Creating the Upload Model Parameters to Cloud Job

How-to

Compatible versions: 5.4, 5.5

Works with: Fusion

Table of Contents

- Deploy a model
- Create the job
 - GCS
 - S3
 - Azure Blob Storage
- Deploy a new model with the Upload Model Parameters job

In Fusion 5.4, you can use the Upload Model Parameters to Cloud job to store a trained model's training to a blob in cloud storage.

This document describes the steps to create the Upload Model Parameters to Cloud job and deploy a model with it.

See <u>Upload Model Parameters to Cloud</u> to learn more about the Upload Model Parameters to Cloud job.

Deploy a model

First, deploy a Smart Answers Supervised Model.

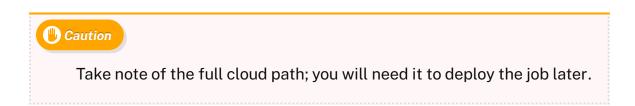
Create the job

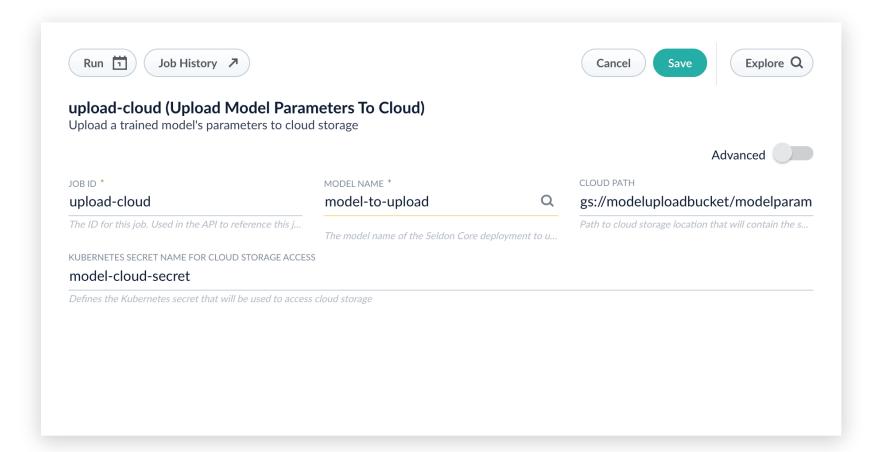
- In the Collections > Jobs menu, click Add + and select Upload Model Parameters to Cloud to create a new instance of that job.
- 2. Set the job ID to **upload-cloud** (or something appropriate for your model) and find the name of the deployed model in the **Model Name** text box.
- 3. Enter the Cloud Path to direct to the location you would like to store the parameters blob in.

You will also need to provide the Kubernetes secret that provides the access details for the

Lucidworks Fusion 5.5

The exported file will also have the version number (based on epoch time) of the model appended to it.





GCS

- Obtain the Google Cloud Storage credentials JSON file for accessing the storage bucket you intend to use for transferring models.
- 2. Add the credentials JSON file as a secret with the following command:

```
kubectl create secret generic model-cloud-secret --from-file=credentials.json -n [NAMESPACE]

Copy
```

S3

1. Create a JSON file with the following keys:

```
{
    "key": [KEY],
    "secret": [SECRET],
    "region": [REGION]
}
```

2. Upload the JSON file as a secret in the cluster:

```
kubectl create secret generic model-cloud-secret --from-file=model.json -n [NAMESPACE]

Copy
```

The secret can now be referenced as **model-cloud-secret** in the job configuration.

Azure Blob Storage

1. Create a JSON file with the layout below:

```
{
    "key": [KEY],
    "secret": [SECRET]
}
```

2. Create the model-cloud-secret secret in the Kubernetes cluster:

```
kubectl create secret generic model-cloud-secret --from-file=model.json -n [NAMESPACE]
Copy
```

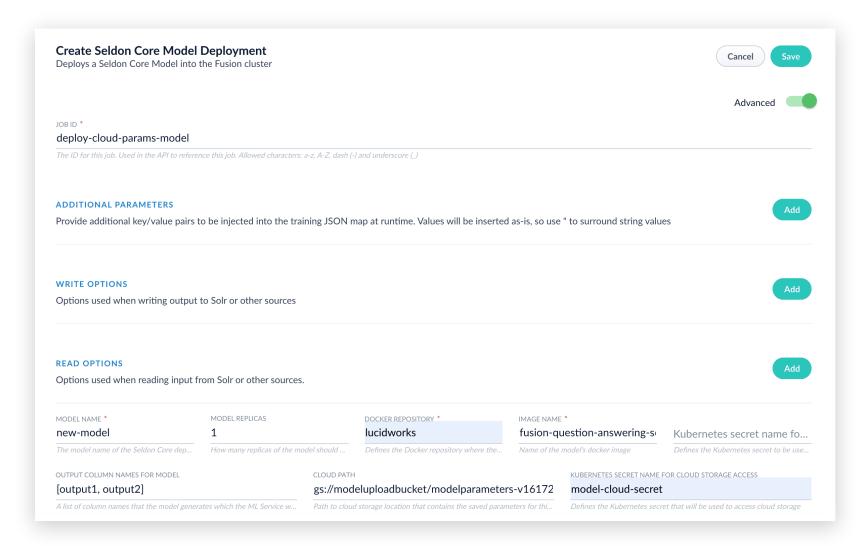




up the Kubernetes secrets in that cluster as well, though the access for deploying can be limited to read-only.

ou are planning on transferring a medic to a anneron elactor, note that jed mit heed to

- 1. Create a new Create Seldon Core Model Deployment job and toggle Advanced.
- 2. Enter the full cloud path, including the version number, in the *Cloud Path* box.
- 3. Enter the Kubernetes secret that will be used to access the path in the Kubernetes Secret Name for Cloud Storage Access box.
- 4. For this example, you will be using the job to deploy a trained Smart Answers model, so use the fusion-question-answering-seldon:5.4.0 model image from the lucidworks repository in the Image Name box.



4. Run the job. Confirm that the model deploys into the cluster and is running.