómo utilizar el complemento de VS Code Toolkit de ModelArts para conectarse remotamente a una instancia de notebook.

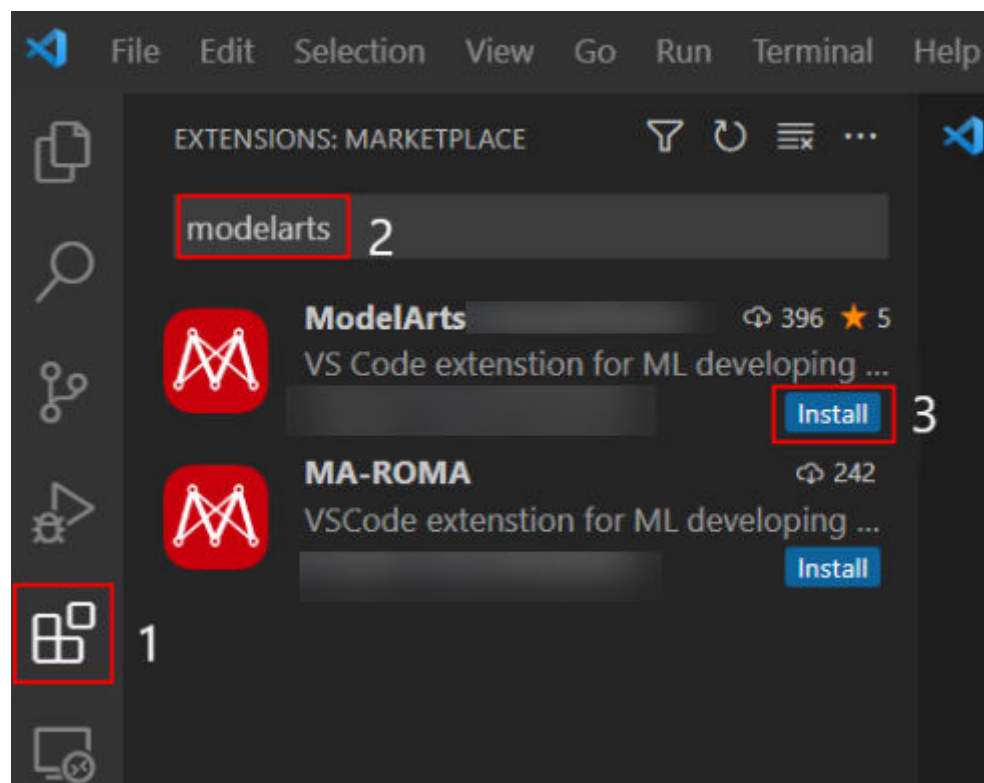
Requisitos previos

Ha descargado e instalado VS Code. Para obtener más detalles, véase [Instalación de VS Code](#).

Paso 1 Instalar el complemento de VS Code

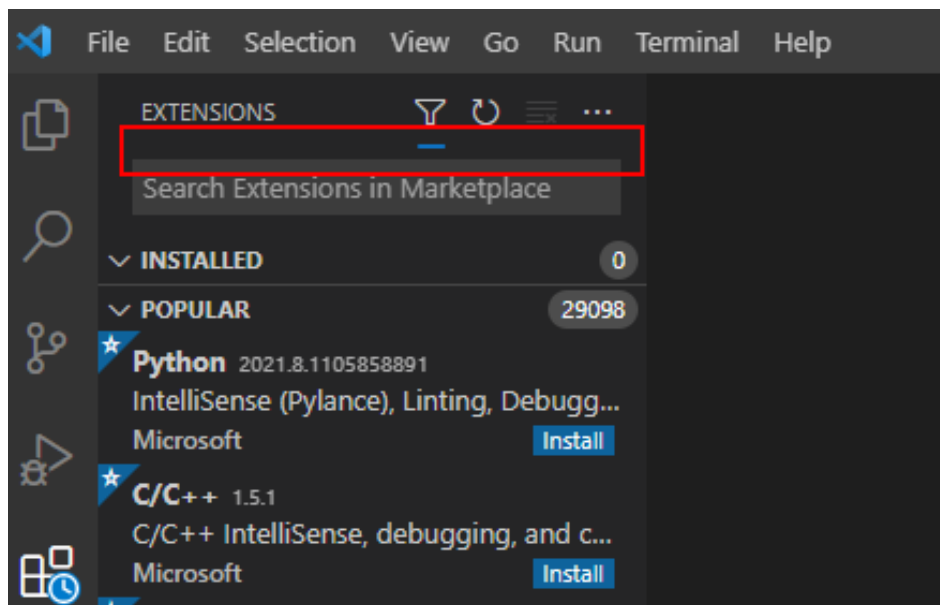
1. Busque **ModelArts-HuaweiCloud** en el cuadro de texto **EXTENSIONS** y haga clic en **Install**.

Figura 5-36 Instalación del complemento de VS Code



2. Espere de 1 a 2 minutos.

Figura 5-37 Proceso de instalación





3. Una vez completada la instalación, compruebe el mensaje que aparece en la esquina inferior derecha. Si el icono  de ModelArts y el icono  de SSH remoto se muestran en el panel de navegación de la izquierda, se instala el complemento de VS Code.

Figura 5-38 Mensaje de finalización de la instalación

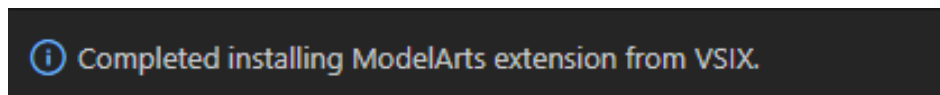
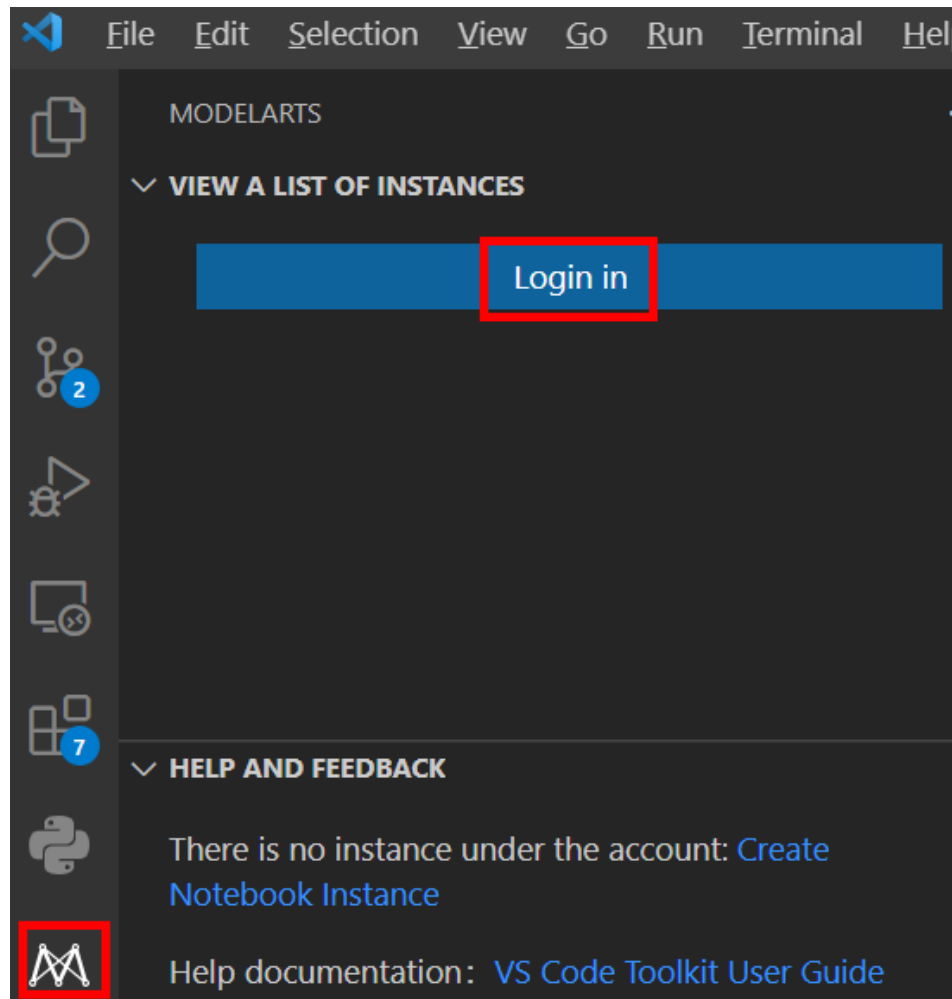
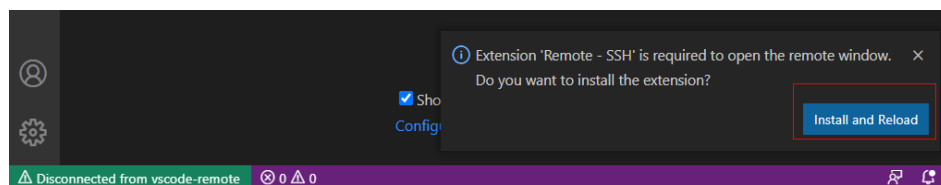


Figura 5-39 Instalación completada



Los problemas de red pueden provocar una falla en la instalación. Si esto ocurre, proceda con las operaciones de seguimiento. Después de realizar **1** en **Paso 4 Acceder a la instancia de notebook**, el sistema mostrará automáticamente un cuadro de diálogo que se muestra en la siguiente figura. En este caso, haga clic en **Install and Reload**.

Figura 5-40 Reconexión de SSH remoto



Paso 2 Iniciar sesión en el complemento de VS Code


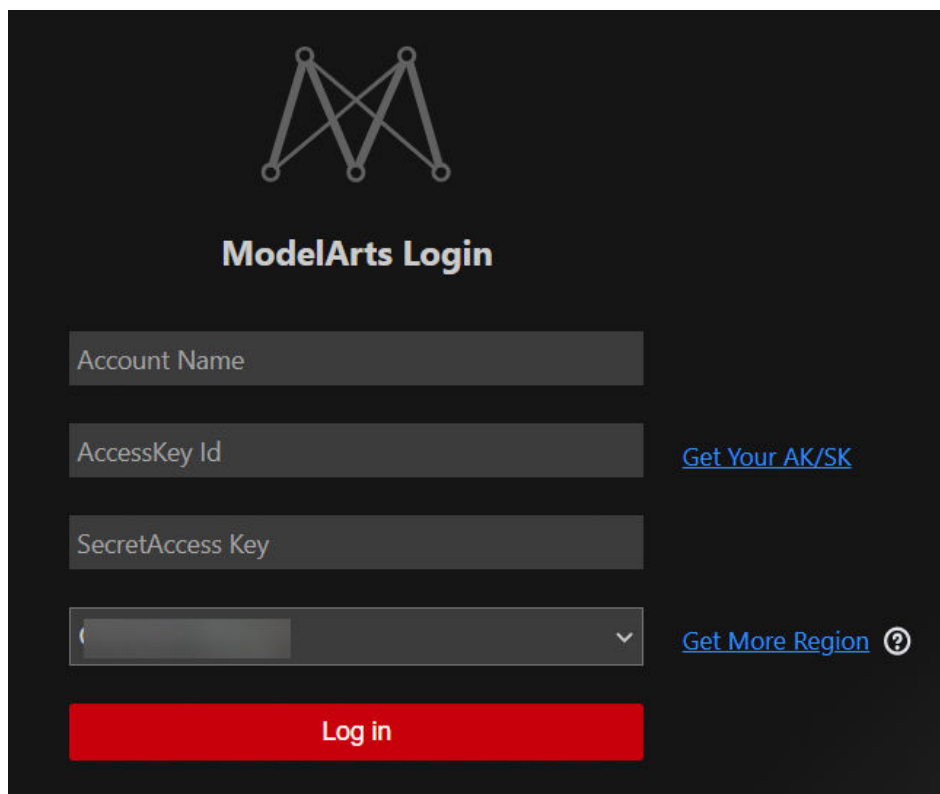
1. En el entorno de desarrollo local de VS Code, haga clic en  y **User Settings** y configure la información de inicio de sesión.

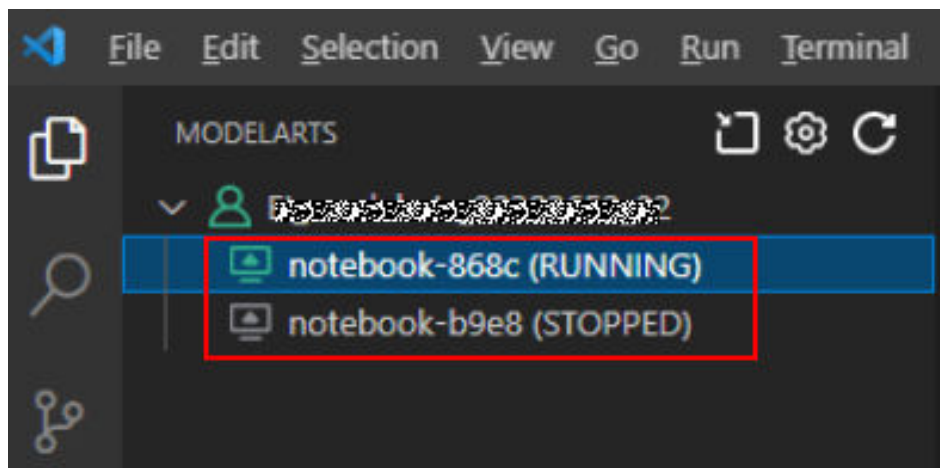
Figura 5-41 Iniciar sesión en el complemento



Ingrese la información de inicio de sesión y haga clic en **Log in**.

- **Name:** Nombre de usuario personalizado, que se muestra solo en la página de VS Code y no está asociado con ninguna cuenta de Huawei Cloud.
 - **AK y SK:** Par de claves de acceso. Para crear un par de claves, elija **My Credentials > API Credentials > Access Keys** y haga clic en **Create Access Key**.
 - **Region:** debe ser la misma que la de la instancia de notebook que se va a conectar de forma remota. De lo contrario, la conexión fallará.
2. Después del inicio de sesión, compruebe la lista de instancias del notebook.

Figura 5-42 Se ha iniciado sesión



Paso 3 Crear una instancia de notebook

⚠ ATENCIÓN

- Cree una instancia de notebook con SSH remoto habilitado y descargue el archivo de clave en cualquiera de los siguientes directorios basados en su SO:
Windows: `C:\Users\{{user}}`
macOS o Linux: `Users/{{user}}`
- Un par de claves se descarga automáticamente después de crearlo. Guárdelo de forma segura. Si se pierde un par de claves existente, cree uno nuevo.

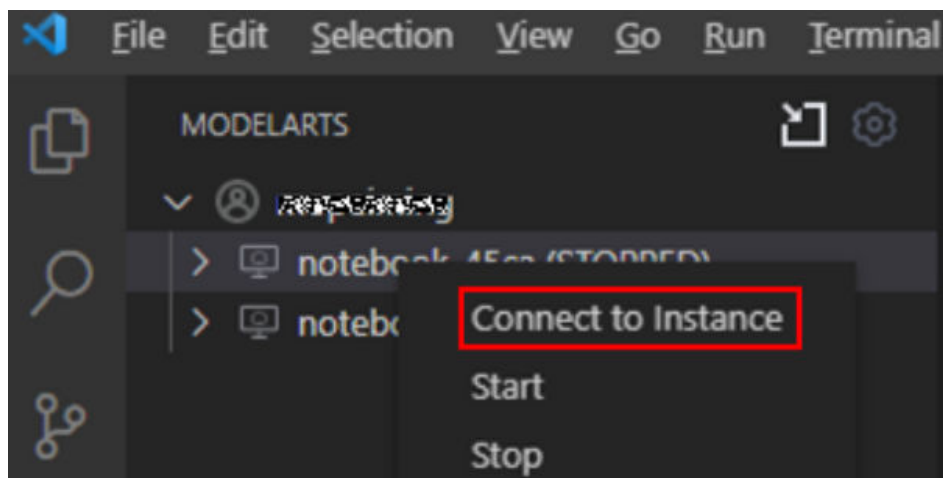
Cree una instancia de notebook con SSH remoto habilitado. Para obtener más detalles, véase [Creación de una instancia de notebook](#).

Paso 4 Acceder a la instancia de notebook

1. En el entorno de desarrollo local de VS Code, haga clic con el botón secundario del mouse ratón en el nombre de la instancia y elija **Connect to Instance** en el menú contextual para iniciar y conectarse a la instancia del notebook.

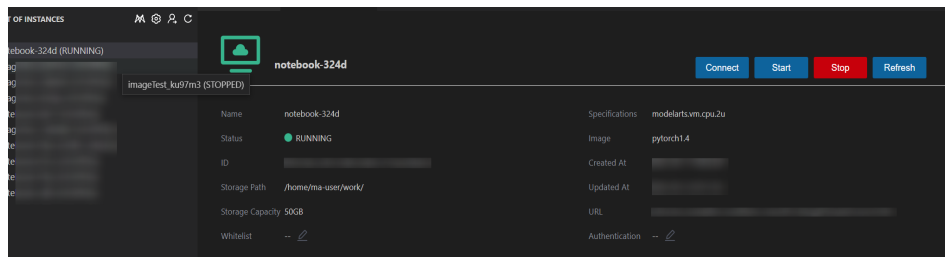
La instancia del notebook puede estar ejecutándose o parada. Si se detiene, el complemento de VS Code inicia la instancia y luego se conecta a ella.

Figura 5-43 Conexión a una instancia de notebook



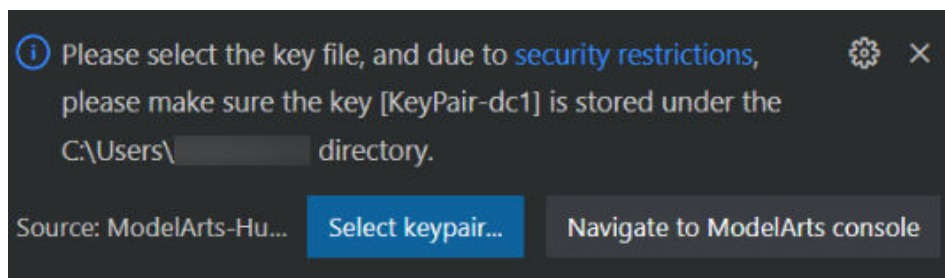
También puede hacer clic en el nombre de la instancia. En la página de detalles de la instancia, haga clic en **Connect**. A continuación, el sistema se inicia automáticamente y se conecta a la instancia del notebook.

Figura 5-44 Consulta de detalles sobre una instancia de notebook



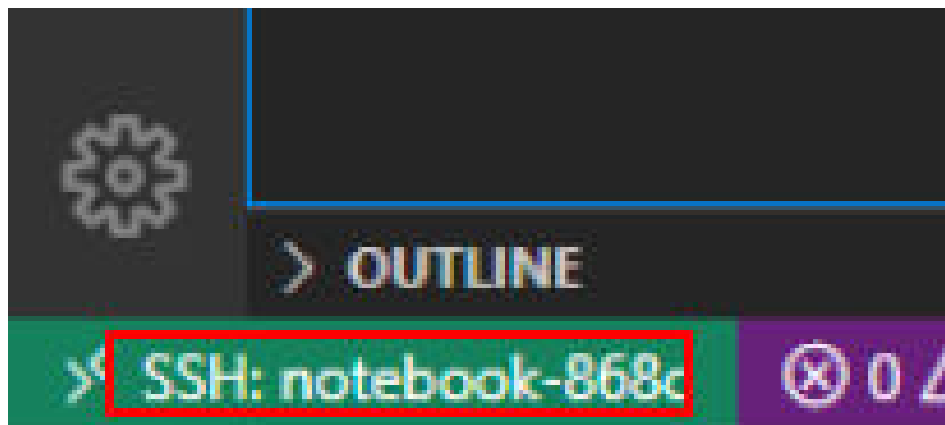
2. Cuando se conecta a una instancia de notebook por primera vez, el sistema le pedirá en la esquina inferior derecha que configure el archivo de clave. En este caso, seleccione el archivo de clave local .pem y haga clic en **OK**.

Figura 5-45 Configuración del archivo de clave



3. Espere de 1 a 2 minutos aproximadamente hasta que se acceda a la instancia del notebook. Después de que se muestre información similar a la siguiente en la esquina inferior izquierda del entorno de VS Code, la conexión se realiza correctamente.

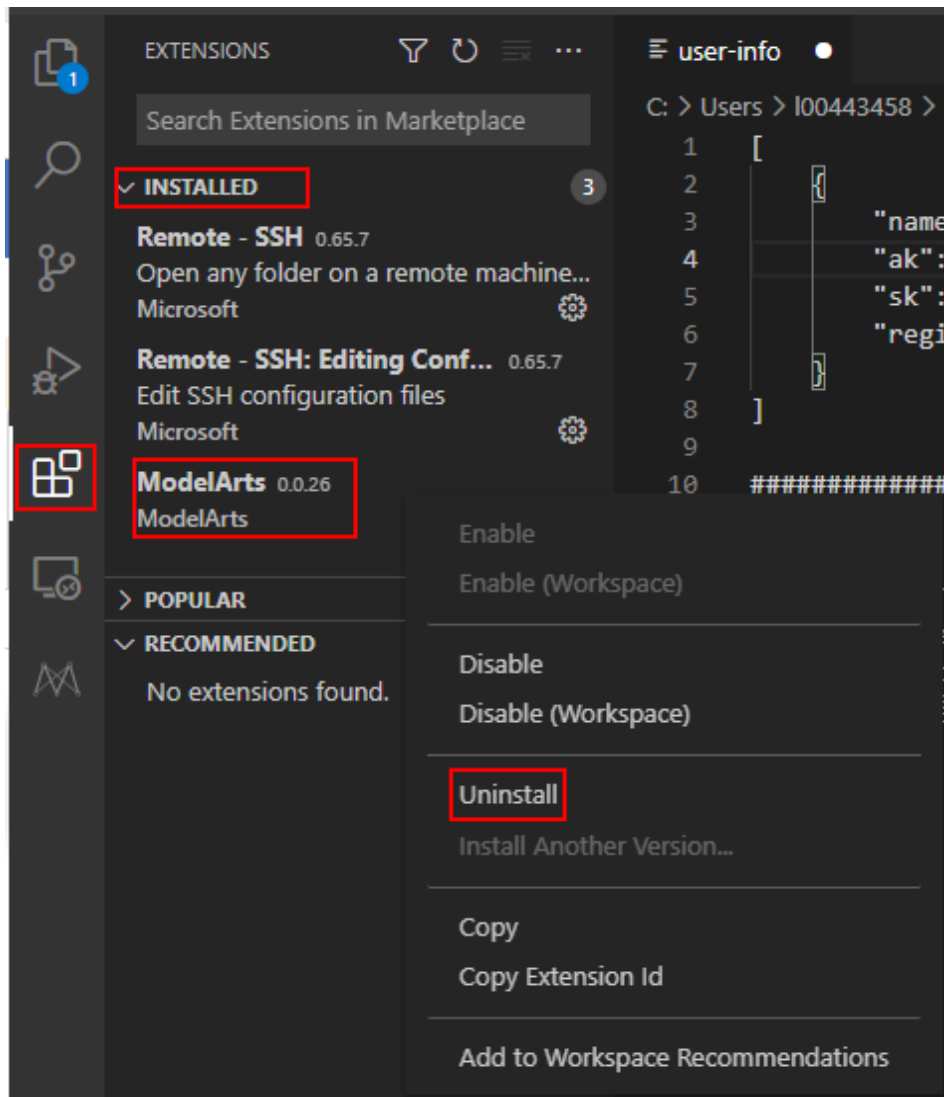
Figura 5-46 Conexión correcta



Operaciones relacionadas

Para obtener más información sobre cómo desinstalar el complemento de VS Code, consulte [Figura 5-47](#).

Figura 5-47 Desinstalar el complemento de VS Code



5.3.4 Conexión manual a una instancia de notebook por VS Code

Un IDE local admite PyCharm y VS Code. Puede utilizar VS Code o PyCharm para conectar remotamente el IDE local a la instancia de notebook de destino de ModelArts para ejecutar y depurar el código.

En esta sección se describe cómo utilizar VS Code para acceder a una instancia de notebook.

Requisitos previos

- Ha descargado e instalado VS Code. Para obtener más detalles, véase [Instalación de VS Code](#).
- Python se ha instalado en su PC o servidor local. Para obtener más información, consulte la [documentación oficial de VS Code](#).
- Se ha creado una instancia de notebook con SSH remoto habilitado. Asegúrese de que la instancia se está ejecutando. Para obtener más detalles, véase [Creación de una instancia de notebook](#).

- La dirección y el número de puerto del entorno de desarrollo están disponibles. Para obtener la información, vaya a la página de detalles de la instancia del notebook.

Figura 5-48 Página de detalles de la Instancia

Address ssh://ma-user@lev-modelarts- [redacted] :80581
Authentication KeyPair-e744 Address Port number

- El par de claves está disponible.
Un par de claves se descarga automáticamente después de crearlo. Guárdelo de forma segura. Si se pierde un par de claves existente, cree uno nuevo.

Paso 1 Agregar el complemento Remote-SSH


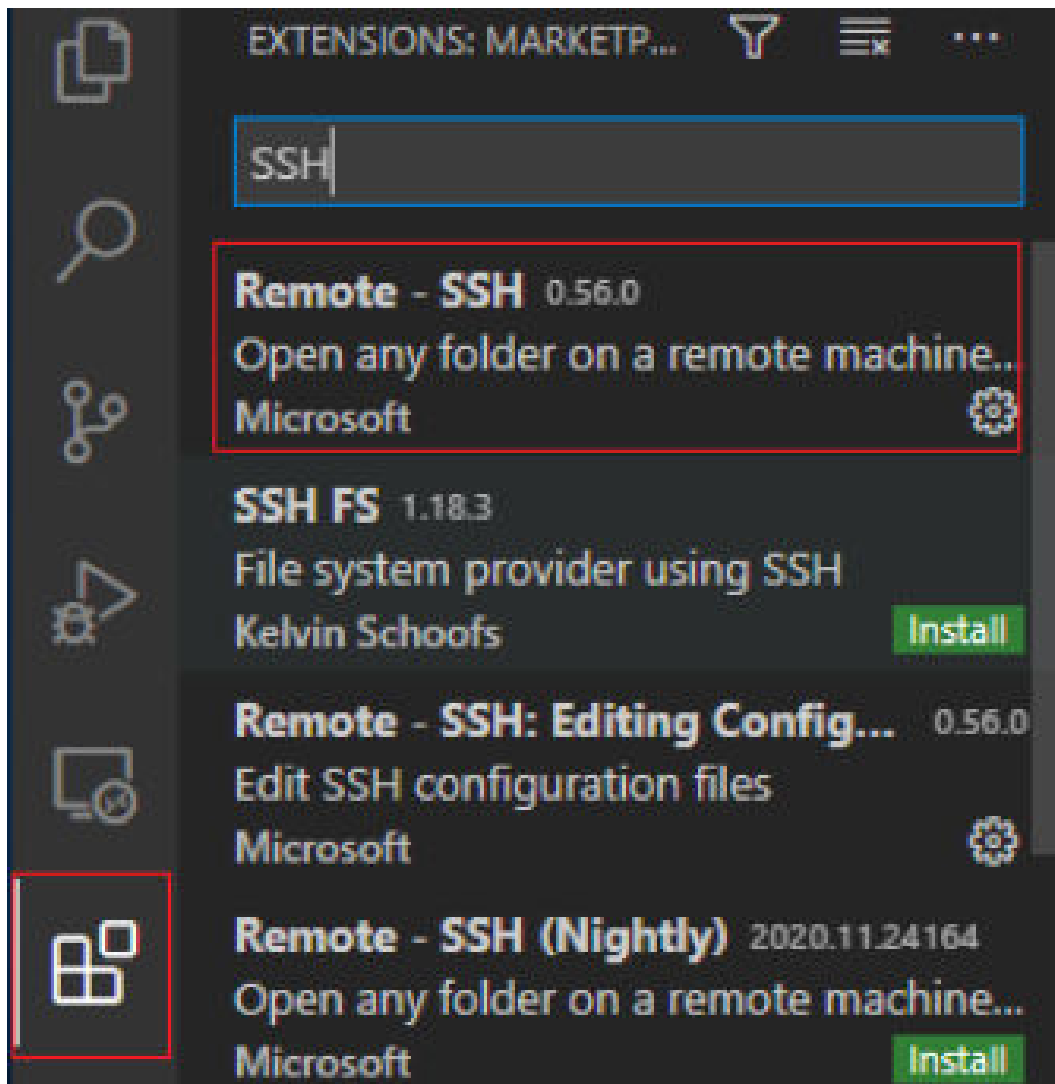
En el entorno de desarrollo local de VS Code, haga clic en  , ingrese **SSH** en el cuadro de búsqueda y haga clic en **install** del complemento Remote-SSH para instalarlo.

Figura 5-49 Adición del complemento Remote-SSH



Paso 2 Configurar SSH



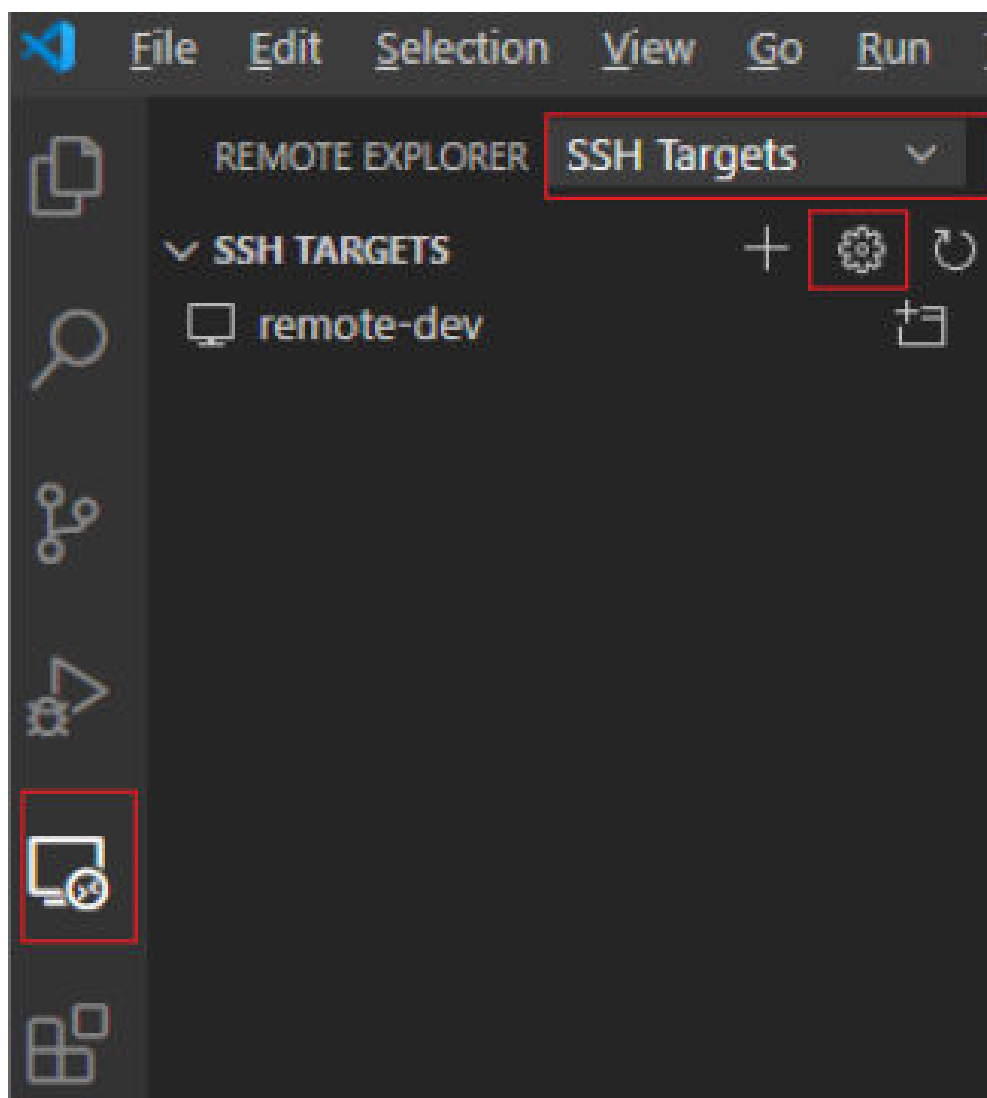
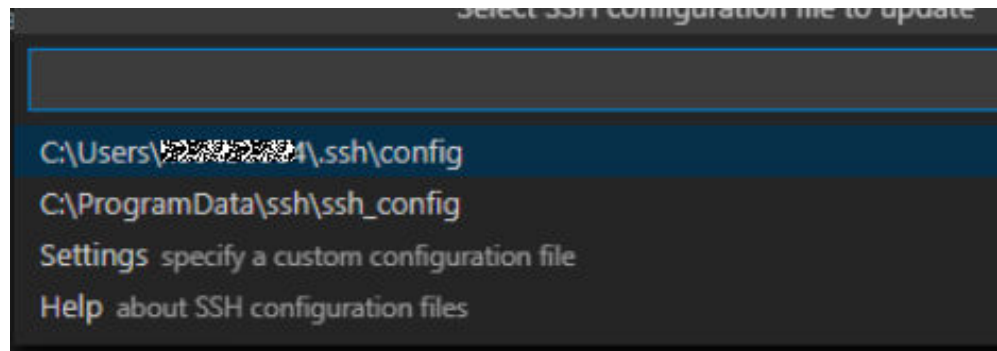
1. En el entorno de desarrollo local de VS Code, haga clic en  a la izquierda, seleccione **SSH Targets** en el cuadro de lista desplegable y haga clic en . Se muestra la ruta del archivo de configuración de SSH.

Figura 5-50 Configuración de destinos de SSH



2. Haga clic en la ruta de configuración de SSH y configure SSH.

Figura 5-51 Ruta del archivo de configuración de SSH

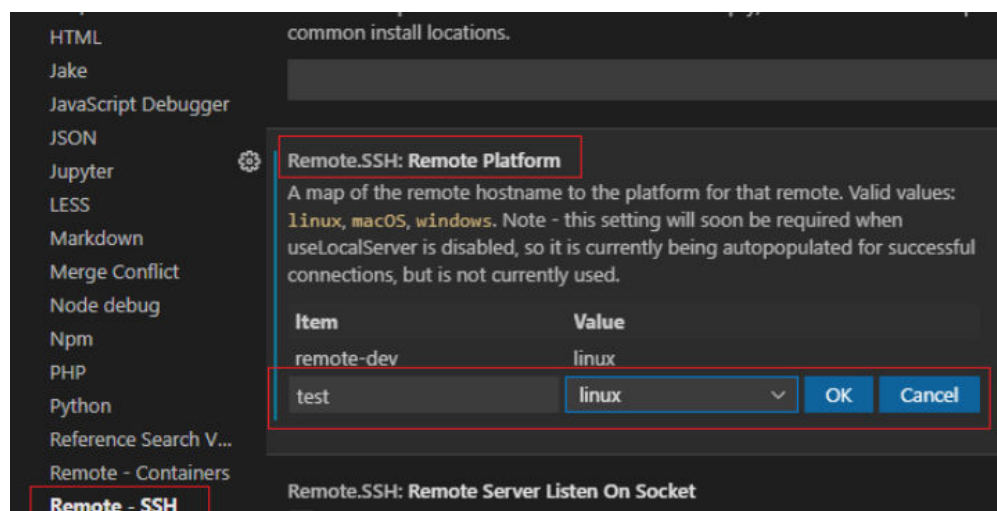


```
HOST remote-dev
hostname <Instance connection host>
port <Instance connection port>
user ma-user
IdentityFile ~/.ssh/test.pem
UserKnownHostsFile=/dev/null
StrictHostKeyChecking no
```

- **HOST**: nombre del entorno de desarrollo en la nube
- **HostName**: dirección para acceder al entorno de desarrollo en la nube. Obtenga la dirección en la página que proporciona información detallada de la instancia de destino del notebook.
- **Port**: número de puertos para acceder al entorno de desarrollo en la nube. Obtenga el número de puerto en la página que proporciona la información detallada de la instancia del notebook de destino.
- **user**: ma-user
- **IdentityFile**: archivo de clave privada almacenado localmente del entorno de desarrollo en la nube. Es el archivo de par de claves en [Requisitos previos](#).

3. Elija **File > Preference > Settings > Extensions > Remote-SSH**. En la página **Remote Platform**, haga clic en **Add Item**, configure **Item** y **Value** y luego haga clic en **OK**.

Figura 5-52 Configuración de la plataforma remota



Item: nombre de host configurado en la configuración de SSH

Value: plataforma de entorno de desarrollo remoto


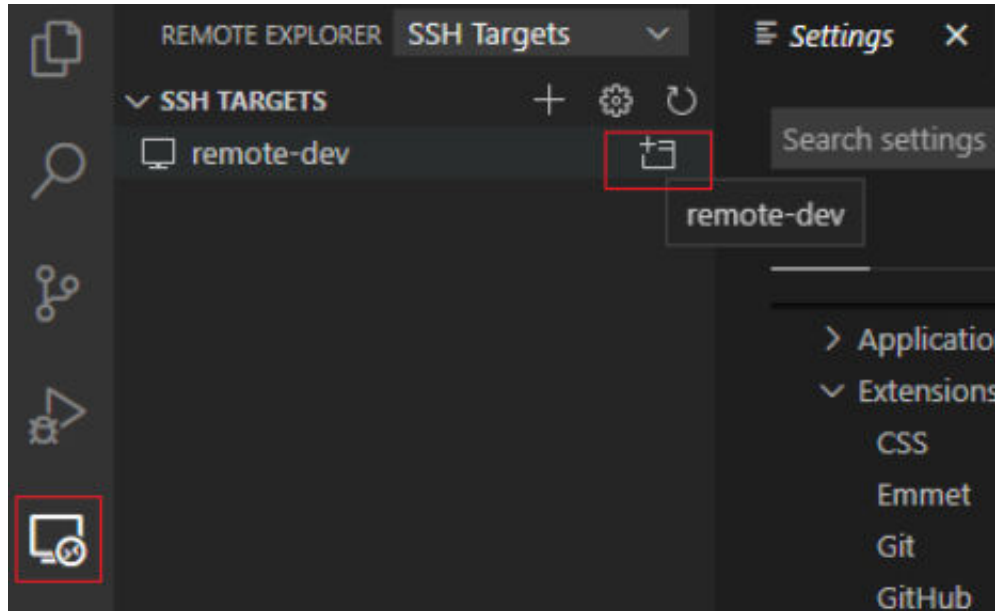
4. Vuelva a la página **SSH Targets** y haga clic en  a la derecha. A continuación, haga clic en el nombre del entorno de desarrollo para abrirlo.

Figura 5-53 Apertura del entorno de desarrollo



Después de mostrarse la página siguiente, la conexión se realiza correctamente.

Figura 5-54 Conexión remota correcta

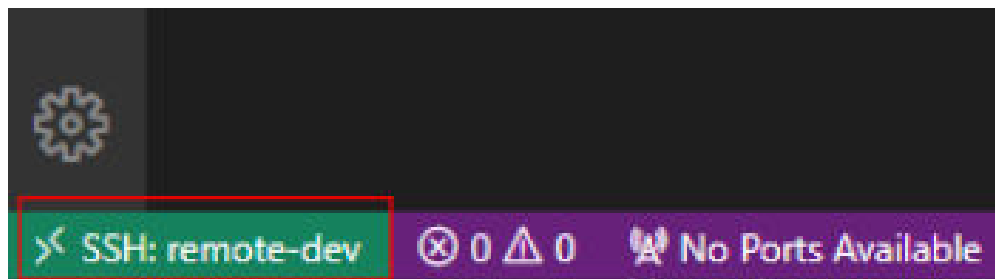
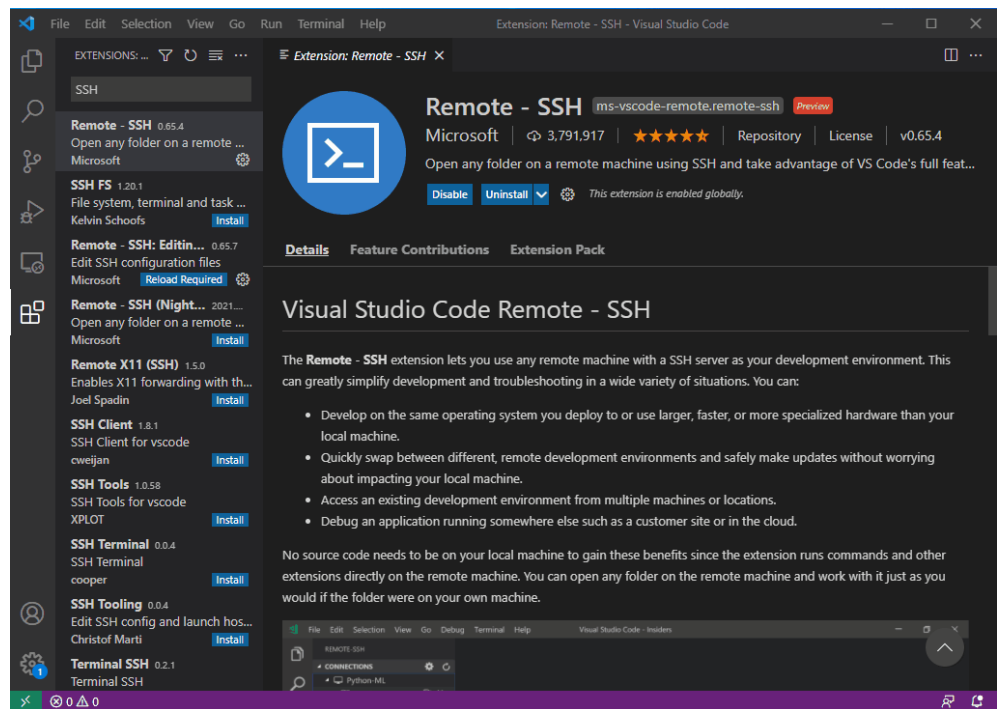


Figura 5-55 Ejemplo de configuración completo



Paso 3 Instale el complemento de Python en el entorno de desarrollo en la nube


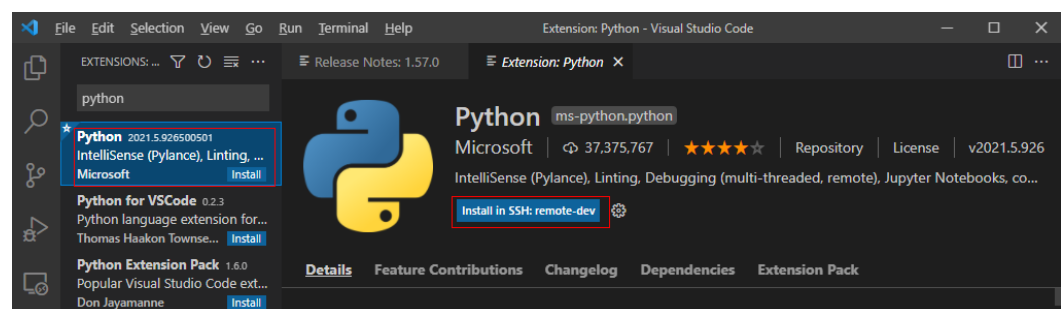
En la página de VS Code que aparece, haga clic en  a la izquierda, escriba **Python** en el cuadro de búsqueda y haga clic en **Install**.

Figura 5-56 Instalación del complemento de Python en el entorno de desarrollo en la nube



Si el complemento de Python no se instala en la nube, instálelo usando un paquete sin conexión.

Paso 4 Instalar la biblioteca dependiente para el entorno de nube

Después de acceder al entorno de contenedores, puede utilizar diferentes entornos virtuales, como TensorFlow y PyTorch. Sin embargo, en el desarrollo real, necesita instalar los paquetes de dependencias. A continuación, puede acceder al entorno a través de la terminal para realizar operaciones.

1. En VS Code, pulse **Ctrl+Shift+P**.

2. Encuentre **Python: Select Interpreter** y seleccione el Python de destino.
3. Elija **Terminal > New Terminal**. Se muestra CLI del contenedor remoto.
4. Ejecute el siguiente comando para instalar el paquete de dependencias:

```
pip install spacy
```

5.3.5 Depuración remota en VS Code

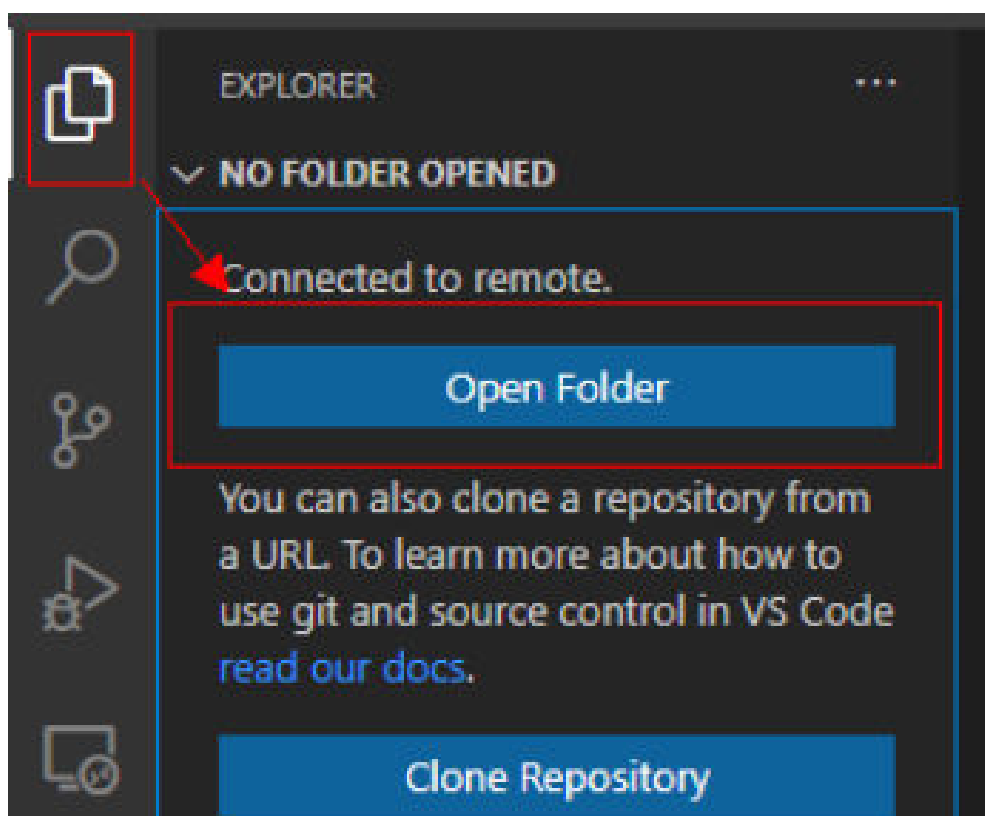
Requisitos previos

Se ha accedido a una instancia de notebook a través de VS Code.

Paso 1 Subir el código local al entorno de desarrollo de la nube

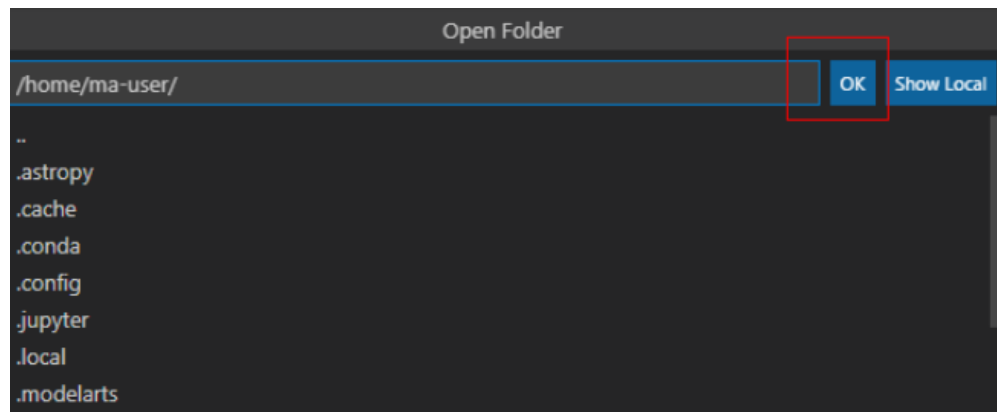
1. En la página VS Code, elija **File > Open Folder** para acceder a la ruta de acceso a la nube.

Figura 5-57 Abrir carpeta



2. Seleccione una ruta y haga clic en **OK**.

Figura 5-58 Selección de una ruta de archivo

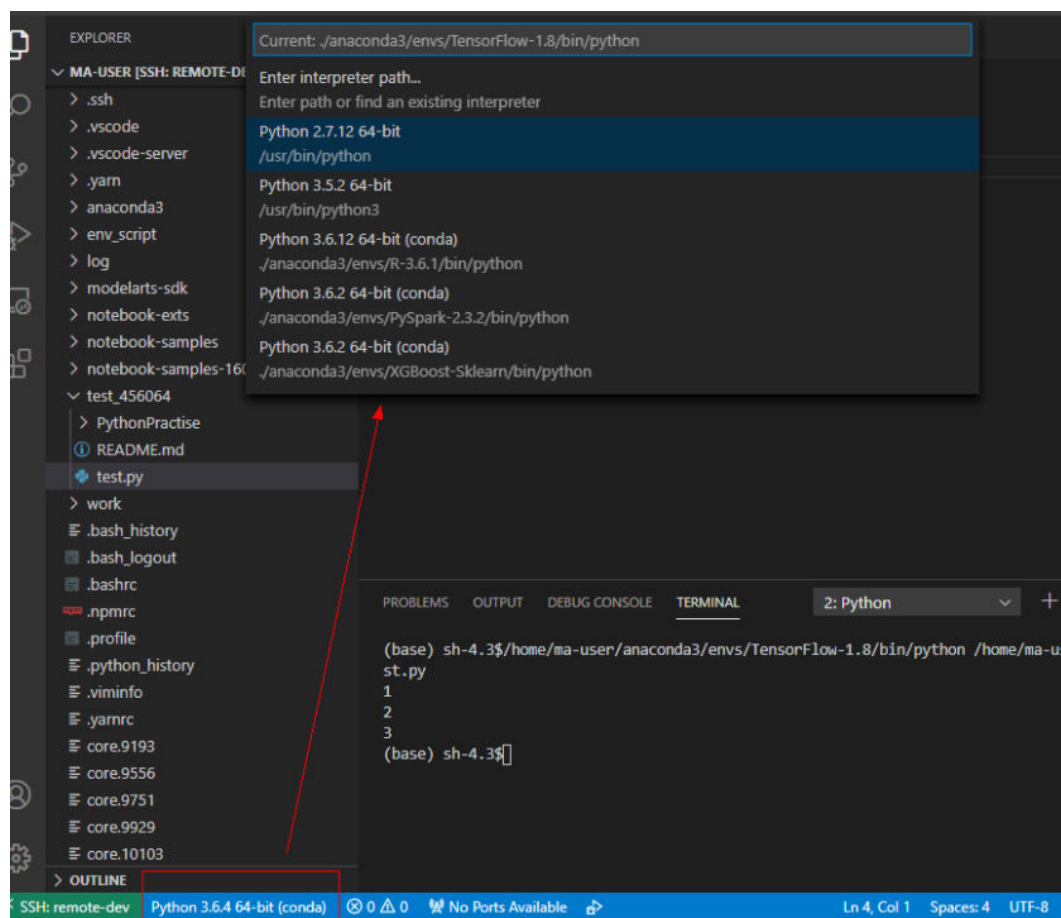


3. En la estructura de directorios mostrada a la izquierda del IDE, arrastre el código y los archivos que desea cargar a las carpetas correspondientes. A continuación, el código se carga en el entorno de desarrollo de la nube.

Paso 2 Depurar el código de forma remota

Abra el archivo de código que se va a depurar en VS Code. Antes de ejecutar el código, haga clic en la versión predeterminada de Python en la parte inferior izquierda y seleccione una versión según sea necesario.

Figura 5-59 Selección de una versión de Python



- Haga clic en el botón de ejecución para ejecutar el código. La salida del código se muestra en la pestaña **TERMINAL**.
- Si un trabajo de entrenamiento tarda mucho en ejecutarse, ejecute el trabajo en el backend a través del comando nohup. Esto evita que la desconexión de una sesión SSH o un error de red afecte a la ejecución del trabajo. A continuación se muestra un ejemplo de comando nohup:

```
nohup your_train_job.sh > output.log 2>&1 & tail -f output.log
```
- Para depurar el código, realice las siguientes operaciones:
 - a. Elija **Run > Run and Debug** a la izquierda.
 - b. Seleccione el archivo de código predeterminado de Python.
 - c. Haga clic a la izquierda del código para establecer puntos de interrupción.
 - d. Depure el código de acuerdo con el procedimiento de depuración que se muestra encima del código, y la información de depuración se muestra a la izquierda de la página.

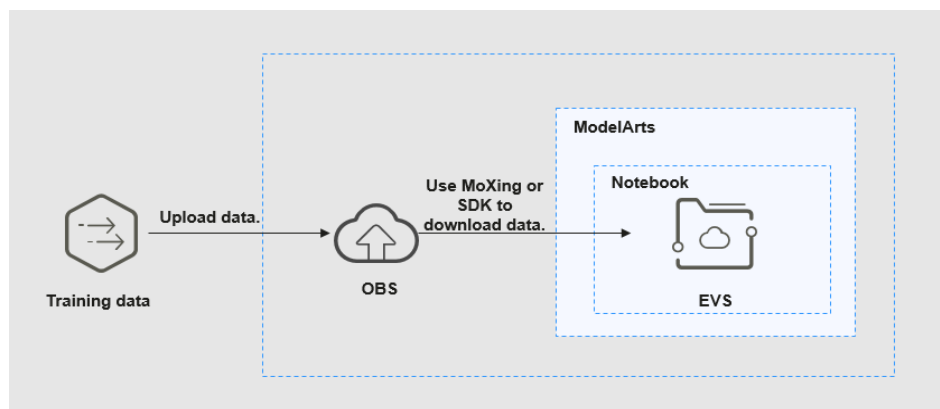
5.3.6 Carga y descarga de archivos en VS Code

Carga de datos desde un IDE local a una instancia de notebook

Si los datos son menores o iguales a 500 MB, copie directamente los datos en el IDE local.

Si los datos superan los 500 MB, cárguelos en OBS y luego en la instancia de notebook.

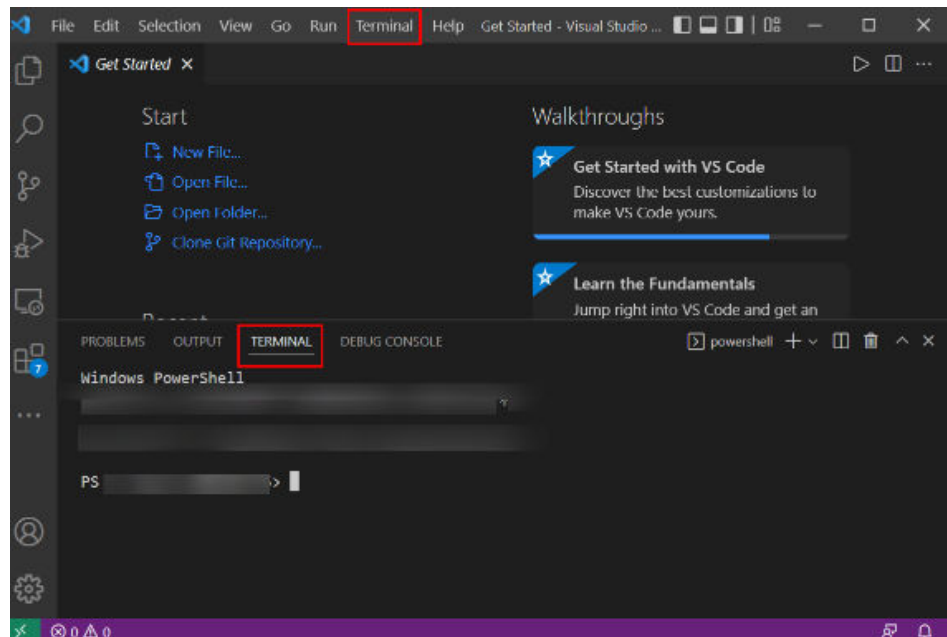
Figura 5-60 Carga de datos a una instancia de notebook a través de OBS



Procedimiento

1. Cargue los datos en OBS. Para obtener más información, véase [Carga de un objeto](#). Alternativamente, utilice el SDK de ModelArts en un terminal de VS Code local. Abra el terminal en el entorno local de VS Code.

Figura 5-61 Apertura del terminal en el entorno local de VS Code



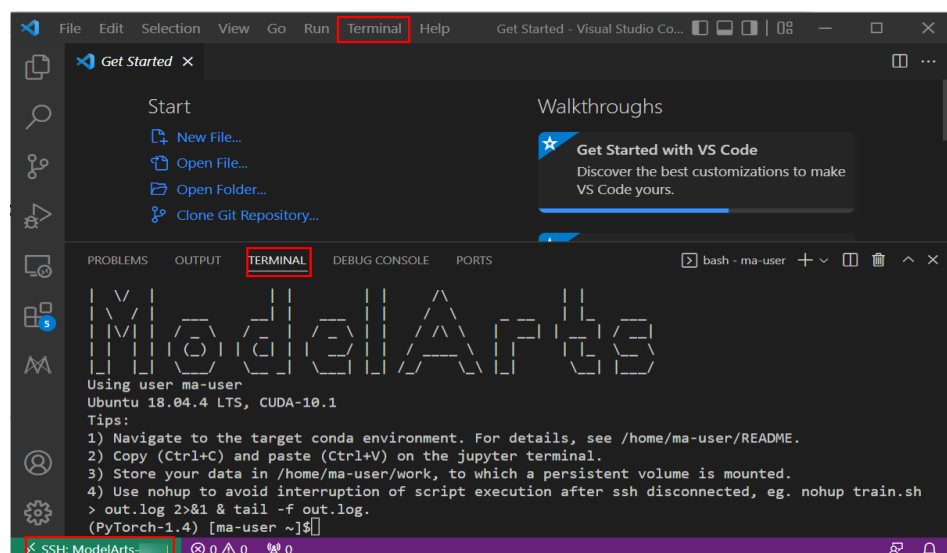
Ingrese **python** y presione **Enter** para acceder al entorno de Python.

```
python
```

En el terminal del VS Code local, utilice el SDK de ModelArts para cargar el archivo local de destino en OBS. Para obtener más detalles, véase [Transferencia de archivos](#).

2. Utilice el SDK de ModelArts en el terminal del entorno remoto de VS Code para descargar el archivo desde OBS a un entorno de desarrollo.

Figura 5-62 Apertura del terminal en el entorno remoto de VS Code



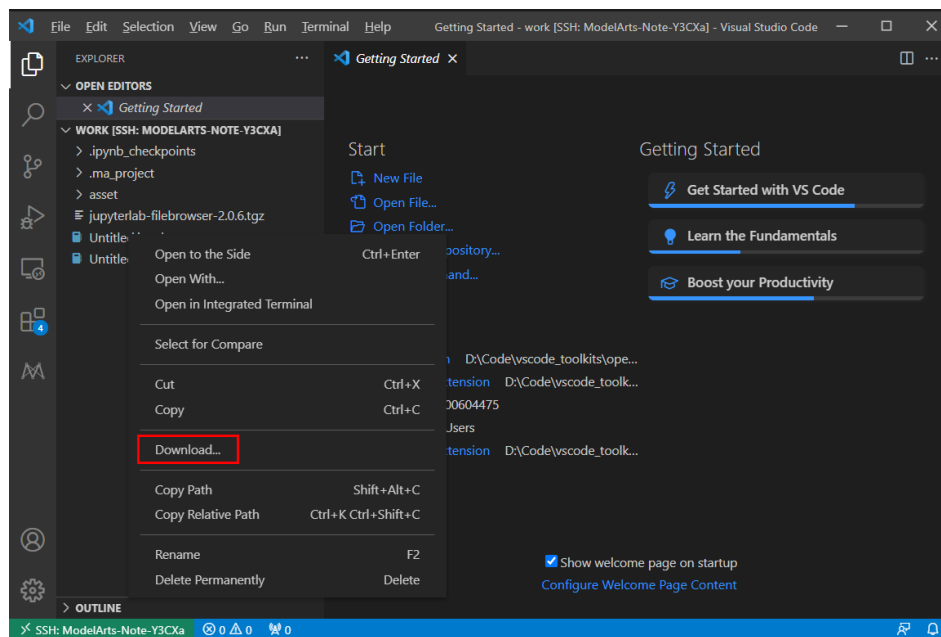
```
# Manually access the development environment using the source command.  
cat /home/ma-user/README  
# Select the target environment.  
source /home/ma-user/miniconda3/bin/activate MindSpore-python3.7-aarch64  
# Enter python and press Enter to access the Python environment.  
python
```


Luego, realice operaciones de transferencia de OBS consultando [Carga de un archivo en OBS](#).

Descarga de archivos de una instancia de notebook en un directorio local

Los archivos creados en notebook se pueden descargar en una ruta local. En el directorio **Project** del IDE local, haga clic con el botón derecho en el proyecto **Notebook2.0** y elija **Download** en el menú contextual para descargar el archivo del proyecto en el PC local.

Figura 5-63 Descargar archivos desde una instancia de notebook a un directorio local en VS Code



5.4 IDE local (acceso con SSH)

En esta sección se describe cómo utilizar PuTTY para iniciar sesión de forma remota en una instancia de notebook en la nube en el entorno de Windows.

Requisitos previos

- Ha creado una instancia de notebook con SSH remoto habilitado y la lista blanca configurada. Asegúrese de que la instancia se está ejecutando. Para obtener más detalles, véase [Creación de una instancia de notebook](#).
- La dirección y el número de puerto del entorno de desarrollo están disponibles. Para obtener esta información, vaya a la página de detalles de la instancia del notebook.

Figura 5-64 Página de detalles de la Instancia

Address	ssh://ma-user@dev-modelarts- [redacted] .com	32651
Authentication	KeyPair-9a64	Access address of the development environment Port number

- El par de claves está disponible.
Un par de claves se descarga automáticamente después de crearlo. Guárdelo de forma segura. Si se pierde un par de claves existente, cree uno nuevo.

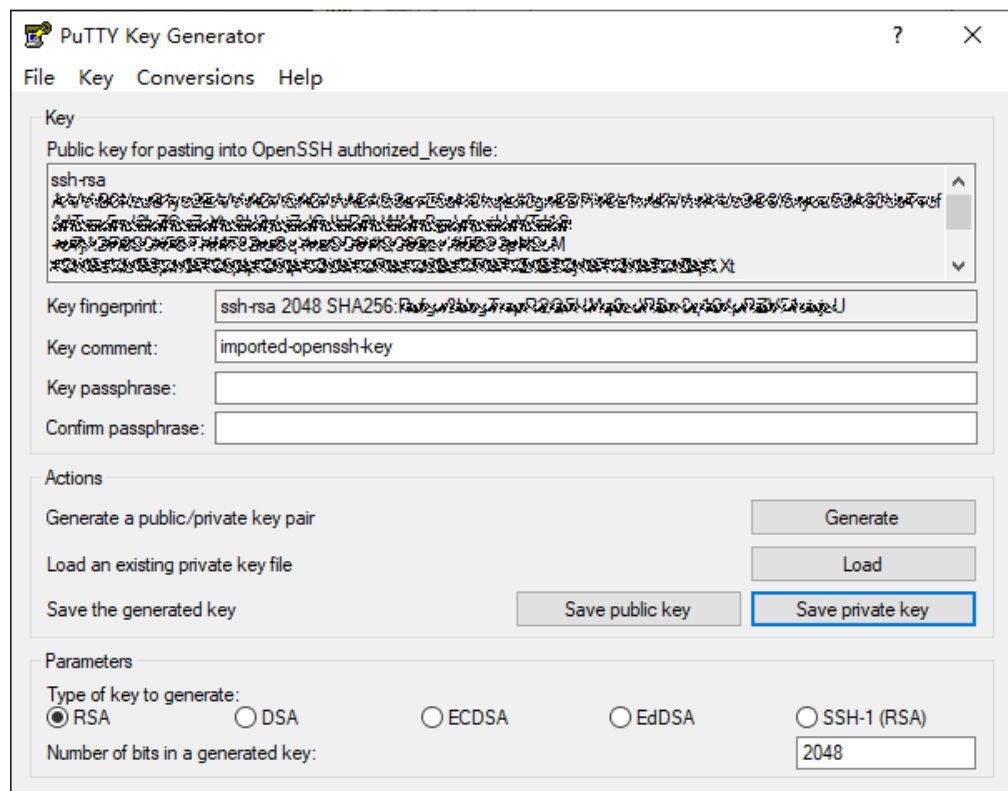
Paso 1 Instalar la herramienta de SSH

[Descargue](#) e instale la herramienta de acceso remoto de SSH, por ejemplo, PuTTY.

Paso 2 Utilizar PuTTYgen para convertir el archivo de par de claves .pem en un archivo de par de claves .ppk

1. [Descargue PuTTYgen](#) y haga doble clic en él para ejecutarlo.
2. Haga clic en **Load** para cargar el archivo de claves .pem creado y guardado durante la creación de la instancia de notebook.
3. Haga clic en **Save private key** para guardar el archivo .ppk generado. El nombre del archivo se puede personalizar, por ejemplo, **key.ppk**.

Figura 5-65 Conversión del archivo de par de claves .pem en un archivo de par de claves .ppk

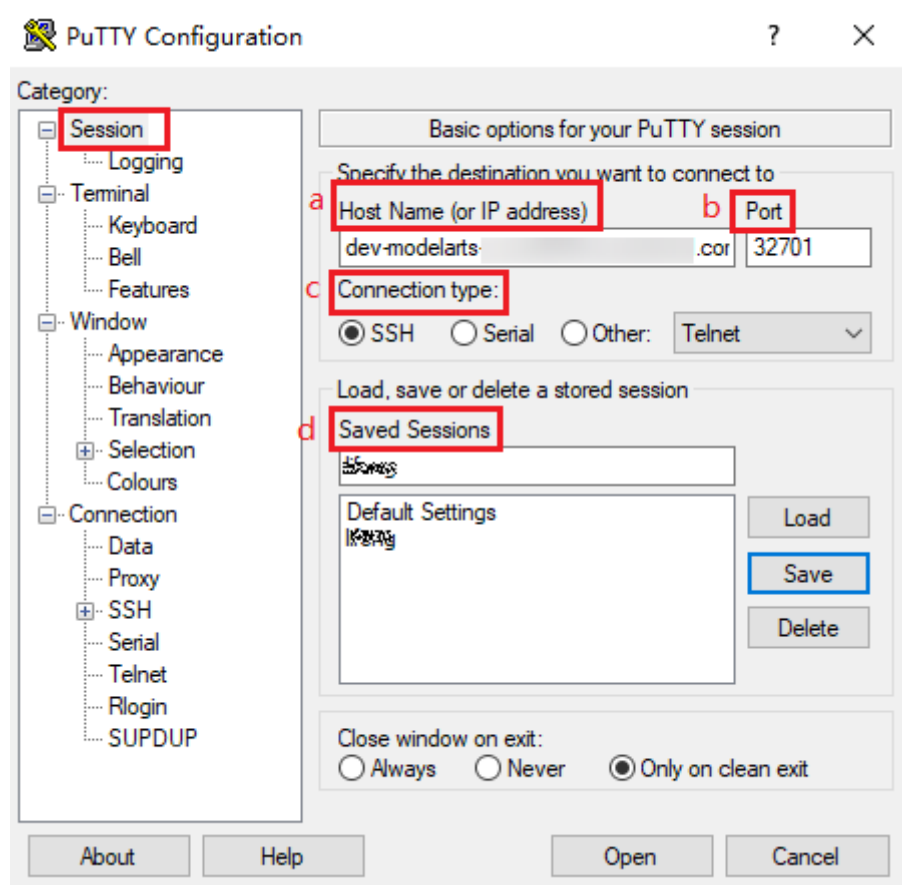


Paso 3 Utilizar SSH para conectarse a una instancia de notebook

1. Ejecute PuTTY.
2. Haga clic en **Session** y configure los siguientes parámetros:
 - a. **Host Name (or IP address)**: dirección para acceder a la instancia de notebook en la nube. Obtenga la dirección en la página que proporciona la información detallada de la instancia del notebook de destino.

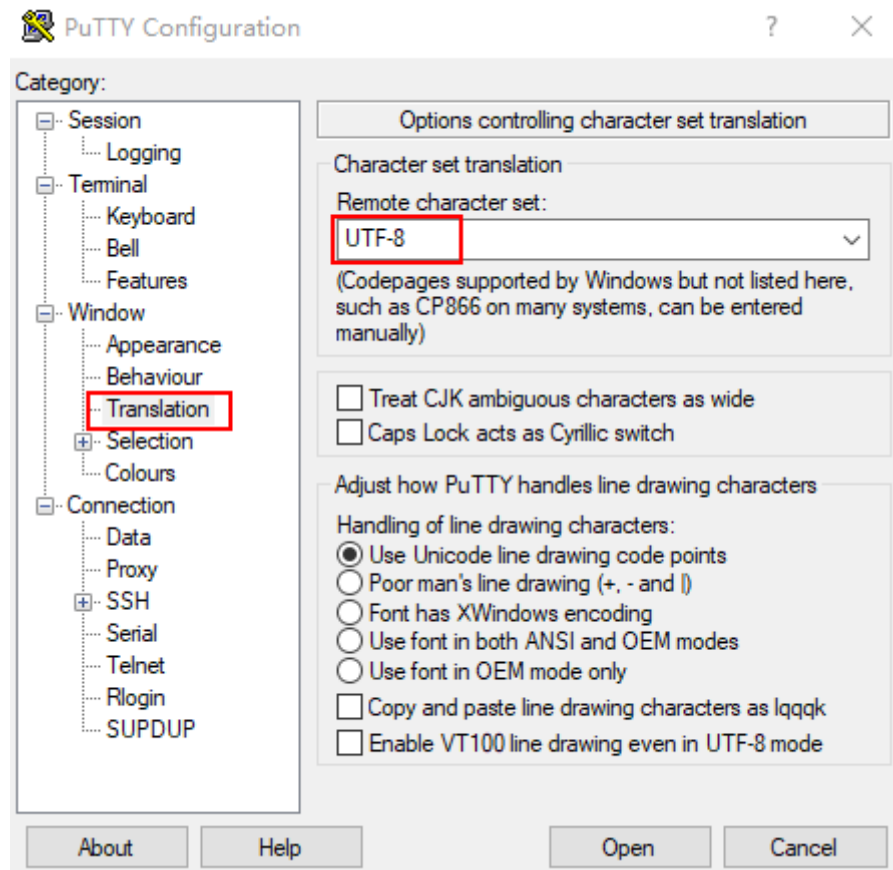
- b. **Port:** número de puerto para acceder a la instancia del notebook en la nube. Obtenga el número de puerto en la página que proporciona información detallada de la instancia de notebook de destino, por ejemplo, **32701**.
- c. **Connection type:** **SSH**
- d. **Saved Sessions:** nombre de la tarea, en el que se puede hacer clic para acceder de forma remota la próxima vez que utilice PuTTY

Figura 5-66 Configuración de Session



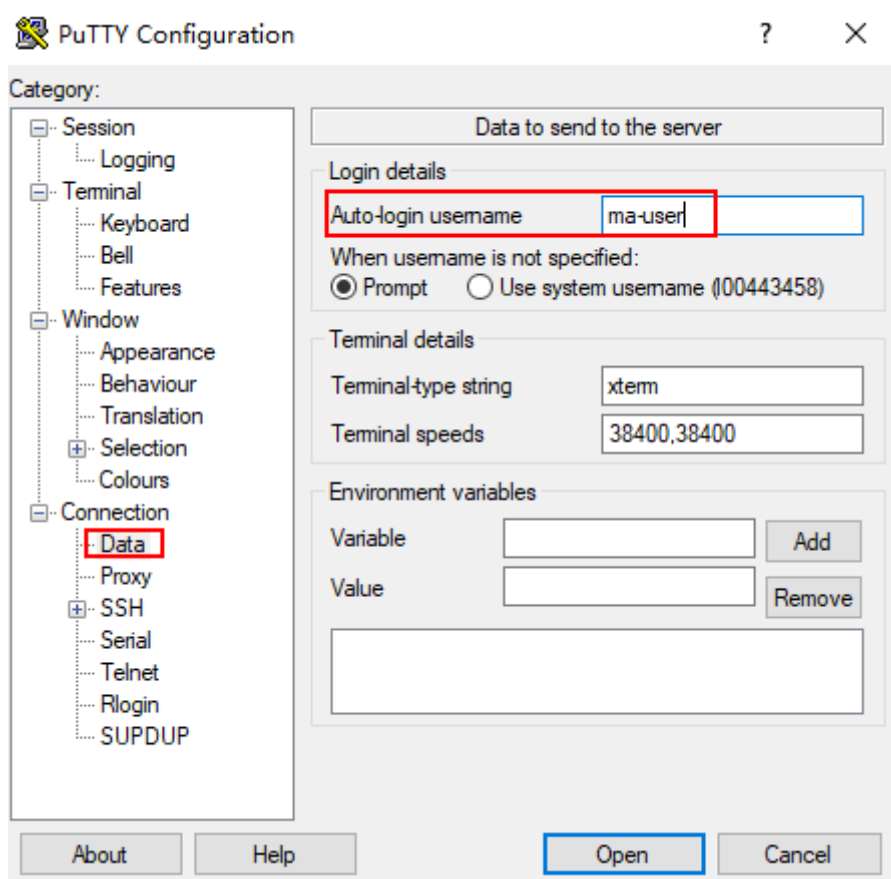
- 3. Seleccione **Window > Translation** y seleccione **UTF-8** en el cuadro de lista desplegable del área **Remote character set**.

Figura 5-67 Configuración del formato de carácter

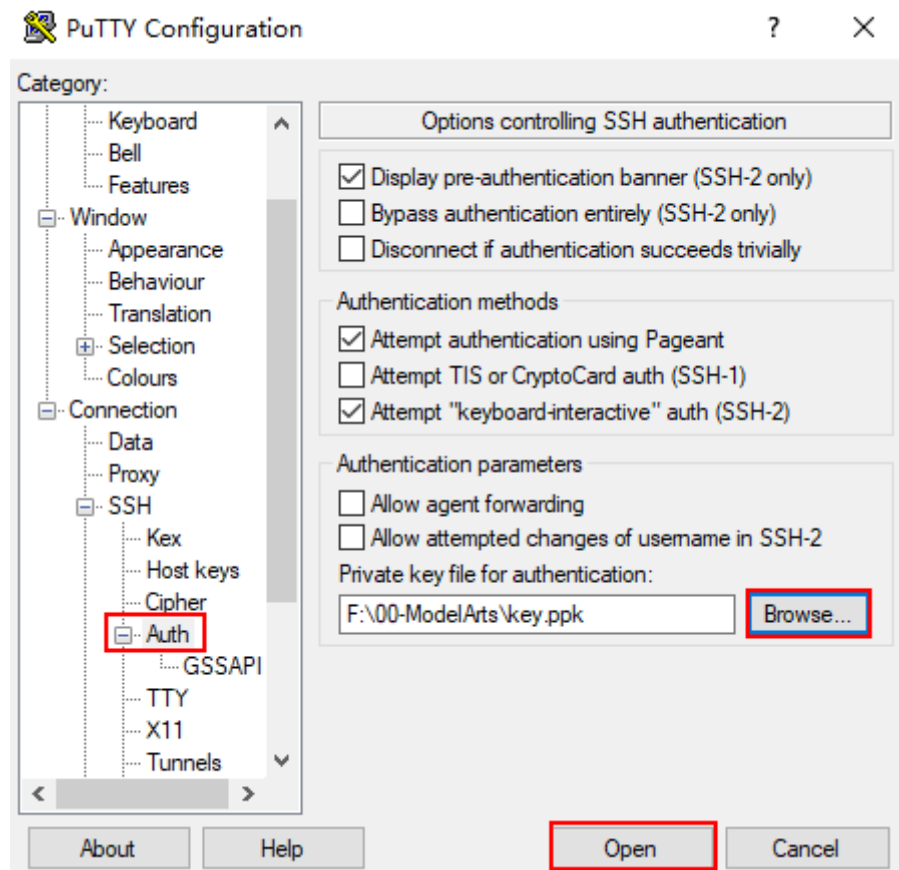


4. Seleccione **Connection > Data** e introduzca **ma-user** para **Auto-login username**.

Figura 5-68 Introducir un nombre de usuario

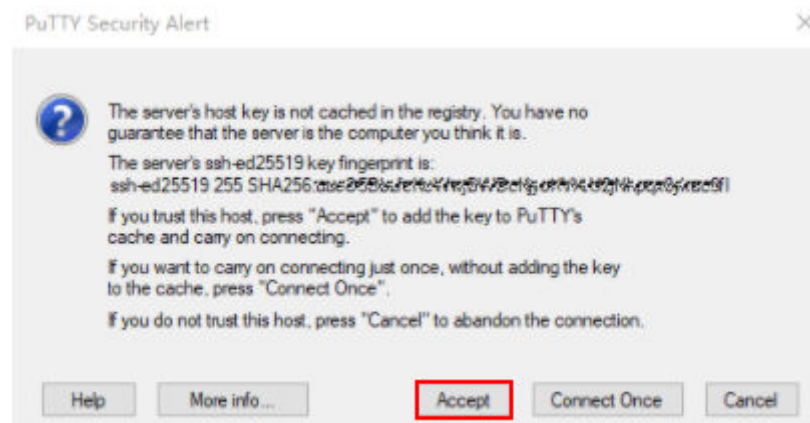


5. Seleccione **Connection** > **SSH** > **Auth**, haga clic en **Browse** y seleccione el archivo .ppk generado en el [paso 2](#).



6. Haga clic en **Open**. Si inicia sesión en la instancia por primera vez, PuTTY muestra un cuadro de diálogo de advertencia de seguridad que le pregunta si desea aceptar el certificado de seguridad de la instancia. Haga clic en **Accept** para guardar el certificado en el registro local.

Figura 5-69 Preguntar si desea aceptar el certificado de seguridad de instancia



7. Conéctese a la instancia de notebook.

Figura 5-70 Conexión a una instancia de notebook

```
Using username "ma-user".
Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 3.10.0-862.14.1.5.h328.eulerosv2r7.x86_64 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Thu Aug 12 15:10:57 2021 from 192.168.1.100
sh-4.4$
```

6 Referencia de comandos de ModelArts CLI

6.1 Descripción de ModelArts CLI

Descripción

ModelArts CLI, también llamada ma-cli, es una herramienta de línea de comandos entre las plataformas que se utiliza para conectarse a ModelArts y ejecutar comandos de gestión en recursos de ModelArts. Puede utilizar el símbolo del sistema interactivo o la secuencia de comandos para ejecutar comandos en una terminal. ma-cli le permite interactuar con servicios en la nube con notebook de ModelArts y máquinas virtuales locales. Puede ejecutar comandos de ma-cli para autocompletar y autenticar comandos, así como crear imágenes, enviar trabajos de entrenamiento de ModelArts y DLI Spark, y copiar datos de OBS.

Escenarios de aplicación

- ma-cli se ha integrado en notebook de ModelArts y puede utilizarse directamente. Inicie sesión en la consola de ModelArts, elija **DevEnviron > Notebook**, cree una instancia de notebook, inicie un terminal y ejecute comandos de ma-cli.
- En Windows o Linux locales, instale ma-cli y luego utilícelo en un terminal local. Para obtener más detalles, véase [\(Opcional\) Instalación de ma-cli localmente](#).

NOTA

- ma-cli no se puede utilizar en Git Bash.
- Se recomienda utilizar terminales como Linux Bash, Zsh, Fish, WSL y PowerShell. Para garantizar la seguridad de su información confidencial, es importante evitar cualquier posible fuga al utilizar terminales.

Vista previa del comando

```
$ ma-cli -h
Usage: ma-cli [OPTIONS] COMMAND [ARGS]...

Options:
  -V, -v, --version           1.2.1
  -C, --config-file TEXT      Configure a file path for authorization.
  -D, --debug                  Debugging mode, in which the full stack trace will be
displayed when an error occurs.
  -P, --profile TEXT          CLI connection profile to be used. The default profile
```



```
is DEFAULT.
-h, -H, --help          Show the help information and exit.

Commands:
  configure              Configure authentication and endpoints for the CLI.
  image                  Obtain registered images, register or unregister images, debug
images, and create images in Notebook.
  obs-copy               Copy files or directories between OBS and a local path.
  ma-job                 Submit ModelArts jobs and obtain job details.
  dli-job                Submit DLI spark jobs and obtain job details.
  auto-completion       Auto complete ma-cli command in terminal, support
"bash(default)/zsh/fish".
```

Entre los parámetros anteriores, los parámetros **-C**, **-D**, **-P** y **-h** son globalmente opcionales.

- **-C** indica que puede especificar manualmente el archivo de configuración de autenticación al ejecutar este comando. De forma predeterminada, se utiliza el archivo de configuración `~/.modelarts/ma-cli-profile.yaml`.
- **-P** indica un grupo de información de autenticación en el archivo de autenticación. El valor predeterminado es **DEFAULT**.
- **-D** indica si se debe habilitar el modo de depuración (deshabilitado de forma predeterminada). Una vez habilitado el modo de depuración, se imprimirá la información de pila de errores del comando. Si este modo está desactivado, solo se imprimirá la información del error.
- **-h** indica que se mostrará la información de ayuda sobre el comando.

Comandos

Tabla 6-1 Comandos de ma-cli

Comando	Descripción
configure	Autenticación de ma-cli utilizando un nombre de usuario y una contraseña o una AK/SK
image	Creación, registro y consulta de imágenes registradas de ModelArts
obs-copy	Copiar archivos o carpetas entre una ruta local y OBS
ma-job	Gestión de trabajos de entrenamiento de ModelArts, incluido el envío de trabajos y la consulta de recursos
dli-job	Envío de trabajos de DLI Spark y gestión de recursos
auto-completion	Completar automáticamente el comando

6.2 (Opcional) Instalación de ma-cli localmente

Escenarios de aplicación

Este documento describe cómo instalar ma-cli en Windows.

Paso 1: Instalar los SDK de ModelArts

Instale los SDK de ModelArts consultando [Instalación local de SDK de ModelArts](#).

Paso 2: Descargar ma-cli

1. [Descargue el paquete de software de ma-cli](#).
2. Verifique la firma del paquete de software.
 - a. [Descargue el archivo de verificación de firmas del paquete de software](#).
 - b. Instale OpenSSL y ejecute el siguiente comando para verificar la firma:

```
openssl cms -verify -binary -in D:\ma_cli-latest-py3-none-any.whl.cms -inform DER -content D:\ma_cli-latest-py3-none-any.whl -noverify > ./test
```

NOTA

En este ejemplo, el paquete de software se almacena en el archivo **D:**. Reemplácelo con la ruta real.

```
$openssl cms -verify -binary -in package.tar.gz.cms -signer "root" -inform DER -content package.tar.gz -noverify > ./test
CMS Verification successful
```

Paso 3: Instalar ma-cli

1. Ejecute **python --version** en la ventana del símbolo del sistema de su entorno local para verificar si se ha instalado Python. La versión de Python debe ser posterior a 3.7.x y anterior a 3.10.x. Se recomienda la versión 3.7.x.

```
C:\Users\xxx>python --version
Python *.*.*
```

2. Ejecute **pip --version** para verificar si la herramienta general de gestión de paquetes pip está disponible.

```
C:\Users\xxx>pip --version
pip *.*.* from c:\users\xxx\appdata\local\programs\python\python*\lib\site-packages\pip (python *.*.*)
```

3. Instalar ma-cli.

pip install {Path to the ma-cli software package}\ma_cli-latest-py3-none-any.whl

```
C:\Users\xxx>pip install C:\Users\xxx\Downloads\ma_cli-latest-py3-none-any.whl
.....
Successfully installed ma_cli.*.*.*
```

Cuando se instala ma-cli, los paquetes de dependencias se instalan de forma predeterminada. Si aparece el mensaje "Successfully installed" (Se ha instalado correctamente), significa que se ha instalado ma-cli.

NOTA

Si durante la instalación aparece un mensaje de error que indica que falta un paquete de dependencias, ejecute el siguiente comando para instalar el paquete de dependencias cuando se le solicite:

```
pip install xxxx
```

xxxx es el nombre del paquete de dependencias.

6.3 Completado automático para comandos de ma-cli

El completado automático de CLI le permite obtener una lista de comandos de **ma-cli** admitidos. Para ello, escriba un prefijo de comando y presione **Tab** en su terminal. El

completado automático para comandos **ma-cli** debe estar habilitado en Terminal. Después de ejecutar el comando **ma-cli auto-completion**, puede copiar y ejecutar los comandos según se le solicite en el terminal actual para completar automáticamente los comandos **ma-cli**. Se admiten shell Bash, Fish y Zsh. El shell predeterminado es Bash.

Tomemos como ejemplo el comando Bash. Ejecute el comando **eval "\$(_MA_CLI_COMPLETE=bash_source ma-cli)"** en Terminal para habilitar el completado automático.

```
eval "$(_MA_CLI_COMPLETE=bash_source ma-cli)"
```

Ejecute el comando **ma-cli auto-completion Zsh** o **ma-cli auto-completion Fish** para ver el comando del completado automático en Zsh o Fish.

Comandos disponibles

```
$ ma-cli auto-completion -h
Usage: ma-cli auto-completion [OPTIONS] [[Bash|Zsh|Fish]]

Auto complete ma-cli command in terminal.

Example:

# print bash auto complete command to terminal
ma-cli auto-completion Bash

Options:
  -H, -h, --help  Show this message and exit.
```

```
# By default, the autocomplete command for Bash is displayed.
```

```
$ ma-cli auto-completion
```

```
Tips: please paste following shell command to your terminal to activate auto
completion.
```

```
[ OK ] eval "$(_MA_CLI_COMPLETE=bash_source ma-cli)"
```

```
# After the preceding command is executed, autocomplete has been enabled on the
terminal.
```

```
$ eval "$(_MA_CLI_COMPLETE=bash_source ma-cli)"
```

```
# The autocomplete command for Fish is displayed.
```

```
$ ma-cli auto-completion Fish
```

```
Tips: please paste following shell command to your terminal to activate auto
completion.
```

```
[ OK ] eval (env _MA_CLI_COMPLETE=fish_source ma-cli)
```

6.4 Autenticación de ma-cli

Descripción

- Las máquinas virtuales y las computadoras personales requieren la configuración de autenticación. Se puede utilizar tanto un nombre de usuario y una contraseña (predeterminado) como una AK/SK para la autenticación.
- Cuando utilice una cuenta para la autenticación, especifique un nombre de usuario y una contraseña. Cuando utilice una cuenta de IAM para la autenticación, especifique una cuenta, un nombre de usuario y una contraseña.

- En el notebook de ModelArts, no es necesario configurar la autenticación manualmente porque se utiliza una delegación para la autenticación de forma predeterminada.
- Si ha configurado la autenticación en el notebook de ModelArts, se utilizará preferentemente la autenticación especificada.

 **NOTA**

Para garantizar la seguridad de su información confidencial, es importante evitar cualquier posible fuga durante la autenticación.

Parámetros de CLI

```
$ ma-cli configure -h
Usage: ma-cli configure [OPTIONS]

Options:
  -auth, --auth [PWD|AKSK|ROMA]  Authentication type.
  -rp, --region-profile PATH      ModelArts region file path.
  -a, --account TEXT              Account of an IAM user.
  -u, --username TEXT            Username of an IAM user.
  -p, --password TEXT            Password of an IAM user
  -ak, --access-key TEXT         User access key.
  -sk, --secret-key TEXT         User secret key.
  -r, --region TEXT              The region you want to visit.
  -pi, --project-id TEXT         User project id.
  -C, --config-file TEXT         Configure file path for authorization.
  -D, --debug                     Debug Mode. Shows full stack trace when error
occurs.
  -P, --profile TEXT             CLI connection profile to use. The default
profile is "DEFAULT".
  -h, -H, --help                 Show this message and exit.
```

Tabla 6-2 Parámetros de la CLI de autenticación

Parámetro	Tipo	Obligatorio	Descripción
-auth / --auth	String	No	Modo de autenticación, que puede ser PWD (usuario y contraseña) o AKSK (AK/SK). El valor predeterminado es PWD .
-rp / --region-profile	String	No	Archivo de configuración de la región de ModelArts
-a / --account	String	No	La cuenta de tenant de IAM, que debe especificarse cuando se utiliza la autenticación mediante una cuenta de IAM. Se requiere en la autenticación mediante un nombre de usuario y una contraseña.
-u / --username	String	No	Nombre de usuario, que es un nombre de usuario o un nombre de usuario de IAM para la autenticación mediante una cuenta o una cuenta de IAM. Se requiere en la autenticación mediante un nombre de usuario y una contraseña.
-p / --password	String	No	Contraseña, que se requiere en la autenticación mediante un nombre de usuario y una contraseña
-ak / --access-key	String	No	Clave de acceso, que se requiere en la autenticación mediante una AK/SK

Parámetro	Tipo	Obligatorio	Descripción
-sk / --secret-key	String	No	Clave secreta, que se requiere en la autenticación mediante una AK/SK
-r / --region	String	No	Region name. Si este parámetro se deja en blanco, se utilizará por defecto el valor de la variable de entorno REGION_NAME .
-pi / --project-id	String	No	ID del proyecto. Si este parámetro se deja en blanco, se utilizará el valor region (predeterminado) o el valor de la variable de entorno PROJECT_ID .
-P / --profile	String	No	Configuración de autenticación, cuyo valor predeterminado es DEFAULT
-C / --config-file	String	No	Ruta de acceso local al archivo de configuración, cuyo valor predeterminado es ~/.modelarts/ma-cli-profile.yaml

Autenticación mediante nombre de usuario y contraseña

A continuación se describe cómo utilizar el comando **ma-cli configure** en una VM para configurar la autenticación mediante el nombre de usuario y la contraseña.

NOTA

En el ejemplo siguiente, cualquier cadena con **\${}** es una variable. Puede especificar un valor.

Por ejemplo, **`\${your_password}** indica que debe escribir su contraseña.

```
# The DEFAULT authentication configuration is used by default. You need to type
the account, username, and password one by one. If the account and username are
not required, press Enter to skip them.
$ ma-cli configure --auth PWD --region ${your_region}
account: ${your_account}
username: ${your_username}
password: ${your_password} # The input is not displayed on the console.
```

Autenticación con AK/SK

Este comando utiliza una AK/SK para la autenticación, lo que significa que debe ingresarlos de forma interactiva. Su AK/SK no será visible en la consola.

ATENCIÓN

En el ejemplo siguiente, cualquier cadena con **`\${}** es una variable. Puede especificar un valor. Por ejemplo, debe reemplazar **`\${access key}** por su clave de acceso.

```
ma-cli configure --auth AKSK
access key [***]: ${access key}
secret key [***]: ${secret key}
```

Después de ejecutar el comando de autenticación, la información de autenticación se guardará en el archivo de configuración **~/.modelarts/ma-cli-profile.yaml**.

6.5 Comando de creación de ma-cli Image

6.5.1 Comando de creación de ma-cli Image

El comando **ma-cli image** se puede usar para obtener imágenes registradas, obtener o cargar plantillas de creación de imágenes, crear imágenes con Dockerfiles, obtener o borrar cachés de creación de imágenes, registrar o cancelar el registro de imágenes y depurar si las imágenes se pueden usar en instancias de notebook. Para obtener más detalles, ejecute el comando **ma-cli image -h**.

Comandos para crear una imagen

```
$ ma-cli image -h
Usage: ma-cli image [OPTIONS] COMMAND [ARGS]...
  Obtain registered images, register or unregister images, debug images, and
  create images in Notebook.

Options:
  -H, -h, --help  Show this message and exit.

Commands:
  add-template, at  List build-in dockerfile templates.
  build            Build docker image in Notebook.
  debug           Debug SWR image as a Notebook in ECS.
  df              Query disk usage.
  get-image, gi   Query registered image in ModelArts.
  get-template, gt List build-in dockerfile templates.
  prune          Prune image build cache.
  register        Register image to ModelArts.
  unregister      Unregister image from ModelArts.
```

Tabla 6-3 Comandos para crear una imagen

Comando	Descripción
get-template	Obtener una plantilla de creación de imágenes.
add-template	Cargar una plantilla de creación de imágenes.
get-image	Obtener las imágenes de ModelArts registradas.
register	Registre imágenes de SWR con gestión de imágenes de ModelArts.
unregister	Cancelar el registro de una imagen registrada de la gestión de imágenes de ModelArts.
build	Crear una imagen con un Dockerfile (solo compatible con el notebook de ModelArts).
df	Obtenga la caché de creación de imágenes, que solo se puede utilizar en el notebook de ModelArts.

Comando	Descripción
prune	Borrar la caché de creación de imágenes, que solo se puede utilizar en el notebook de ModelArts.
debug	Depurar una imagen de SWR en un ECS para verificar si la imagen se puede usar en notebook de ModelArts. (Solo se pueden utilizar los ECS con Docker instalado)

6.5.2 Obtención de una plantilla de creación de imágenes

ma-cli proporciona algunas plantillas comunes de creación de imágenes, en las que se proporciona la guía para desarrollar Dockerfiles en notebook de ModelArts.

```
$ ma-cli image get-template -h
Usage: ma-cli image get-template [OPTIONS]

List build-in dockerfile templates.

Example:

# List build-in dockerfile templates
ma-cli image get-template [--filer <filter_info>] [--page-num <yourPageNum>]
[--page-size <yourPageSize>]

Options:
  --filter TEXT           filter by keyword.
  -pn, --page-num INTEGER RANGE Specify which page to query. [x>=1]
  -ps, --page-size INTEGER RANGE The maximum number of results for this query.
[x>=1]
  -D, --debug            Debug Mode. Shows full stack trace when error
occurs.
  -P, --profile TEXT     CLI connection profile to use. The default
profile is "DEFAULT".
  -H, -h, --help        Show this message and exit.
(PyTorch-1.4) [ma-user work]$
```

Tabla 6-4 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
--filter	String	No	Filtrar plantillas basadas en la palabra clave del nombre de la plantilla.
-pn / --page-num	Int	No	Índice de página de imágenes. El valor predeterminado es la página 1.
-ps / --page-size	Int	No	Número de imágenes mostradas en cada página. El valor predeterminado es 20 .

Ejemplos

Obtener una plantilla de creación de imágenes.

```
ma-cli image get-template
```

```
(PyTorch-1.8) [ma-user work]$ma-cli image get-template
-----
Template Name      Description
-----
customize_from_ubuntu_18.04_to_modelarts  Add ma-user, apt install packages and create a new conda environment with pip based on scratch ubuntu 18.04
upgrade_current_notebook_apt_packages     Install apt packages like ffmpeg, gcc-8, g++-8 based on current Notebook image
migrate_3rd_party_image_to_modelarts      General template for migrating your own or open source image to ModelArts
migrate_official_torch_1.10_cu113_image_to_modelarts  Reconstructing and migrating the official torch 1.10.0 with cud11.3 image to ModelArts
build_handwritten_number_inference_application  Create a new AI application, used to generate an image to deploy and infer in ModelArts
update_dli_image_pip_package              Install pip packages based on DLI image
forward_compat_cuda_11_image_to_modelarts  Migrate and forward compat cuda-11.x to ModelArts by upgrading only user-mode CUDA components
```

6.5.3 Carga de una plantilla de creación de imágenes

Se utiliza el comando **add-template** para cargar plantillas de imagen en una carpeta especificada. De forma predeterminada, se utiliza la ruta en la que se encuentra el comando actual.

Por ejemplo, `ma-cli image add-template --dst {current_dir}/.ma/{template_name}/`. También puede ejecutar el comando `--dest` para especificar la ruta. Si ya existe una carpeta de plantillas con el mismo nombre en la ruta de destino, ejecute el parámetro `--force` | `-f` para sobrescribirla forzosamente.

```
$ ma-cli image add-template -h
Usage: ma-cli image add-template [OPTIONS] TEMPLATE_NAME

Add buildin dockerfile templates into disk.

Example:

# List build-in dockerfile templates
ma-cli image add-template customize_from_ubuntu_18.04_to_modelarts --force

Options:
  --dst TEXT          target save path.
  -f, --force         Override templates that has been installed.
  -D, --debug         Debug Mode. Shows full stack trace when error occurs.
  -P, --profile TEXT  CLI connection profile to use. The default profile is "DEFAULT".
  -h, -H, --help     Show this message and exit.
```

Tabla 6-5 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
<code>--dst</code>	String	No	Cargue las plantillas en una ruta especificada. La ruta actual se utiliza de forma predeterminada.
<code>-f/ --force</code>	Bool	No	Si se debe sobrescribir forzosamente una plantilla existente con el mismo nombre. De forma predeterminada, la plantilla no se sobrescribe.

Ejemplos

Cargue la plantilla de creación de imágenes `customize_from_ubuntu_18.04_to_modelarts`.

```
ma-cli image add-template customize_from_ubuntu_18.04_to_modelarts
```



```
(PyTorch-1.8) [ma-user work]$ma-cli image add-template customize_from_ubuntu_18.04_to_modelarts
[ OK ] Successfully add configuration template [ customize_from_ubuntu_18.04_to_modelarts ] under folder [ /home/ma-user/work/ma/customize_from_ubuntu_18.04_to_modelarts ]
```

6.5.4 Obtención de imágenes de ModelArts registradas

Normalmente, se proporciona una ruta de acceso a una imagen base en un Dockerfile. Las imágenes públicas y las imágenes de SWR públicas o privadas se pueden obtener de repositorios de imágenes de código abierto como Docker Hub. ma-cli le permite obtener imágenes de ModelArts preestablecidas e imágenes registradas y sus direcciones de SWR.

```
$ma-cli image get-image -h
Usage: ma-cli image get-image [OPTIONS]

Get registered image list.

Example:

# Query images by image type and only image id, show name and swr_path
ma-cli image get-image --type=DEDICATED

# Query images by image id
ma-cli image get-image --image-id ${image_id}

# Query images by image type and show more information
ma-cli image get-image --type=DEDICATED -v

# Query images by image name
ma-cli image get-image --filter=torch

Options:
-t, --type [BUILD_IN|DEDICATED|ALL]      Image type(default ALL)
-f, --filter TEXT                        Image name to filter
-v, --verbose                            Show detailed information on image.
-i, --image-id TEXT                     Get image details by image id
-n, --image-name TEXT                   Get image details by image name
-wi, --workspace-id TEXT                The workspace where you want to query
image(default "0")
-pn, --page-num INTEGER RANGE           Specify which page to query [x>=1]
-ps, --page-size INTEGER RANGE          The maximum number of results for this query
[x>=1]
-C, --config-file PATH                  Configure file path for authorization.
-D, --debug                              Debug Mode. Shows full stack trace when error
occurs.
-P, --profile TEXT                       CLI connection profile to use. The default
profile is "DEFAULT".
-H, -h, --help                          Show this message and exit.
```

Tabla 6-6 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
-t / --type	String	No	Tipo de imágenes a obtener. Las opciones son BUILD_IN , DEDICATED y ALL . <ul style="list-style-type: none"> ● BUILD_IN: imágenes preestablecidas ● DEDICATED: imágenes personalizadas registradas con ModelArts ● ALL: todas las imágenes

Parámetro	Tipo	Obligatorio	Descripción
-f / --filter	String	No	Palabra clave del nombre de una imagen, que se utiliza para filtrar imágenes
-v / --verbose	Bool	No	Si se debe mostrar información detallada. Esta función está deshabilitada por defecto.
-i / --image-id	String	No	Obtener detalles sobre una imagen con un ID especificado.
-n / --image-name	String	No	Obtener detalles sobre una imagen con un nombre especificado.
-wi / --workspace-id	String	No	Obtener imágenes en un espacio de trabajo especificado.
-pn / --page-num	Int	No	Índice de página de imágenes. El valor predeterminado es la página 1.
-ps / --page-size	Int	No	Número de imágenes mostradas en cada página. El valor predeterminado es 20.

Ejemplos

Obtener imágenes personalizadas registradas con ModelArts.

```
ma-cli image get-image --type=DEDICATED
```

```
(PyTorch-1.8) [ma-user work]$ma-cli image get-image --type=DEDICATED
```

INDEX	IMAGE ID	NAME	SMR PATH
1	c857e5a8	fc5e3d002f_0314test	huaweicloud.com/notebook_test/0314test:1.0.0
2	193b2557	d39093a811_0328	7.myhuaweicloud.com/notebook_test/0328:1
3	171fe036	3b37e9aa7c_0926	aweicloud.com/ei_modelarts_y00218826_05/0926:1
4	1b48bb0a	689b0a7267_0926	weicloud.com/ei_modelarts_y00218826_05/0926:111
5	c8667cf0	d2e3563107_1	huaweicloud.com/ei_modelarts_y00218826_05/1:6
6	3e6cd46a	a360eea80e_1	huaweicloud.com/ei_modelarts_y00218826_05/1:1
7	42e86ca5	ec198be968_111	.myhuaweicloud.com/notebook_test/111:1227
8	0f349cef	c411011ef2_11111110801	aweicloud.com/notebook_test/11111110801:111111
9	3a082e32	4f485aad66_112121	weicloud.com/ei_modelarts_y00218826_05/112121:123
10	db0002f6	74eb00e1ce_1203	myhuaweicloud.com/notebook_test/1203:1.2.3
11	031dc02e	fd92cd457d8_1227	.myhuaweicloud.com/notebook_test/1227:111
12	f7d95648	7aaec8b1cc_1227	.myhuaweicloud.com/notebook_test/1227:888
13	2f720610	a1d1db947d_1227	.myhuaweicloud.com/notebook_test/1227:6666
14	42221bf2	f22d726d270_1229	.myhuaweicloud.com/notebook_test/1229:123
15	70deea1e	70b2414ae7_123	myhuaweicloud.com/mindspore-dis-train/123:2
16	e6cc5414	ce318069f4_123	.myhuaweicloud.com/notebook_test/123:45678
17	6e7a86c9	319fb3bb28_1234	.myhuaweicloud.com/notebook_test/1234:666
18	ec036306	8c9dc6b391_1234	7.myhuaweicloud.com/notebook_test/1234:1
19	b37f8f3b	7a9941c978_441211	.myhuaweicloud.com/notebook_test/441211:11
20	d5acd51b	lef16534d68_aaa	.myhuaweicloud.com/notebook_test/aaa:1.1.1

6.5.5 Creación de una imagen en notebook de ModelArts

Ejecute el comando **ma-cli image build** para crear una imagen basada en un Dockerfile especificado. Este comando solo está disponible en instancias de notebook de ModelArts.

```
$ ma-cli image build -h
Usage: ma-cli image build [OPTIONS] FILE_PATH

Build docker image in Notebook.

Example:

# Build a image and push to SWR
ma-cli image build .ma/customize_from_ubuntu_18.04_to_modelarts/Dockerfile -swr
my_organization/my_image:0.0.1

# Build a image and push to SWR, dockerfile context path is current dir
ma-cli image build .ma/customize_from_ubuntu_18.04_to_modelarts/Dockerfile -swr
my_organization/my_image:0.0.1 -context .

# Build a local image and save to local path and OBS
ma-cli image build .ma/customize_from_ubuntu_18.04_to_modelarts/Dockerfile --
target ./build.tar --obs_path obs://bucket/object --swr-path my_organization/
my_image:0.0.1

Options:
  -t, --target TEXT          Name and optionally a tag in the 'name:tag' format.
  -swr, --swr-path TEXT      SWR path without swr endpoint, eg:organization/
image:tag. [required]
  --context DIRECTORY       build context path.
  -arg, --build-arg TEXT     build arg for Dockerfile.
  -obs, --obs-path TEXT     OBS path to save local built image.
  -f, --force                Force to overwrite the existing swr image with the same
name and tag.
  -C, --config-file PATH    Configure file path for authorization.
  -D, --debug                Debug Mode. Shows full stack trace when error occurs.
  -P, --profile TEXT        CLI connection profile to use. The default profile is
"DEFAULT".
  -H, -h, --help            Show this message and exit.
```

Tabla 6-7 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
FILE_PATH	String	Sí	Directorio donde se almacena el Dockerfile
-t / --target	String	No	Ruta local para almacenar el paquete de TAR generado. El directorio actual se utiliza de forma predeterminada.
-swr / --swr-path	String	Sí	Nombre de imagen de SWR, que tiene el formato "organization/image_name:tag". Este parámetro se puede omitir cuando se guarda un paquete de TAR para crear una imagen.
--context	String	No	Ruta de acceso de la información de contexto para copiar datos al crear un Dockerfile

Parámetro	Tipo	Obligatorio	Descripción
-arg / --build-arg	String	No	Parámetro para crear una imagen. Si existen varios parámetros, ejecute --build-arg VERSION=18.04 --build-arg ARCH=X86_64 .
-obs / --obs-path	String	No	Cargue automáticamente el paquete de TAR generado en OBS.
-f / --force	Bool	No	Si se debe sobrescribir forzosamente una imagen de SWR existente con el mismo nombre. Por defecto, la imagen de SWR no se sobrescribe.

Ejemplos

Crear una imagen en el notebook de ModelArts.

```
ma-cli image build .ma/customize_from_ubuntu_18.04_to_modelarts/  
Dockerfile -swr notebook_test/my_image:0.0.1
```

En este comando, **.ma/customize_from_ubuntu_18.04_to_modelarts/Dockerfile** es la ruta donde se almacena el Dockerfile y **notebook_test/my_image:0.0.1** es la ruta de SWR de la nueva imagen.

```
(PyTorch-1.8) [ma-user work]$ ma-cli image build .ma/customize_from_ubuntu_18.04_to_modelarts/Dockerfile -swr notebook_test/my_image:0.0.1
[*] Building 4.3s (8/8) FINISHED
-> [internal] load dockerignore
-> transferring context: 2B
-> [internal] load build definition from Dockerfile
-> transferring dockerfile: 3.22kB
-> [internal] load metadata for swr.cn-north-7.myhuaweicloud.com/atelier/ubuntu:18.04
-> [auth] atelier/ubuntu:pull token for swr.cn-north-7.myhuaweicloud.com
-> [1/2] FROM swr.cn-north-7.myhuaweicloud.com/atelier/ubuntu:18.04#sha256:b58746cda89938b8c9f5b77de3b8cf1fe78210c69eab03a1442e235eea5d84f
-> resolve swr.cn-north-7.myhuaweicloud.com/atelier/ubuntu:18.04#sha256:b58746cda89938b8c9f5b77de3b8cf1fe78210c69eab03a1442e235eea5d84f
-> sha256:2910811b6c4227c2f42aa9a3d5f53b1d469f672cf7e691f631b119b61ff7 847B / 847B
-> sha256:bc3caa0f5b9414127220aa4f42889209e6af24b05668cd188311e00a635f 35.37kB / 35.37kB
-> sha256:365926dccc64eeb1010b2112e6f73981e1a8246e4f64e287763b57f101b09 161B / 161B
-> sha256:23884877105a74f84a910895ca044061a4561385f9c36488ea89b7ec0e771 26.69kB / 26.69kB
-> extracting sha256:23884877105a74f84a910895ca044061a4561385f9c36488ea89b7ec0e771
-> extracting sha256:bc3caa0f5b9414127220aa4f42889209e6af24b05668cd188311e00a635f
-> extracting sha256:2910811b6c4227c2f42aa9a3d5f53b1d469f672cf7e691f631b119b61ff7
-> extracting sha256:365926dccc64eeb1010b2112e6f73981e1a8246e4f64e287763b57f101b09
-> [2/2] RUN default_user=$(getent passwd 1000 | awk -F ':' '{print $1}') || echo "uid: 1000 does not exist" && default_group=$(getent group 100 | awk -F ':' '{pr
-> exporting to image
-> exporting layers
-> exporting manifest sha256:b239078457d7c75d57a45989cf8d9d08e6fd9dc882a4ede6d4311bc487d88e9
-> exporting config sha256:6794fa8ae0c9464b7f3102345237559f82a372963909954841b1340cc51db
-> pushing layers
-> pushing manifest for swr.cn-north-7.myhuaweicloud.com/notebook_test/my_image:0.0.1#sha256:b239078457d7c75d57a45989cf8d9d08e6fd9dc882a4ede6d4311bc487d88e9
-> [auth] notebook_test/my_image:pull,push token for swr.cn-north-7.myhuaweicloud.com

Summary Board
+-----+
+ Image Build Time: 4.3s
+ Repository: swr.cn-north-7.myhuaweicloud.com/notebook_test/my_image
+ Tag: 0.0.1
+ Compressed Image Size: 259B
+ SWR Download Command: docker pull swr.cn-north-7.myhuaweicloud.com/notebook_test/my_image:0.0.1
+-----+
(PyTorch-1.8) [ma-user work]$
```

6.5.6 Obtención de cachés de creación de imágenes en el notebook de ModelArts

Ejecute el comando **ma-cli image df** para obtener cachés de creación de imágenes. Este comando solo está disponible en instancias de notebook de ModelArts.

```
$ ma-cli image df -h
Usage: ma-cli image df [OPTIONS]
```

Query disk usage used by image-building in Notebook.

Example:

```
# Query image disk usage
ma-cli image df
```

```
Options:
-v, --verbose      Show detailed information on disk usage.
-D, --debug        Debug Mode. Shows full stack trace when error occurs.
-h, -H, --help     Show this message and exit.
```

Tabla 6-8 Parámetros

Parámetro	Tipo	Obligatori o	Descripción
-v / --verbose	Bool	No	Si se debe mostrar información detallada. Esta función está deshabilitada por defecto.

Ejemplos

- Consulte todas las cachés de imágenes del notebook de ModelArts.

```
ma-cli image df
```

```
(PyTorch-1.8) [ma-user work]$ma-cli image df
ID                                     RECLAIMABLE  SIZE      LAST ACCESSED
iwrws19pdcjafel1j6d0r918             true         98.50MB
cp52c4q81ud2abu2vp7s3j5vyt           true         1.04MB
4jbo6v06r2w1575ddq3w8g12e           true         139.68kB
ojdjw5mok71s1nh2cauant051           true         86.86kB
k2jm6g061n5tvmz7gmonmqjsh           true         16.55kB
efu5kwwig1ve44fe7smbnrcnh*          true         8.19kB
uzikwqk5taxnslvajm14jrbje*          true         4.10kB
2g8p0qcb014g3qva7ucawkv87*          true         4.10kB
Reclaimable: 99.80MB
Total: 99.80MB
```

- Consulte detalles sobre una imagen.

```
ma-cli image df --verbose
```

```
(PyTorch-1.8) [ma-user work]$ma-cli image df --verbose
ID: iwrws19pdcjafel1j6d0r918
Created at: 2023-03-28 12:23:28.353759532 +0000 UTC
Mutable: false
Reclaimable: true
Shared: false
Size: 98.50MB
Description: pulled from swr .myhuaweicloud.com/atelier/ubuntu:18.04@sha256:b5874...a65d84f
Usage count: 1
Last used: 2023-03-28 12:23:28.37337776 +0000 UTC
Type: regular

ID: cp52c4q81ud2abu2vp7s3j5vyt
Parents: iwrws19pdcjafel1j6d0r918
Created at: 2023-03-28 12:23:28.36910223 +0000 UTC
Mutable: false
Reclaimable: true
Shared: false
Size: 1.04MB
Description: pulled from swr .myhuaweicloud.com/atelier/ubuntu:18.04@sha256:b5874...2e235ea65d84f
Usage count: 1
Last used: 2023-03-28 12:23:28.38560437 +0000 UTC
Type: regular

ID: 4jbo6v06r2w1575ddq3w8g12e
Parents: k2jm6g061n5tvmz7gmonmqjsh
Created at: 2023-03-28 12:23:30.681643727 +0000 UTC
Mutable: false
Reclaimable: true
Shared: false
Size: 139.68kB
Description: mount / from exec /bin/sh -c default_user=$(getent passwd 1000 | awk -F ':' '{print $1}') || echo "uid: 1000 does not exist" && default_group=$(getent
up 100 | awk -F ':' '{print $1}') || echo "gid: 100 does not exist" && if [ ! -z "$(default_user)" ] && [ "$(default_user)" != "ma-user" ]; then userdel -r $(defau
user); fi && if [ ! -z "$(default_group)" ] && [ "$(default_group)" != "ma-group" ]; then groupdel -f $(default_group); fi && groupadd -g 100 ma-group
useradd -d /home/ma-user -m -u 1000 -g 100 -s /bin/bash ma-user && chmod -R 750 /home/ma-user
Usage count: 2
Last used: 2023-03-28 12:25:39.149880471 +0000 UTC
Type: regular
```

6.5.7 Borrado de cachés de creación de imágenes en el notebook de ModelArts

Ejecute el comando **ma-cli image prune** para borrar las cachés de creación de imágenes. Este comando solo está disponible en instancias de notebook de ModelArts.

```
$ ma-cli image prune -h
Usage: ma-cli image prune [OPTIONS]
```

```
Prune image build cache by image-building in Notebook.

Example:

# Prune image build cache
ma-cli image prune

Options:
  -ks, --keep-storage INTEGER Amount of disk space to keep for cache below this
limit (in MB) (default: 0).
  -kd, --keep-duration TEXT Keep cache newer than this limit, support
second(s), minute(m) and hour(h) (default: 0s).
  -v, --verbose Show more verbose output.
  -D, --debug Debug Mode. Shows full stack trace when error
occurs.
  -h, -H, --help Show this message and exit.
```

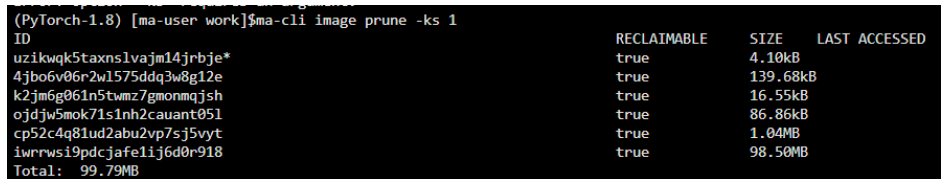
Tabla 6-9 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
-ks / --keep-storage	Int	No	Tamaño de la memoria caché que se va a conservar, en MB. El valor predeterminado es 0 e indica que se borrarán todas las cachés.
-kd / --keep-duration	String	No	Si se deben conservar las cachés más recientes y borrar solo las cachés históricas. La unidad puede ser s (segundo), m (minuto) o h (hora). El valor predeterminado es 0 e indica que se borrarán todas las cachés.
-v / --verbose	Bool	No	Si se debe mostrar información detallada. Esta función está deshabilitada por defecto.

Ejemplos

Conserve 1 MB de caché de imágenes al borrar cachés.

```
ma-cli image prune -ks 1
```



```
(PyTorch-1.8) [ma-user work]$ma-cli image prune -ks 1
ID                                     RECLAIMABLE  SIZE  LAST ACCESSED
uzikwqk5taxnslvajm14jrbje*           true         4.10kB
4jbo6v06r2w1575ddq3w8g12e           true        139.68kB
k2jm6g061n5twmz7gmonmqjsh           true        16.55kB
ojdjw5mok71s1nh2cauant05l           true        86.86kB
cp52c4q81ud2abu2vp7sj5vyt           true         1.04MB
iwrwrsi9pdcjafel1j6d0r918           true        98.50MB
Total: 99.79MB
```

6.5.8 Registro de imágenes de SWR con la gestión de imágenes de ModelArts

Después de depurar una imagen, ejecute el comando **ma-cli image register** para registrarla con la gestión de imágenes de ModelArts para poder usarla en ModelArts.

```
$ma-cli image register -h
Usage: ma-cli image register [OPTIONS]
```

```

Register image to ModelArts.

Example:

# Register image into ModelArts service
ma-cli image register --swr-path=xx

# Share SWR image to DLI service
ma-cli image register -swr xx -td

# Register image into ModelArts service and specify architecture to be 'AARCH64'
ma-cli image register --swr-path=xx --arch AARCH64

Options:
  -swr, --swr-path TEXT           SWR path without swr endpoint, eg:organization/
image:tag. [required]
  -a, --arch [X86_64|AARCH64]     Image architecture (default: X86_64).
  -s, --service [NOTEBOOK|MODELBOX] Services supported by this image(default
NOTEBOOK).
  -rs, --resource-category [CPU|GPU|ASCEND] The resource category supported by this image
(default: CPU and GPU).
  -wi, --workspace-id TEXT        The workspace to register this image (default:
"0").
  -v, --visibility [PUBLIC|PRIVATE] PUBLIC: every user can use this image. PRIVATE:
only image owner can use this image (Default: PRIVATE).
  -td, --to-dli                   Register swr image to DLI, which will share SWR
image to DLI service.
  -d, --description TEXT          Image description (default: "").
  -C, --config-file PATH          Configure file path for authorization.
  -D, --debug                      Debug Mode. Shows full stack trace when error
occurs.
  -P, --profile TEXT              CLI connection profile to use. The default
profile is "DEFAULT".
  -h, -H, --help                  Show this message and exit.

```

Tabla 6-10 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
-swr / --swr-path	String	Sí	Ruta de SWR hacia la imagen que se va a registrar
-a / --arch	String	No	Arquitectura de la imagen registrada. El valor puede ser X86_64 o AARCH64 . El valor predeterminado es X86_64 .
-s / --service	String	No	Tipo de servicio de la imagen registrada. El valor puede ser NOTEBOOK o MODELBOX . El valor predeterminado es NOTEBOOK . También puede especificar ambos valores, -s NOTEBOOK -s MODELBOX .
-rs / --resource-category	String	No	Tipo de recurso que puede utilizar la imagen registrada. El valor puede ser CPU , GPU o ASCEND . El valor predeterminado es CPU y GPU .

Parámetro	Tipo	Obligatorio	Descripción
-wi / --workspace-id	String	No	Registre una imagen en un espacio de trabajo específico. El ID predeterminado del espacio de trabajo es 0.
-v / --visibility	Bool	No	Ámbito disponible de la imagen registrada. El valor puede ser PRIVATE (disponible solo para el propietario de la imagen) o PUBLIC (disponible para todos los usuarios). El valor predeterminado es PRIVATE .
-td / --to-dli	Bool	No	Registre una imagen con DLI.
-d/ --description	String	No	Describa una imagen. De forma predeterminada, este parámetro se deja en blanco.

Ejemplos

Registre una imagen de SWR con ModelArts.

```
ma-cli image register --swr-path=xx
```

```
(PyTorch-1.8) [ma-user work]$ma-cli image register --swr-path=swr.cn-np-myhuaweicloud.com/notebook- /my_image:0.0.1
You are now in a notebook or devcontainer and cannot use 'ImageManagement.debug' to check your image. If you need to debug it, please use a workstation.
[ OK ] Successfully registered this image and image information is
{
  "arch": "x86_64",
  "create_at": "1688006812157",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "85-0a66748",
  "name": "my_image",
  "namespace": "notebook_test",
  "origin": "CUSTOMIZE",
  "resource_categories": [
    "GPU",
    "CPU"
  ],
  "service_type": "UNKNOWN",
  "size": "26735097",
  "status": "ACTIVE",
  "swr_path": "swr.cn-np-myhuaweicloud.com/notebook- /my_image:0.0.1",
  "tag": "0.0.1",
  "tags": [],
  "type": "DEDICATED",
  "update_at": "1688006812157",
  "visibility": "PRIVATE",
  "workspace_id": "0"
}
```

6.5.9 Cancelación del registro de una imagen en la gestión de imágenes de ModelArts

Ejecute el comando **ma-cli image unregister** para cancelar el registro de una imagen registrada de ModelArts.

```
$ ma-cli image unregister -h
Usage: ma-cli image unregister [OPTIONS]

Unregister image from ModelArts.

Example:

# Unregister image
ma-cli image unregister --image-id=xx

# Unregister image and delete it from swr
ma-cli image unregister --image-id=xx -d
```



```
Options:
  -i, --image-id TEXT      Unregister image details by image id. [required]
  -d, --delete-swr-image   Delete the image from swr.
  -C, --config-file PATH   Configure file path for authorization.
  -D, --debug              Debug Mode. Shows full stack trace when error occurs.
  -P, --profile TEXT       CLI connection profile to use. The default profile is
  "DEFAULT".
  -h, -H, --help          Show this message and exit.
```

Tabla 6-11 Parámetros

Parámetro	Tipo	Obliga torio	Descripción
-i / -image-id	String	Sí	ID de la imagen que se va a dar de baja
-d / --delete-swr-image	Bool	No	Si se debe eliminar una imagen de SWR dada de baja. Esta función está deshabilitada por defecto.

Ejemplos

Cancelar el registro de una imagen registrada de la gestión de imágenes de ModelArts.

```
ma-cli image unregister --image-id=xx
```

```
(PyTorch-1.8) [ma-user work]$ma-cli image unregister --image-id=852f85c1590a66748
[ OK ] Successfully unregistered image 852f85dd-1590a66748
```

6.5.10 Depuración de una imagen de SWR en un ECS

ma-cli permite depurar una imagen de SWR en un ECS para determinar si usarla en un entorno de desarrollo de ModelArts.

```
ma-cli image debug -h
Usage: ma-cli image debug [OPTIONS]

  Debug SWR image as a Notebook in ECS.

  Example:

  # Debug cpu notebook image
  ma-cli image debug --swr-path=xx --service=NOTEBOOK --region=us-east-1

  # Debug gpu notebook image
  ma-cli image debug --swr-path=xx --service=NOTEBOOK --region=us-east-1 --gpu

Options:
  -swr, --swr-path TEXT      SWR path without SWR endpoint, eg:organization/image:tag. [required]
  -r, --region TEXT          Region name. [required]
  -s, --service [NOTEBOOK|MODELBOX]
                              Services supported by this image(default NOTEBOOK).
  -a, --arch [X86_64|AArch64] Image architecture(default X86_64).
  -g, --gpu                  Use all gpus to debug.
  -D, --debug               Debug Mode. Shows full stack trace when error occurs.
  -P, --profile TEXT        CLI connection profile to use. The default profile is "DEFAULT".
  -h, -H, --help           Show this message and exit.
```

Tabla 6-12 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
-swr / --swr-path	String	Sí	Ruta de SWR a la imagen que se va a depurar
-r / --region	String	Sí	Región donde se encuentra la imagen que se va a depurar
-s / --service	String	No	Tipo de servicio de la imagen depurada. El valor puede ser NOTEBOOK o MODELBOX . El valor predeterminado es NOTEBOOK .
-a / --arch	String	No	Arquitectura de la imagen depurada. El valor puede ser X86_64 o AARCH64 . El valor predeterminado es X86_64 .
-g / --gpu	Bool	No	Estado de depuración de la GPU. Esta función está deshabilitada por defecto.

6.6 Uso del comando `ma-cli ma-job` para enviar un trabajo de entrenamiento de ModelArts

6.6.1 Descripción del comando `ma-cli ma-job`

Ejecute el comando `ma-cli ma-job` para enviar trabajos de entrenamiento, obtener registros, eventos, motores de IA usados y especificaciones de recursos, y detener trabajos de entrenamiento.

```
$ ma-cli ma-job -h
Usage: ma-cli ma-job [OPTIONS] COMMAND [ARGS]...

ModelArts job submission and query job details.

Options:
  -h, -H, --help  Show this message and exit.

Commands:
  delete      Delete training job by job id.
  get-engine  Get job engines.
  get-event   Get job running event.
  get-flavor  Get job flavors.
  get-job     Get job details.
  get-log     Get job log details.
  get-pool    Get job engines.
  stop        Stop training job by job id.
  submit      Submit training job.
```

Tabla 6-13 Comandos soportados por trabajos de entrenamiento

Comando	Descripción
<code>get-job</code>	Obtener trabajos de entrenamiento de ModelArts y sus detalles.

Comando	Descripción
get-log	Obtener logs de tiempo de ejecución de un trabajo de entrenamiento de ModelArts.
get-engine	Obtenga motores de IA de ModelArts para entrenamiento.
get-event	Obtener eventos de trabajo de entrenamiento de ModelArts.
get-flavor	Obtener especificaciones de recursos de ModelArts para entrenamiento.
get-pool	Obtener grupos de recursos dedicados de ModelArts al entrenamiento.
stop	Detener un trabajo de entrenamiento de ModelArts.
submit	Enviar un trabajo de entrenamiento de ModelArts.
delete	Eliminar un trabajo de entrenamiento con un ID de trabajo especificado.

6.6.2 Obtención de trabajos de entrenamiento de ModelArts

Ejecute el comando **ma-cli ma-job get-job** para ver trabajos de entrenamiento o detalles sobre un trabajo específico.

```
$ ma-cli ma-job get-job -h
Usage: ma-cli ma-job get-job [OPTIONS]

Get job details.

Example:

# Get train job details by job name
ma-cli ma-job get-job -n ${job_name}

# Get train job details by job id
ma-cli ma-job get-job -i ${job_id}

# Get train job list
ma-cli ma-job get-job --page-size 5 --page-num 1

Options:
  -i, --job-id TEXT           Get training job details by job id.
  -n, --job-name TEXT        Get training job details by job name.
  -pn, --page-num INTEGER    Specify which page to query. [x>=1]
  -ps, --page-size INTEGER RANGE The maximum number of results for this query.
                              [1<=x<=50]
  -v, --verbose              Show detailed information about training job
                              details.
  -C, --config-file TEXT     Configure file path for authorization.
  -D, --debug                Debug Mode. Shows full stack trace when error
                              occurs.
  -P, --profile TEXT         CLI connection profile to use. The default
                              profile is "DEFAULT".
  -h, -H, --help            Show this message and exit.
```

Tabla 6-14 Descripción

Parámetro	Tipo	Obligato rio	Descripción
-i / --job-id	String	No	Obtener detalles sobre un trabajo de entrenamiento con un ID de trabajo específico.
-n / --job-name	String	No	Obtenga un trabajo de entrenamiento con un nombre de trabajo especificado o filtre los trabajos de entrenamiento por nombre de trabajo.
-pn / --page-num	Int	No	Número de página. El valor predeterminado es la página 1.
-ps / --page-size	Int	No	Número de trabajos de entrenamiento mostrados en cada página. El valor predeterminado es 10 .
-v / --verbose	Bool	No	Si se debe mostrar información detallada. Esta función está deshabilitada por defecto.

Ejemplos

- Obtener un trabajo de entrenamiento con el ID de trabajo especificado.

```
ma-cli ma-job get-job -i b63e90xxx
```

```
(PyTorch-1.4) [ma-user work]$ma-cli ma-job get-job -i b63e90ba-91
```

id	name	status	user_name	duration	create_time	start_time	descripti
b63e90ba-91	workflow_created_job_ed3a963f-5438-4a99-9a19-c97ce88c48bb	Completed	ei_modelarts_6_05	00h:01m:16s	2023-03-29 03:41:21	2023-03-29 03:41:30	

- Filtrar trabajos de entrenamiento por nombre de trabajo **auto**.

```
ma-cli ma-job get-job -n auto
```

```
(PyTorch-1.4) [ma-user work]$ma-cli ma-job get-job -n auto
```

inde	id	name	status	user_name	duration	create_time	start_time
1	9b495c	autotest_0he278-copy-4502	Completed	ei_modelarts_y0021882_6_05	00h:01m:31s	2023-03-29 07:03:08	2023-03-29 07:05:20
2	af2147f5-	autotest_0he278-copy-ae52	Terminated	ei_modelarts_y0021882_6_05	00h:10m:49s	2023-03-29 06:52:16	2023-03-29 06:52:32
3	2c1855b1-	autotest_nv487q	Failed	ei_modelarts_y0021882_6_05	00h:37m:29s	2023-03-29 03:22:31	2023-03-29 03:22:58
4	4525b3c9-	autotest_x2cjf6	Failed	ei_modelarts_y0021882_6_05	00h:00m:01s	2023-03-29 03:19:41	2023-03-29 03:19:49
5	4234455d-	autotest_sx71zc	Terminated	ei_modelarts_y0021882_6_05	00h:00m:00s	2023-03-29 02:25:18	N/A
6	9810ae49-	autotest_s62gz3	Terminated	ei_modelarts_y0021882_6_05	00h:09m:06s	2023-03-29 02:19:49	2023-03-29 02:20:13
7	90c7de89-	autotest_wf8z2g	Abnormal	ei_modelarts_y0021882_6_05	00h:00m:00s	2023-03-29 01:43:18	N/A
8	fc740dc5-	autotest_g17mit	Terminated	ei_modelarts_y0021882_6_05	00h:00m:00s	2023-03-29 01:22:19	N/A
9	5d16fdfe-	autotest_02df046i	Terminated	ei_modelarts_y0021882_6_05	00h:00m:00s	2023-03-29 01:11:26	N/A
10	3737e56d-	autotest_clutp0	Completed	ei_modelarts_y0021882_6_05	00h:05m:59s	2023-03-29 00:59:28	2023-03-29 01:04:20

6.6.3 Envío de un trabajo de entrenamiento de ModelArts

Ejecute el comando **ma-cli ma-job submit** para enviar un trabajo de entrenamiento de ModelArts.

Antes de ejecutar este comando, configure **YAML_FILE** para especificar la ruta al archivo de configuración del trabajo de destino. Si no se especifica este parámetro, el archivo de configuración estará vacío. El archivo de configuración está en formato de YAML y sus parámetros son el parámetro **option** del comando. Si especifica tanto el archivo de configuración **YAML_FILE** como el parámetro **option** en la CLI, el valor del parámetro **option** lo sobrescribirá en el archivo de configuración.

```
$ma-cli ma-job submit -h
Usage: ma-cli ma-job submit [OPTIONS] [YAML_FILE]...

Submit training job.

Example:

ma-cli ma-job submit --code-dir obs://your_bucket/code/
                    --boot-file main.py
                    --framework-type PyTorch
                    --working-dir /home/ma-user/modelarts/user-job-dir/code
                    --framework-version pytorch_1.8.0-cuda_10.2-py_3.7-
ubuntu_18.04-x86_64
                    --data-url obs://your_bucket/dataset/
                    --log-url obs://your_bucket/logs/
                    --train-instance-type modelarts.vm.cpu.8u
                    --train-instance-count 1

Options:
  --name TEXT                Job name.
  --description TEXT         Job description.
  --image-url TEXT           Full swr custom image path.
  --uid TEXT                 Uid for custom image (default: 1000).
  --working-dir TEXT        ModelArts training job working directory.
  --local-code-dir TEXT     ModelArts training job local code directory.
  --user-command TEXT       Execution command for custom image.
  --pool-id TEXT            Dedicated pool id.
  --train-instance-type TEXT Train worker specification.
  --train-instance-count INTEGER Number of workers.
  --data-url TEXT           OBS path for training data.
  --log-url TEXT            OBS path for training log.
  --code-dir TEXT           OBS path for source code.
  --output TEXT             Training output parameter with OBS path.
  --input TEXT              Training input parameter with OBS path.
  --env-variables TEXT      Env variables for training job.
  --parameters TEXT         Training job parameters (only keyword
parameters are supported).
  --boot-file TEXT          Training job boot file path behinds `code_dir`.
  --framework-type TEXT     Training job framework type.
  --framework-version TEXT  Training job framework version.
  --workspace-id TEXT       The workspace where you submit training
job(default "0")
  --policy [regular|economic|turbo|auto]
                             Training job policy, default is regular.
  --volumes TEXT            Information about the volumes attached to the
training job.
  -q, --quiet                Exit without waiting after submit successfully.
  -C, --config-file PATH     Configure file path for authorization.
  -D, --debug                Debug Mode. Shows full stack trace when error
occurs.
  -P, --profile TEXT         CLI connection profile to use. The default
profile is "DEFAULT".
  -H, -h, --help            Show this message and exit.
```

Tabla 6-15 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
YAML_FILE	String	No	Archivo de configuración de un trabajo de entrenamiento. Si no se especifica este parámetro, el archivo de configuración estará vacío.
--code-dir	String	Sí	Ruta de OBS hacia el código fuente del entrenamiento
--data-url	String	Sí	Ruta de OBS hacia los datos de entrenamiento
--log-url	String	Sí	Ruta de OBS a los logs de entrenamiento
--train-instance-count	String	Sí	Número de nodos de cómputo en un trabajo de entrenamiento. El valor predeterminado es 1 e indica que se trata de un nodo independiente.
--boot-file	String	No	El archivo de inicio especificado cuando se utiliza un comando preestablecido se utiliza para enviar un trabajo de entrenamiento. Este parámetro se puede omitir cuando se utiliza una imagen o un comando personalizados para enviar un trabajo de entrenamiento.
--name	String	No	Nombre de un trabajo de entrenamiento
--description	String	No	Descripción de un trabajo de entrenamiento
--image-url	String	No	URL de SWR de una imagen personalizada, que tiene el formato "organization/image_name:tag".
--uid	String	No	UID en tiempo de ejecución de una imagen personalizada. El valor predeterminado es 1000 .
--working-dir	String	No	Directorio de trabajo donde se ejecuta un algoritmo
--local-code-dir	String	No	Directorio local del contenedor de entrenamiento en el que se descarga el directorio de códigos del algoritmo
--user-command	String	No	Comando para ejecutar una imagen personalizada. El directorio debe estar bajo /home . Este parámetro no tiene efecto cuando el prefijo file:// es code-dir .
--pool-id	String	No	ID del grupo de recursos seleccionado para un trabajo de entrenamiento. Para obtener el ID, haga lo siguiente: Inicie sesión en la consola de gestión de ModelArts y elija Dedicated Resource Pools en el panel de navegación de la izquierda y vea el ID del grupo de recursos en la lista del grupo de recursos dedicado.
--train-instance-type	String	No	Variante de recurso seleccionada para un trabajo de entrenamiento

Parámetro	Tipo	Obligatorio	Descripción
--output	String	No	Training output. Una vez especificado este parámetro, el trabajo de entrenamiento cargará el directorio de salida del contenedor de entrenamiento correspondiente al parámetro de salida especificado en el script de entrenamiento en una ruta de OBS especificada. Para especificar múltiples parámetros, utilice --output output1=obs://bucket/output1 --output output2=obs://bucket/output2 .
--input	String	No	Entradas de entrenamiento. Una vez especificado este parámetro, el trabajo de entrenamiento descargará los datos de OBS al contenedor de entrenamiento y transferirá la ruta de almacenamiento de datos al script de entrenamiento a través del parámetro especificado. Para especificar múltiples parámetros, utilice --input data_path1=obs://bucket/data1 --input data_path2=obs://bucket/data2 .
--env-variables	String	No	Entrada de variables de entorno durante el entrenamiento. Para especificar múltiples parámetros, utilice --env-variables ENV1=env1 --env-variables ENV2=env2 .
--parameters	String	No	Parámetros de entrada de entrenamiento. Para especificar varios parámetros, utilice --parameters "--epoch 0 --pretrained" .
--framework-type	String	No	Motor seleccionado para un trabajo de entrenamiento
--framework-version	String	No	Versión del motor seleccionada para un trabajo de entrenamiento
-q / --quiet	Bool	No	Después de enviar un trabajo de entrenamiento, el sistema se cierra directamente y no imprime el estado del trabajo de manera sincrónica.
--workspace-id	String	No	Espacio de trabajo donde se despliega un trabajo de entrenamiento. El valor predeterminado es 0 .
--policy	String	No	Modo de especificación de recursos de entrenamiento. Las opciones son regular , economic , turbo y auto .
--volumes	String	No	Montar discos de EFS. Para especificar varios parámetros, utilice --volumes . "local_path=/xx/yy/ zz;read_only=false;nfs_server_path=xxx.xxx.xxx.xxx:/" - volumes "local_path=/xxx/yyy/ zzz;read_only=false;nfs_server_path=xxx.xxx.xxx.xxx:/"

Envío de un trabajo de entrenamiento basado en una imagen preestablecida de ModelArts

Enviar un trabajo de entrenamiento especificando el parámetro **options** en la CLI.

```
ma-cli ma-job submit --code-dir obs://your-bucket/mnist/code/ \  
    --boot-file main.py \  
    --framework-type PyTorch \  
    --working-dir /home/ma-user/modelarts/user-job-dir/code \  
    --framework-version pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-  
x86_64 \  
    --data-url obs://your-bucket/mnist/dataset/MNIST/ \  
    --log-url obs://your-bucket/mnist/logs/ \  
    --train-instance-type modelarts.vm.cpu.8u \  
    --train-instance-count 1 \  
-q
```

A continuación se muestra un ejemplo de **train.yaml** utilizando una imagen predefinida:

```
# Example .ma/train.yaml (preset image)  
# pool_id: pool_xxxx  
train-instance-type: modelarts.vm.cpu.8u  
train-instance-count: 1  
data-url: obs://your-bucket/mnist/dataset/MNIST/  
code-dir: obs://your-bucket/mnist/code/  
working-dir: /home/ma-user/modelarts/user-job-dir/code  
framework-type: PyTorch  
framework-version: pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64  
boot-file: main.py  
log-url: obs://your-bucket/mnist/logs/  
  
##[Optional] Uncomment to set uid when use custom image mode  
uid: 1000  
  
##[Optional] Uncomment to upload output file/dir to OBS from training platform  
output:  
  - name: output_dir  
    obs_path: obs://your-bucket/mnist/output1/  
  
##[Optional] Uncomment to download input file/dir from OBS to training platform  
input:  
  - name: data_url  
    obs_path: obs://your-bucket/mnist/dataset/MNIST/  
  
##[Optional] Uncomment pass hyperparameters  
parameters:  
  - epoch: 10  
  - learning_rate: 0.01  
  - pretrained:  
  
##[Optional] Uncomment to use dedicated pool  
pool_id: pool_xxxx  
  
##[Optional] Uncomment to use volumes attached to the training job  
volumes:  
  - efs:  
    local_path: /xx/yy/zz  
    read_only: false  
    nfs_server_path: xxx.xxx.xxx.xxx:/
```

Uso de una imagen personalizada para crear un trabajo de entrenamiento

Enviar un trabajo de entrenamiento especificando el parámetro **options** en la CLI.

```
ma-cli ma-job submit --image-url atelier/pytorch_1_8:pytorch_1.8.0-cuda_10.2-  
py_3.7-ubuntu_18.04-x86_64-20220926104358-041ba2e \  
    --code-dir obs://your-bucket/mnist/code/ \  
-q
```



```

--user-command "export LD_LIBRARY_PATH=/usr/local/cuda/
compat:$LD_LIBRARY_PATH && cd /home/ma-user/modelarts/user-job-dir/code && /
home/ma-user/anaconda3/envs/PyTorch-1.8/bin/python main.py" \
--data-url obs://your-bucket/mnist/dataset/MNIST/ \
--log-url obs://your-bucket/mnist/logs/ \
--train-instance-type modelarts.vm.cpu.8u \
--train-instance-count 1 \
-g

```

A continuación se muestra un ejemplo de **train.yaml** utilizando una imagen personalizada:

```

# Example .ma/train.yaml (custom image)
image-url: atelier/pytorch_1_8:pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-
x86_64-20220926104358-041ba2e
user-command: export LD_LIBRARY_PATH=/usr/local/cuda/compat:$LD_LIBRARY_PATH &&
cd /home/ma-user/modelarts/user-job-dir/code && /home/ma-user/anaconda3/envs/
PyTorch-1.8/bin/python main.py
train-instance-type: modelarts.vm.cpu.8u
train-instance-count: 1
data-url: obs://your-bucket/mnist/dataset/MNIST/
code-dir: obs://your-bucket/mnist/code/
log-url: obs://your-bucket/mnist/logs/

##[Optional] Uncomment to set uid when use custom image mode
uid: 1000

##[Optional] Uncomment to upload output file/dir to OBS from training platform
output:
  - name: output_dir
    obs_path: obs://your-bucket/mnist/output1/

##[Optional] Uncomment to download input file/dir from OBS to training platform
input:
  - name: data_url
    obs_path: obs://your-bucket/mnist/dataset/MNIST/

##[Optional] Uncomment pass hyperparameters
parameters:
  - epoch: 10
  - learning_rate: 0.01
  - pretrained:

##[Optional] Uncomment to use dedicated pool
pool_id: pool_xxxx

##[Optional] Uncomment to use volumes attached to the training job
volumes:
  - efs:
    local_path: /xx/yy/zz
    read_only: false
    nfs_server_path: xxx.xxx.xxx.xxx:/

```

Ejemplos

- Enviar un trabajo de entrenamiento basado en un archivo YAML.

```
ma-cli ma-job submit ./train-job.yaml
```

```

(PyTorch-1.4) [ma-user work]$ma-cli ma-job submit ./train_job.yaml
[ OK ] Current training job id is: 4d7c8584-b213-4f88-9833-d3e6a82f9e42
[ OK ] Creating
[ OK ] Running

```

- Enviar un trabajo de entrenamiento utilizando **pytorch1.8-cuda10.2-cudnn7-ubuntu18.04** de imágenes preestablecidas con la CLI.

```

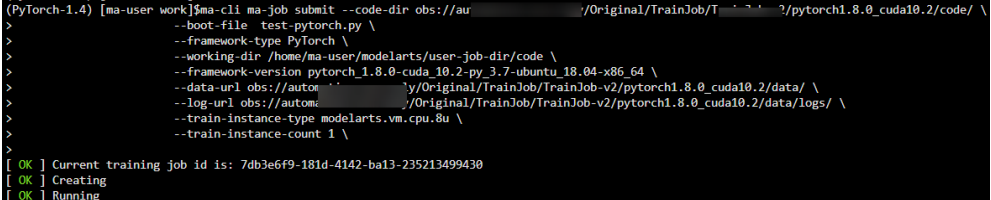
ma-cli ma-job submit --code-dir obs://automation-use-only/Original/
TrainJob/TrainJob-v2/pytorch1.8_0_cuda10.2/code/ \
--boot-file test-pytorch.py \

```

```

--framework-type PyTorch \
--working-dir /home/ma-user/modelarts/user-job-
dir/code \
--framework-version pytorch_1.8.0-cuda_10.2-
py_3.7-ubuntu_18.04-x86_64 \
--data-url obs://automation-use-only/Original/
TrainJob/TrainJob-v2/pytorch1.8.0_cuda10.2/data/ \
--log-url obs://automation-use-only/Original/
TrainJob/TrainJob-v2/pytorch1.8.0_cuda10.2/data/logs/ \
--train-instance-type modelarts.vm.cpu.8u \
--train-instance-count 1 \

```



```

(PyTorch-1.4) [ma-user work]$ma-cli ma-job submit --code-dir obs://au
.../Original/TrainJob/T...?/pytorch1.8.0_cuda10.2/code/ \
> --boot-file test-pytorch.py \
> --framework-type PyTorch \
> --working-dir /home/ma-user/modelarts/user-job-dir/code \
> --framework-version pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64 \
> --data-url obs://autom...y/Original/TrainJob/TrainJob-v2/pytorch1.8.0_cuda10.2/data/ \
> --log-url obs://autom.../Original/TrainJob/TrainJob-v2/pytorch1.8.0_cuda10.2/data/logs/ \
> --train-instance-type modelarts.vm.cpu.8u \
> --train-instance-count 1 \
>
[ OK ] Current training job id is: 7db3e6f9-181d-4142-ba13-235213499430
[ OK ] Creating
[ OK ] Running

```

6.6.4 Obtención de logs de trabajos de entrenamiento de ModelArts

Ejecute el comando **ma-cli ma-job get-log** para obtener los logs de trabajo de entrenamiento de ModelArts.

```

$ ma-cli ma-job get-log -h
Usage: ma-cli ma-job get-log [OPTIONS]

Get job log details.

Example:

# Get job log by job id
ma-cli ma-job get-log --job-id ${job_id}

Options:
  -i, --job-id TEXT          Get training job details by job id. [required]
  -t, --task-id TEXT        Get training job details by task id (default
"worker-0").
  -C, --config-file TEXT    Configure file path for authorization.
  -D, --debug                Debug Mode. Shows full stack trace when error occurs.
  -P, --profile TEXT        CLI connection profile to use. The default profile is
"DEFAULT".
  -h, -H, --help            Show this message and exit.

```

Parámetro	Tipo	Obligatorio	Descripción
-i / --job-id	String	Sí	Obtener logs de un trabajo de entrenamiento con un ID de trabajo especificado.
-t / --task-id	String	No	Obtener logs de una tarea específica, cuyo valor predeterminado es work-0 .

Ejemplos

Obtener logs de un trabajo de entrenamiento con un ID de trabajo especificado.

```
ma-cli ma-job get-log --job-id b63e90baxxx
```

```
(PyTorch-1.4) [ma-user work]$ma-cli ma-job get-log --job-id b63e90ba-
time="2023-03-29T11:41:26+08:00" level-info msg="init logger successful" file="init.go:55" Command-bootstrap/init Component=ma-training-toolkit Platform=ModelArts-Service
time="2023-03-29T11:41:26+08:00" level-info msg="current user 1000:1000" file="init.go:57" Command-bootstrap/init Component=ma-training-toolkit Platform=ModelArts-Service
time="2023-03-29T11:41:27+08:00" level-info msg="report event" file="init.go:81" Command-bootstrap/init Component=ma-training-toolkit Platform=ModelArts-Service
time="2023-03-29T11:41:27+08:00" level-info msg="init command" file="init.go:81" Command-bootstrap/init Component=ma-training-toolkit Platform=ModelArts-Service
time="2023-03-29T11:41:27+08:00" level-info msg="src is already running" file="init.go:81" Command-bootstrap/init Component=ma-training-toolkit Platform=ModelArts-Service
time="2023-03-29T11:41:27+08:00" level-info msg="[init] tool" file="init.go:81" Command-bootstrap/init Component=ma-training-toolkit Platform=ModelArts-Service
time="2023-03-29T11:41:27+08:00" level-info msg="[init] running" file="init.go:81" Command-bootstrap/init Component=ma-training-toolkit Platform=ModelArts-Service
time="2023-03-29T11:41:27+08:00" level-info msg="[init] ip or port" file="init.go:81" Command-bootstrap/init Component=ma-training-toolkit Platform=ModelArts-Service
time="2023-03-29T11:41:27+08:00" level-info msg="local dir = s" file="init.go:81" Command-bootstrap/init Component=ma-training-toolkit Platform=ModelArts-Service
time="2023-03-29T11:41:27+08:00" level-info msg="obs dir = s" file="init.go:81" Command-bootstrap/init Component=ma-training-toolkit Platform=ModelArts-Service
time="2023-03-29T11:41:27+08:00" level-info msg="num of workers = 8" file="upload.go:214" Command-obs/upload Component=ma-training-toolkit Platform=ModelArts-Service Task=
time="2023-03-29T11:41:27+08:00" level-info msg="start the periodic upload task, upload Period = 5 seconds" file="upload.go:220" Command-obs/upload Component=ma-training-toolkit Platform=ModelArts-Service Task=
time="2023-03-29T11:41:27+08:00" level-info msg="report event DetectStart success" file="event.go:63" Command-report Component=ma-training-toolkit Platform=ModelArts-Service
```

6.6.5 Obtención de eventos de trabajo de entrenamiento de ModelArts

Ejecute el comando **ma-cli ma-job get-event** para ver los eventos de trabajo de entrenamiento de ModelArts.

```
$ ma-cli ma-job get-event -h
Usage: ma-cli ma-job get-event [OPTIONS]

Get job running event.

Example:

# Get training job running event
ma-cli ma-job get-event --job-id ${job_id}

Options:
  -i, --job-id TEXT          Get training job event by job id. [required]
  -C, --config-file TEXT    Configure file path for authorization.
  -D, --debug                Debug Mode. Shows full stack trace when error occurs.
  -P, --profile TEXT        CLI connection profile to use. The default profile is "DEFAULT".
  -H, -h, --help            Show this message and exit.
```

Parámetro	Tipo	Obligatorio	Descripción
-i / --job-id	String	Sí	Obtener eventos de un trabajo de entrenamiento con un ID de trabajo especificado.

Ejemplos

Obtener eventos de un trabajo de entrenamiento con un ID de trabajo especificado.

```
ma-cli ma-job get-event --job-id b63e90baxxx
```

```
(PyTorch-1.4) [ma-user work]$ma-cli ma-job get-event --job-id b63e90ba-
-----
| STAT | INFO | TIME |
-----
| [ ] | Training job completed. | 2023-03-29T11:41:26+08:00 |
| [OK] | | 2:47:48:00 |
| [ ] | [worker-0][time used: 0.136s] Upload training output(parameter name: output_url) finished. | 2023-03-29T11:41:26+08:00 |
| [ ] | | 2:42:48:00 |
| [OK] | | 2:42:48:00 |
| [ ] | [worker-0] Training output(parameter name: output_url) Uploading. | 2023-03-29T11:41:26+08:00 |
| [ ] | | 2:42:48:00 |
| [OK] | [Job: modelarts-job-b63e90ba-xxxx] ExecuteAction: Start to execute action CompleteJob | 2023-03-29T11:41:26+08:00 |
| [ ] | | 2:42:48:00 |
| [OK] | [worker-0] Training finished. Exit code 0. | 2023-03-29T11:41:26+08:00 |
| [ ] | | 2:48:48:00 |
| [OK] | [worker-0] training started. | 2023-03-29T11:41:26+08:00 |
| [ ] | | 1:38:48:00 |
```

6.6.6 Obtención de motores de IA de ModelArts para entrenamiento

Ejecute el comando **ma-cli ma-job get-engine** para obtener motores de IA de ModelArts para entrenamiento.

```
$ ma-cli ma-job get-engine -h
Usage: ma-cli ma-job get-engine [OPTIONS]

Get job engine info.

Example:

# Get training job engines
ma-cli ma-job get-engine

Options:
  -v, --verbose           Show detailed information about training engines.
  -C, --config-file TEXT Configure file path for authorization.
  -D, --debug            Debug Mode. Shows full stack trace when error occurs.
  -P, --profile TEXT     CLI connection profile to use. The default profile is
  "DEFAULT".
  -H, -h, --help        Show this message and exit.
```

Tabla 6-16 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
-v / --verbose	Bool	No	Si se debe mostrar información detallada. Esta función está deshabilitada por defecto.

Ejemplos

Consultar el motor de IA de un trabajo de entrenamiento.

```
ma-cli ma-job get-engine
```

```
(PyTorch-1.4) [ma-user work]$ma-cli ma-job get-engine
```

Index	engine id	engine name	run user
1	caffe-1.0.0-python2.7	Caffe	
2	horovod-cp36-tf-1.16.2	Horovod	
3	horovod_0.20.0-pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64	Horovod	1102
4	horovod_0.20.0-tensorflow_2.1.0-cuda_10.1-py_3.7-ubuntu_18.04-x86_64	Horovod	1102
5	kungfu-0.2.2-tf-1.13.1-python3.6	KungFu	
6	mindspore_1.3.0-cuda_10.1-py_3.7-ubuntu_1804-x86_64	MPI	1102
7	mindspore_1.7.0-cann_5.1.0-py_3.7-euler_2.8.3-aarch64	Ascend-Powered-Engine	1000
8	mindspore_1.8.0-cann_5.1.2-py_3.7-euler_2.8.3-aarch64	Ascend-Powered-Engine	1000
9	mindspore_1.9.0-cann_6.0.0-py_3.7-euler_2.8.3-aarch64	Ascend-Powered-Engine	1000
10	mxnet-1.2.1-python3.6	MXNet	
11	optverse_0.2.0-pygrassland_1.1.0-py_3.7-ubuntu_18.04-x86_64	OR	1000
12	pytorch-cp36-1.0.0	PyTorch	
13	pytorch-cp36-1.3.0	PyTorch	
14	pytorch-cp36-1.4.0	PyTorch	
15	pytorch_1.8.0-cann_5.1.0-py_3.7-euler_2.8.3-aarch64	Ascend-Powered-Engine	1000
16	pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64	PyTorch	1000
17	pytorch_1.8.1-cann_5.1.2-py_3.7-euler_2.8.3-aarch64	Ascend-Powered-Engine	1000
18	pytorch_1.8.1-cann_6.0.0-py_3.7-euler_2.8.3-aarch64	Ascend-Powered-Engine	1000
19	pytorch_1.8.1-cuda_11.1-py_3.7-ubuntu_18.04-x86_64	PyTorch	1000
20	pytorch_1.8.2-cuda_10.2-py_3.7-ubuntu_18.04-x86_64	PyTorch	1000
21	pytorch_1.9.1-cuda_11.1-py_3.7-ubuntu_18.04-x86_64	PyTorch	1000
22	ray-cp36-0.7.4	Ray	

6.6.7 Obtención de especificaciones de recursos de ModelArts para entrenamiento

Ejecute el comando **ma-cli ma-job get-flavor** para obtener las especificaciones de recursos de ModelArts para el entrenamiento.

```
$ ma-cli ma-job get-flavor -h
Usage: ma-cli ma-job get-flavor [OPTIONS]

Get job flavor info.

Example:

# Get training job flavors
ma-cli ma-job get-flavor

Options:
  -t, --flavor-type [CPU|GPU|Ascend]  Type of training job flavor.
  -v, --verbose                          Show detailed information about training
flavors.
  -C, --config-file TEXT                 Configure file path for authorization.
  -D, --debug                             Debug Mode. Shows full stack trace when error
occurs.
  -P, --profile TEXT                     CLI connection profile to use. The default
profile is "DEFAULT".
  -H, -h, --help                          Show this message and exit.
```

Tabla 6-17 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
-t / --flavor-type	String	No	Variante de recursos. Si no se especifica este parámetro, se devuelven todas las variantes de recurso de forma predeterminada.
-v / --verbose	Bool	No	Si se debe mostrar información detallada. Esta función está deshabilitada por defecto.

Ejemplos

Consultar la variante de recurso y el tipo de un trabajo de entrenamiento.

```
ma-cli ma-job get-flavor
```

```
(PyTorch-1.4) [ma-user work]$ma-cli ma-job get-flavor
```

index	flavor id	flavor name	flavor type
1	modelarts.kat1.8xlarge	Computing NPU(8*Ascend) instance	Ascend
2	modelarts.kat1.xlarge	Computing NPU(Ascend) instance	Ascend
3	modelarts.vm.cpu.2u	Computing CPU(2U) instance	CPU
4	modelarts.vm.cpu.8u	Computing CPU(8U) instance	CPU
5	modelarts.vm.cpu.8u16g.119	Computing CPU(8U) instance	CPU
6	modelarts.vm.v100.large	Computing GPU(V100) instance	GPU
7	modelarts.vm.v100.large.free	Computing GPU(V100) instance	GPU

6.6.8 Detención de un trabajo de entrenamiento de ModelArts

Ejecute el comando **ma-cli ma-job stop** para detener un trabajo de entrenamiento con un ID de trabajo especificado.

```
$ ma-cli ma-job stop -h
Usage: ma-cli ma-job stop [OPTIONS]

Stop training job by job id.

Example:

Stop training job by job id
```

```
ma-cli ma-job stop --job-id ${job_id}

Options:
-i, --job-id TEXT      Get training job event by job id. [required]
-y, --yes              Confirm stop operation.
-C, --config-file TEXT Configure file path for authorization.
-D, --debug            Debug Mode. Shows full stack trace when error occurs.
-P, --profile TEXT     CLI connection profile to use. The default profile is
"DEFAULT".
-H, -h, --help        Show this message and exit.
```

Tabla 6-18 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
-i / --job-id	String	Sí	ID de trabajo de entrenamiento
-y / --yes	Bool	No	Si se debe detener por la fuerza un trabajo de entrenamiento específico

Ejemplos

Detener un trabajo de entrenamiento en ejecución.

```
ma-cli ma-job stop --job-id efd3e2f8xxx
```

6.7 Uso del comando ma-cli dli-job para enviar un trabajo de DLI Spark

6.7.1 Descripción

```
$ma-cli dli-job -h
Usage: ma-cli dli-job [OPTIONS] COMMAND [ARGS]...

DLI spark job submission and query job details.

Options:
-h, -H, --help Show this message and exit.

Commands:
get-job      Get DLI spark job details.
get-log      Get DLI spark log details.
get-queue    Get DLI spark queues info.
get-resource Get DLI resources info.
stop         Stop DLI spark job by job id.
submit       Submit dli spark batch job.
upload       Upload local file or OBS object to DLI resources.
```

Tabla 6-19 Comandos para enviar trabajos de DLI Spark

Comando	Descripción
get-job	Obtenga trabajos de DLI Spark y sus detalles.
get-log	Obtener logs de tiempo de ejecución de un trabajo de DLI Spark.
get-queue	Obtener colas de DLI.
get-resource	Obtener recursos del grupo de DLI.
stop	Detener un trabajo de DLI Spark.
submit	Enviar un trabajo de DLI Spark.
upload	Cargar archivos locales o archivos de OBS en un grupo DLI.

6.7.2 Consulta de trabajos de DLI Spark

Ejecute **ma-cli dli-job get-job** para consultar la lista de trabajos de DLI Spark o los detalles de un trabajo.

```
ma-cli dli-job get-job -h
Usage: ma-cli dli-job get-job [OPTIONS]

  Get DLI Spark details.

  Example:

  # Get DLI Spark job details by job name
  ma-cli dli-job get-job -n ${job_name}

  # Get DLI Spark job details by job id
  ma-cli dli-job get-job -i ${job_id}

  # Get DLI Spark job list
  ma-cli dli-job get-job --page-size 5 --page-num 1

Options:
  -i, --job-id TEXT           Get DLI Spark job details by job id.
  -n, --job-name TEXT        Get DLI Spark job details by job name.
  -pn, --page-num INTEGER RANGE Specify which page to query. [x>=1]
  -ps, --page-size INTEGER RANGE The maximum number of results for this query.
[x>=1]
  -v, --verbose               Show detailed information about DLI Spark job
details.
  -C, --config-file PATH     Configure file path for authorization.
  -D, --debug                 Debug Mode. Shows full stack trace when error
occurs.
  -P, --profile TEXT         CLI connection profile to use. The default
profile is "DEFAULT".
  -H, -h, --help             Show this message and exit.
```


Tabla 6-20 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
-i / --job-id	String	No	ID de un trabajo de DLI Spark utilizado para obtener detalles del trabajo
-n / --job-name	String	No	Nombre de un trabajo de DLI Spark utilizado para consultar el trabajo o las palabras clave contenidas en los nombres de trabajo utilizados para filtrar trabajos de DLI Spark
-pn / --page-num	Integer	No	Página de índice de trabajos. El valor predeterminado es la página 1.
-ps / --page-size	Integer	No	Número de trabajos mostrados en cada página. El valor predeterminado es 20 .
-v / --verbose	Bool	No	Si se debe mostrar información detallada. Esta función está deshabilitada por defecto.

Ejemplo

Ejecute el siguiente comando para consultar todos los trabajos de DLI Spark:

```
ma-cli dli-job get-job
```

```
(PyTorch-1.8) [ma-user work]$ma-cli dli-job get-job
```

index	id	name	status	queue	sc_type	image
1	15c87f3a-973e	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
2	656dd759-b04e	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
3	1a193b8d-335f	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
4	12fbcc37-8df6	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
5	794dfd57-bb2e	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
6	76a3aa43-43bf	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
7	82856087-5bd3	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
8	095c0c3f-b0c5	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
9	a2324e0f-81e1	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
10	d70717e2-1a36	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
11	85358931-99af	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
12	d5546f21-430e	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
13	7b3b9fac-0141	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
14	2495b20b-4c2c	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
15	59924d24-ef02	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
16	dab5d88f-cdb6	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
17	eff42ca1-074e	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
18	9357a261-72df	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
19	e5157750-59cc	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook
20	7b273ef2-8e52	zh	dead	dli_ma_notebook	CUSTOMIZED	notebook

6.7.3 Envío de un trabajo de DLI Spark

Ejecute el comando **ma-cli dli-job submit** para enviar un trabajo de DLI Spark.

Antes de ejecutar este comando, configure **YAML_FILE** para especificar la ruta al archivo de configuración del trabajo de destino. Si no se especifica este parámetro, el archivo de configuración estará vacío. El archivo de configuración está en formato de YAML y sus parámetros son el parámetro **option** del comando. Si especifica tanto el archivo de configuración **YAML_FILE** como el parámetro **option** en la CLI, el valor del parámetro **option** lo sobrescribirá en el archivo de configuración.

Parámetros de CLI

```
ma-cli dli-job submit -h
Usage: ma-cli dli-job submit [OPTIONS] [YAML_FILE]...

Submit DLI Spark job.

Example:

ma-cli dli-job submit --name test-spark-from-sdk
                    --file test/sub_dli_task.py
                    --obs-bucket dli-bucket
                    --queue dli_test
                    --spark-version 2.4.5
                    --driver-cores 1
                    --driver-memory 1G
                    --executor-cores 1
                    --executor-memory 1G
                    --num-executors 1

Options:
  --file TEXT           Python file or app jar.
  -cn, --class-name TEXT Your application's main class (for Java / Scala
apps).
  --name TEXT           Job name.
```

<code>--image TEXT</code>	Full swr custom image path.
<code>--queue TEXT</code>	Execute queue name.
<code>-obs, --obs-bucket TEXT</code>	DLI obs bucket to save logs.
<code>-sv, --spark-version TEXT</code>	Spark version.
<code>-st, --sc-type [A B C]</code>	Compute resource type.
<code>--feature [basic custom ai]</code>	Type of the Spark image used by a job (default: basic).
<code>-ec, --executor-cores INTEGER</code>	Executor cores.
<code>-em, --executor-memory TEXT</code>	Executor memory (eg. 2G/2048MB).
<code>-ne, --num-executors INTEGER</code>	Executor number.
<code>-dc, --driver-cores INTEGER</code>	Driver cores.
<code>-dm, --driver-memory TEXT</code>	Driver memory (eg. 2G/2048MB).
<code>--conf TEXT</code>	Arbitrary Spark configuration property (eg. <PROP=VALUE>).
<code>--resources TEXT</code>	Resources package path.
<code>--files TEXT</code>	Files to be placed in the working directory of each executor.
<code>--jars TEXT</code>	Jars to include on the driver and executor class paths.
<code>-pf, --py-files TEXT</code>	Python files to place on the PYTHONPATH for Python apps.
<code>--groups TEXT</code>	User group resources.
<code>--args TEXT</code>	Spark batch job parameter args.
<code>-q, --quiet</code>	Exit without waiting after submit successfully.
<code>-C, --config-file PATH</code>	Configure file path for authorization.
<code>-D, --debug</code>	Debug Mode. Shows full stack trace when error occurs.
<code>-P, --profile TEXT</code>	CLI connection profile to use. The default profile is "DEFAULT".
<code>-H, -h, --help</code>	Show this message and exit.

Vista previa del archivo de YAML

```
# dli-demo.yaml
name: test-spark-from-sdk
file: test/sub_dli_task.py
obs-bucket: ${your_bucket}
queue: dli_notebook
spark-version: 2.4.5
driver-cores: 1
driver-memory: 1G
executor-cores: 1
executor-memory: 1G
num-executors: 1

## [Optional]
jars:
  - ./test.jar
  - obs://your-bucket/jars/test.jar
  - your_group/test.jar

## [Optional]
files:
  - ./test.csv
  - obs://your-bucket/files/test.csv
  - your_group/test.csv

## [Optional]
python-files:
  - ./test.py
  - obs://your-bucket/files/test.py
  - your_group/test.py

## [Optional]
resources:
  - name: your_group/test.py
    type: pyFile
  - name: your_group/test.csv
    type: file
```

```

- name: your_group/test.jar
  type: jar
- name: ./test.py
  type: pyFile
- name: obs://your-bucket/files/test.py
  type: pyFile

## [Optional]
groups:
- group1
- group2

```

Ejemplo de envío de un trabajo de DLI Spark con **options** especificado:

```

$ ma-cli dli-job submit --name test-spark-from-sdk \
  --file test/sub_dli_task.py \
  --obs-bucket ${your_bucket} \
  --queue dli_test \
  --spark-version 2.4.5 \
  --driver-cores 1 \
  --driver-memory 1G \
  --executor-cores 1 \
  --executor-memory 1G \
  --num-executors 1

```

Tabla 6-21 Descripción

Parámetro	Tipo	Obligatorio	Descripción
YAML_FILE	String, una ruta de acceso de archivo local	No	Archivo de configuración de un trabajo de DLI Spark. Si no se especifica este parámetro, el archivo de configuración estará vacío.
--file	String	Sí	Archivo de entrada para ejecutar el programa. El valor puede ser una ruta de archivo local, una ruta de OBS o el nombre de un paquete de JAR o de PyFile cargado en el sistema de gestión de recursos de DLI.
-cn / --class_name	String	Sí	Clase principal de Java/Spark del trabajo de procesamiento por lotes.
--name	String	No	Nombre del trabajo especificado. El valor tiene un máximo de 128 caracteres.
--image	String	No	Ruta de acceso a una imagen personalizada con el formato "Organization name/Image name:Image version". Este parámetro solo es válido cuando feature se establece en custom . Puede utilizar este parámetro con el parámetro feature para especificar una imagen de Spark personalizada para ejecutar el trabajo.

Parámetro	Tipo	Obligatorio	Descripción
-obs / --obs-bucket	String	No	Bucket de OBS para almacenar un trabajo de Spark. Configure este parámetro cuando necesite guardar trabajos. También se puede utilizar como estación de tránsito para enviar archivos locales a los recursos.
-sv / --spark-version	String	No	Versión del componente de Spark utilizada por un trabajo.
-st / `--sc-type	String	No	Si la versión actual del componente de Spark es 2.3.2, deje este parámetro en blanco. Si la versión actual del componente de Spark es 2.3.3, configure este parámetro cuando feature se establezca en basic o ai . Si no se especifica este parámetro, se utilizará la versión 2.3.2 por defecto del componente de Spark.
--feature	String	No	Función de trabajo, que indica el tipo de imagen de Spark utilizada por un trabajo. El valor predeterminado es basic . <ul style="list-style-type: none"> ● basic: se utiliza una imagen de Spark básica proporcionada por DLI. ● custom: se utiliza una imagen de Spark personalizada. ● ai: se utiliza una imagen de IA proporcionada por DLI.
--queue	String	No	Nombre de la cola. Configure este parámetro como el nombre de una cola de DLI creada. La cola debe ser del tipo común. Para obtener detalles sobre cómo obtener un nombre de cola, véase Tabla 6-23 .
-ec / --executor-cores	String	No	Cantidad de núcleos de CPU de cada ejecutor en la aplicación de Spark. Esta configuración reemplazará la predeterminada de sc_type .
-em / --executor-memory	String	No	Memoria del ejecutor de la aplicación de Spark, por ejemplo, 2 GB o 2048 MB . Esta configuración reemplazará la predeterminada de sc_type . Se debe proveer la unidad. De lo contrario, el inicio falla.
-ne / --num-executors	String	No	Número de ejecutores en una aplicación de Spark. Esta configuración reemplazará la predeterminada de sc_type .
-dc / --driver-cores	String	No	Número de núcleos de CPU del controlador de la aplicación de Spark. Esta configuración reemplazará la predeterminada de sc_type .

Parámetro	Tipo	Obligatorio	Descripción
-dm / --driver-memory	String	No	Memoria del controlador de la aplicación de Spark, por ejemplo, 2 GB o 2048 MB . Esta configuración reemplazará la predeterminada de sc_type . Se debe proveer la unidad. De lo contrario, el inicio falla.
--conf	Array of string	No	Configuración de Batch . Para obtener más detalles, véase Configuración de Spark . Para especificar varios parámetros, utilice --conf conf1 --conf conf2 .
--resources	Array of string	No	Nombre de un paquete de recursos, que puede ser un archivo local, una ruta de OBS o un archivo cargado en el sistema de gestión de recursos de DLI. Para especificar varios parámetros, utilice --resources resource1 --resources resource2 .
--files	Array of string	No	Nombre del paquete de archivos que se cargó en el sistema de gestión de recursos de DLI. También puede especificar una ruta de OBS, por ejemplo, obs://Bucket name/Package name . También se admiten archivos locales. Para especificar varios parámetros, utilice --files file1 --files file2 .
--jars	Array of string	No	Nombre del paquete de JAR que se ha cargado en el sistema de gestión de recursos de DLI. También puede especificar una ruta de OBS, por ejemplo, obs://Bucket name/Package name . También se admiten archivos locales. Para especificar varios parámetros, utilice --jars jar1 --jars jar2 .
-pf / --python-files	Array of string	No	Nombre del paquete de PyFile que se ha cargado en el sistema de gestión de recursos de la DLI. También puede especificar una ruta de OBS, por ejemplo, obs://Bucket name/Package name . También se admiten archivos locales. Para especificar varios parámetros, utilice --python-files py1 --python-files py2 .
--groups	Array of string	No	Nombre del grupo de recursos. Para especificar varios parámetros, utilice --groups group1 --groups group2 .
--args	Array of string	No	Parámetros de entrada de la clase principal, que son parámetros de aplicación. Para especificar varios parámetros, utilice --args arg1 --args arg2 .
-q / --quiet	Bool	No	Después de enviar un trabajo de DLI Spark, el sistema sale directamente y no imprime el estado del trabajo de manera sincrónica.

Ejemplos

- Envíe un trabajo de DLI Spark mediante el archivo **YAML_FILE**.

```
$ma-cli dli-job submit dli_job.yaml
```

```
(PyTorch-1.4) [ma-user work]$ma-cli dli-job submit ./dli-job.yaml
[ OK ] Current DLI job id is: 01b698b8-9fd6-4a8e-bc3c-6821c6405b14
[ OK ] starting
[ OK ] running
[ OK ] success
[ OK ] Successfully submit DLI spark job [ 01b698b8-9fd6-4a8e-bc3c-6821c6405b14 ].
```

- Enviar un trabajo de DLI Spark especificando el parámetro **options** en la CLI.

```
$ma-cli dli-job submit --name test-spark-from-sdk \
> --file test/jumpstart-trainingjob-gallery-pytorch-sample.ipynb \
> --queue dli_ma_notebook \
> --spark-version 2.4.5 \
> --driver-cores 1 \
> --driver-memory 1G \
> --executor-cores 1 \
> --executor-memory 1G \
> --num-executors 1
```

```
(PyTorch-1.4) [ma-user work]$ma-cli dli-job submit --name test-spark-from-sdk \
> --file test/jumpstart-trainingjob-gallery-pytorch-sample.ipynb \
> --queue dli_ma_notebook \
> --spark-version 2.4.5 \
> --driver-cores 1 \
> --driver-memory 1G \
> --executor-cores 1 \
> --executor-memory 1G \
> --num-executors 1
[ OK ] Current DLI job id is: ae856c20-e9ae-49ca-8409-7a02652297b8
[ OK ] starting
```

6.7.4 Consulta de logs de ejecución de DLI Spark

Ejecute **ma-cli dli-job get-log** para consultar los logs de backend de los trabajos de DLI Spark.

```
$ ma-cli dli-job get-log -h
Usage: ma-cli dli-job get-log [OPTIONS]

Get DLI spark job log details.

Example:

# Get job log by job id
ma-cli dli-job get-log --job-id ${job_id}

Options:
  -i, --job-id TEXT           Get DLI spark job details by job id. [required]
  -C, --config-file TEXT     Configure file path for authorization.
  -D, --debug                 Debug Mode. Shows full stack trace when error occurs.
  -P, --profile TEXT         CLI connection profile to use. The default profile is "DEFAULT".
  -H, -h, --help             Show this message and exit.
```

Tabla 6-22 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
-i / --job-id	String	Sí	ID de un trabajo de DLI Spark utilizado para obtener logs de trabajo

Ejemplo

Ejecute el siguiente comando para obtener registros de ejecución de un trabajo de DLI Spark mediante su ID de trabajo:

```
ma-cli dli-job get-log --job-id ${your_job_id}
```

```
(PyTorch-1.8) [ma-user work]$ma-cli dli-job get-log --job-id 7b273ef2-8e5
driver:~ umask 027
++ id -u
+ myuid=2010
++ id -g
+ mygid=2010
+ set +e
++ getent
+ uidentry                               'bin/bash'
+ set -e
+ '[' -z o                               'n/bash ']'
+ SPARK_CL
+ grep SPA
+ sort -t
+ sed 's/[
+ env
+ readarra
+ '[' -n '
+ '[' 3 ==
+ '[' 3 ==
++ python3
+ pyv3='Py
+ export P
+ PYTHON_V
+ export P
+ PYSPARK
+ export P
+ PYSPARK
+ '[' -z x
+ SPARK_CL                               'pt/spark/jars/*'
+ '[' -z '
+ case "$1
+ shift 1
+ CMD="(("$S                               --conf "spark.driver.bindAddress=${SPARK_DRIVER_BIND_ADDRESS}" --deploy-mode client "$@"
+ '[' true
+ '[' true
+ '[' -z x
+ '[' -z x
+ exec /us                               in/spark-submit --conf spark.driver.bindAddress=172.30.0.62 --deploy-mode client --properties-
oy_PythonR                               ib/submit_notebook.py
++ tee -a
++ tee -a
++ sed -u -e 's/[0-9;]*m/g' -e 's/\x1b//g'
```

6.7.5 Consulta de colas de DLI

Ejecute **ma-cli dli-job get-queue** para consultar las colas de DLI.

```
ma-cli dli-job get-queue -h
Usage: ma-cli dli-job get-queue [OPTIONS]

Get DLI queues info.

Example:

# Get DLI queue details by queue name
ma-cli dli-job get-queue --queue-name $queue_name}

Options:
```



```

-pn, --page-num INTEGER RANGE Specify which page to query. [x>=1]
-ps, --page-size INTEGER RANGE The maximum number of results for this query.
[x>=1]
-n, --queue-name TEXT Get DLI queue details by queue name.
-t, --queue-type [sql|general|all] DLI queue type (default "all").
-tags, --tags TEXT Get DLI queues by tags.
-C, --config-file PATH Configure file path for authorization.
-D, --debug Debug Mode. Shows full stack trace when error
occurs.
-P, --profile TEXT CLI connection profile to use. The default
profile is "DEFAULT".
-H, -h, --help Show this message and exit.

```

Tabla 6-23 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
-n / --queue-name	String	No	Nombre de una cola de DLI que se va a consultar
-t / --queue-type	String	No	Tipo de colas de DLI a consultar. El valor puede ser sql , general o all . El valor predeterminado es all .
-tags / --tags	String	No	Etiquetas de colas de DLI que se van a consultar
-pn / --page-num	Integer	No	Índice de la página de cola de DLI. El valor predeterminado es la página 1.
-ps / --page-size	Integer	No	Cantidad de colas de DLI mostradas en cada página. El valor predeterminado es 20 .

Ejemplo

Ejecute el siguiente comando para consultar información sobre la cola **dli_ma_notebook**:

```
ma-cli dli-job get-queue --queue-name dli_ma_notebook
```

```
(PyTorch-1.8) [ma-user work]$ ma-cli dli-job get-queue --queue-name dli_ma_notebook
{'chargingMode': 1,
 'create_time': 1668585417422,
 'cuCount': 16,
 'cu_spec': 16,
 'description': '',
 'enterprise_project_id': '0',
 'is_success': True,
 'message': '',
 'owner': '...',
 'queueName': 'dli_ma_notebook',
 'queueType': 'general',
 'queue_id': 8...,
 'resource_id': '242e7af8-c9d1...',
 'resource_mode': 1,
 'resource_type': 'container',
 'support_spark_versions': ['2.3.2', '2.4.5', '3.1.1']}
```

6.7.6 Obtención de recursos del grupo de DLI

Ejecute el comando **ma-cli dli-job get-resource** para obtener detalles sobre los recursos de DLI, como el nombre y el tipo de recurso.

```
$ ma-cli dli-job get-resource -h
Usage: ma-cli dli-job get-resource [OPTIONS]

Get DLI resource info.

Example:

# Get DLI resource details by resource name
ma-cli dli-job get-resource --resource-name ${resource_name}

Options:
  -n, --resource-name TEXT          Get DLI resource details by resource name.
  -k, --kind [jar|pyFile|file|modelFile]
                                     DLI resources type.
  -g, --group TEXT                  Get DLI resources by group.
  -tags, --tags TEXT                Get DLI resources by tags.
  -C, --config-file TEXT            Configure file path for authorization.
  -D, --debug                        Debug Mode. Shows full stack trace when error
occurs.
  -P, --profile TEXT                CLI connection profile to use. The default
profile is "DEFAULT".
  -H, -h, --help                    Show this message and exit.
```

Tabla 6-24 Parámetros

Parámetro	Tipo	Obligato rio	Descripción
-n / --resource-name	String	No	Obtenga detalles sobre los recursos del grupo de DLI por nombre de recurso.
-k / --kind	String	No	Obtenga detalles sobre los recursos del grupo de DLI por tipo de recurso, que pueden ser JAR, PyFile o modelFile.
-g / --group	String	No	Obtenga detalles acerca de los recursos del grupo de DLI por nombre de grupo.

Parámetro	Tipo	Obligato rio	Descripción
-tags / --tags	String	No	Obtenga detalles sobre los recursos del grupo de DLI por etiqueta de recurso.

Ejemplos

Obtener todos los recursos del grupo de DLI.

```
ma-cli dli-job get-resource
```

```
(PyTorch-1.4) [ma-user work]$ma-cli dli-job get-resource
{'groups': [{'create_time': 1679561988580,
  'details': [{'create_time': 1679561988692,
    'owner': 'ei_...',
    'resource_name': 'Untitled.ipynb',
    'resource_type': 'file',
    'status': 'READY',
    'underlying_name': 'Untitled.ipynb',
    'update_time': 1679561989683}],
  'group_name': 'mrn',
  'is_async': False,
  'owner': 'ei_...',
  'resources': ['Untitled.ipynb'],
  'status': 'READY',
  'update_time': 1679561989683},
  {'create_time': 1679561437096,
  'details': [{'create_time': 1679561437233,
    'owner': 'ei_...',
    'resource_name': 'jumpstart-trainingjob-gallery-pytorch-sample.ipynb',
    'resource_type': 'file',
    'status': 'READY',
    'underlying_name': 'jumpstart-trainingjob-gallery-pytorch-sample.ipynb',
    'update_time': 1679561438810},
    {'create_time': 1679561929606,
    'owner': 'ei_...',
    'resource_name': 'Untitled.ipynb',
    'resource_type': 'file',
    'status': 'READY',
    'underlying_name': 'Untitled.ipynb',
    'update_time': 1679561930312}],
  'group_name': 'test',
  'is_async': False,
  'owner': 'ei_...',
  'resources': ['jumpstart-trainingjob-gallery-pytorch-sample.ipynb',
    'Untitled.ipynb'],
  'status': 'READY',
  'update_time': 1679561930312}],
  'modules': [{'create_time': 1560249470326,
    'description': '',
    'module_name': 'sys.dli.test',
    'module_type': 'jar',
    'resources': [],
    'status': 'READY',
    'update_time': 1560249470339},
    {'create_time': 1564118513494,
    'description': '...',
    'module_name': 'sys.dli.module',
    'module_type': 'jar',
    'resources': ['spark-examples 2.11-2.1.0.luxor.jar']}]}
```

6.7.7 Carga de archivos locales o archivos de OBS en un grupo de DLI

Ejecute el comando **ma-cli dli-job upload** para cargar archivos locales o archivos de OBS en un grupo de DLI.

```
$ ma-cli dli-job upload -h
Usage: ma-cli dli-job upload [OPTIONS] PATHS...

Upload DLI resource.

Tips: --obs-path is need when upload local file.

Example:

# Upload an OBS path to DLI resource
ma-cli dli-job upload obs://your-bucket/test.py -g test-group --kind pyFile

# Upload a local path to DLI resource
ma-cli dli-job upload ./test.py -g test-group -obs ${your-bucket} --kind pyFile

# Upload local path and OBS path to DLI resource
ma-cli dli-job upload ./test.py obs://your-bucket/test.py -g test-group -obs ${your-bucket}

Options:
-k, --kind [jar|pyFile|file]  DLI resources type.
-g, --group TEXT              DLI resources group.
-tags, --tags TEXT           DLI resources tags, follow --tags `key1`=`value1`.
-obs, --obs-bucket TEXT      OBS bucket for upload local file.
-async, --is-async           whether to upload resource packages in asynchronous mode. The default value is False.
-C, --config-file TEXT       Configure file path for authorization.
-D, --debug                  Debug Mode. Shows full stack trace when error occurs.
-P, --profile TEXT           CLI connection profile to use. The default profile is "DEFAULT".
-H, -h, --help              Show this message and exit.
```

Tabla 6-25 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
PATHS	String	Sí	Rutas de acceso a los archivos locales o archivos de OBS que se cargarán en un grupo de DLI. Se pueden especificar varias rutas a la vez.
-k / --kind	String	No	Tipo de archivo que se va a cargar, que puede ser JAR, PyFile o archivo
-g / --group	String	No	Nombre del grupo de DLI en el que se va a cargar el archivo
-tags / --tags	String	No	Etiqueta del archivo que se va a cargar
-obs / --obs-bucket	String	No	Si el archivo que se va a cargar contiene una ruta local, especifique un bucket de OBS para el tránsito.

6.8 Uso de ma-cli para copiar datos de OBS

Ejecute el comando **ma-cli obs-copy [SRC] [DST]** para copiar un archivo local en una carpeta de OBS o un archivo o carpeta de OBS en una ruta local.

```
$ma-cli obs-copy -h
Usage: ma-cli obs-copy [OPTIONS] SRC DST

Copy file or directory between OBS and local path. Example:

# Upload local file to OBS path
ma-cli obs-copy ./test.zip obs://your-bucket/copy-data/

# Upload local directory to OBS path
ma-cli obs-copy ./test/ obs://your-bucket/copy-data/

# Download OBS file to local path
ma-cli obs-copy obs://your-bucket/copy-data/test.zip ./test.zip

# Download OBS directory to local path
ma-cli obs-copy obs://your-bucket/copy-data/ ./test/

Options:
  -d, --drop-last-dir      Whether to drop last directory when copy folder. if
                           True, the last directory of the source folder will not copy to the destination
                           folder. [default: False]
  -C, --config-file PATH  Configure file path for authorization.
  -D, --debug              Debug Mode. Shows full stack trace when error occurs.
  -P, --profile TEXT      CLI connection profile to use. The default profile is
                           "DEFAULT".
  -H, -h, --help          Show this message and exit.
```

Tabla 6-27 Parámetros

Parámetro	Tipo	Obligatorio	Descripción
-d / --drop-last-dir	Bool	No	Si especifica este parámetro, el directorio de último nivel de la carpeta de origen no se copiará en la carpeta de destino. Este parámetro solo es válido para copiar carpetas.

Ejemplos

Cargar un archivo en OBS.

```
$ ma-cli obs-copy ./test.csv obs://${your_bucket}/test-copy/
[ OK ] local src path: [ /home/ma-user/work/test.csv ]
[ OK ] obs dst path: [ obs://${your_bucket}/test-copy/ ]
```

Cargar una carpeta en **obs://\${your_bucket}/test-copy/data/**.

```
$ ma-cli obs-copy /home/ma-user/work/data/ obs://${your_bucket}/test-copy/
[ OK ] local src path: [ /home/ma-user/work/data/ ]
[ OK ] obs dst path: [ obs://${your_bucket}/test-copy/ ]
```

Cargar una carpeta en **obs://\${your_bucket}/test-copy/** con **--drop-last-dir** especificado.

```
$ ma-cli obs-copy /home/ma-user/work/data/ obs://${your_bucket}/test-copy/ --drop-last-dir
```

```
[ OK ] local src path: [ /home/ma-user/work/data ]  
[ OK ] obs dst path: [ obs://{your_bucket}/test-copy/ ]
```

Descargar una carpeta de OBS a un disco local.

```
$ ma-cli obs-copy obs://{your_bucket}/test-copy/ ~/work/test-data/  
[ OK ] obs src path: [ obs://{your_bucket}/test-copy/ ]  
[ OK ] local dst path: [ /home/ma-user/work/test-data/ ]
```