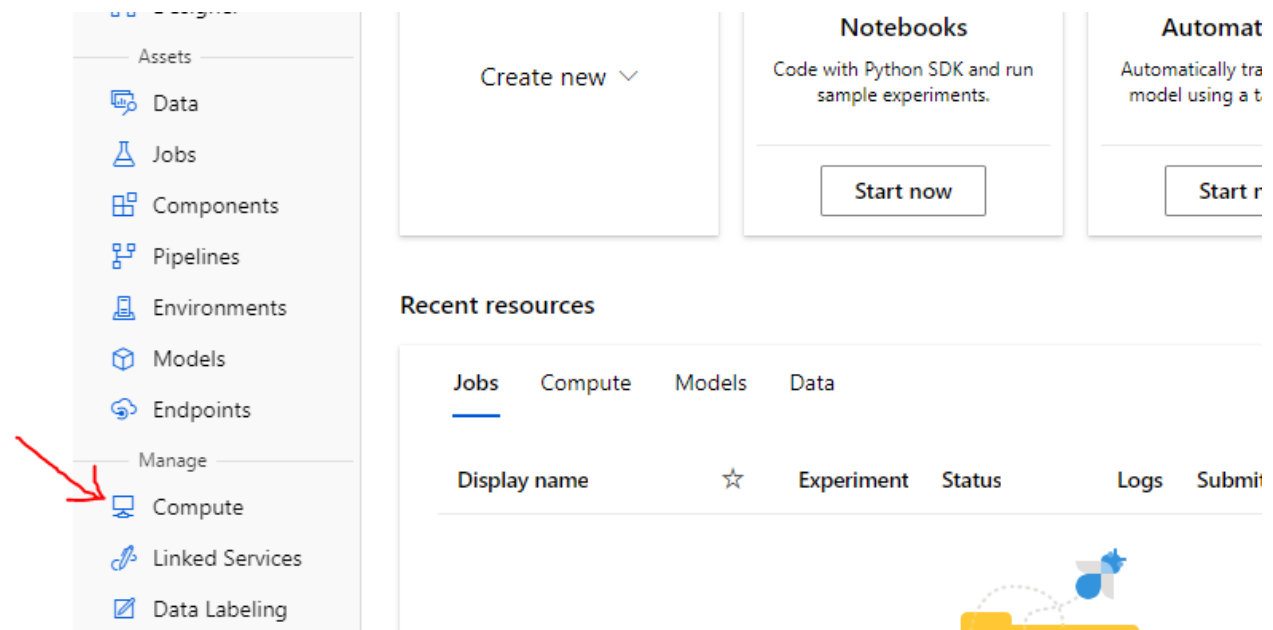


Step 1: Create a workspace in Azure Machine learning, wait till the workspace is created and then click on it.

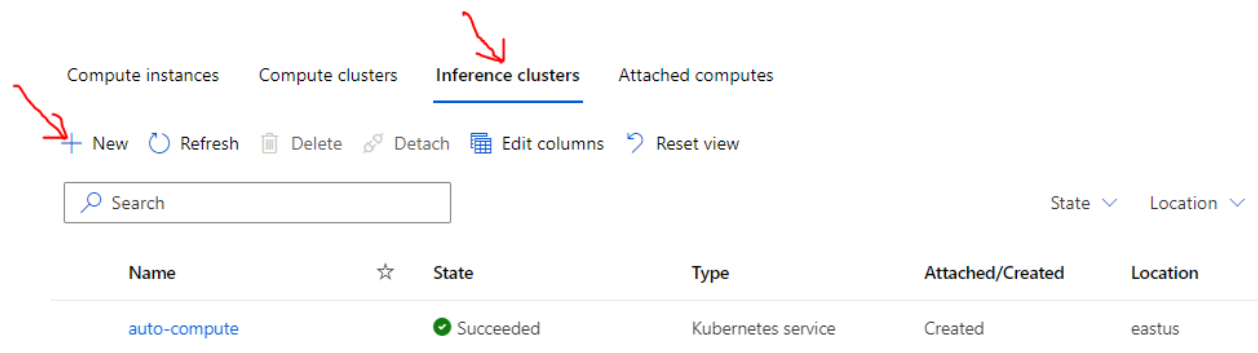
The screenshot shows the Microsoft Azure Machine Learning Studio interface. The top navigation bar includes the logo, a search bar, and the text 'All workspaces'. The left sidebar contains a menu with options: Default Directory, Workspaces (selected), Registries, Shared assets, Components, Environments, Models, Admin, and Quota. The main content area is titled 'Workspaces' and includes a description: 'A workspace provides a centralized place to keep track of all the artifacts you create while performing machine learning experiments.' Below the description are action buttons: '+ New', 'Refresh', 'Edit workspace', 'Edit columns', and 'Reset view'. A search bar is present, followed by filters for 'Region' and 'Subscription', a 'Private link enabled' toggle, and 'All filters' and 'Clear all' options. The table shows 'Showing 1-1 of 1 workspaces' and a 'Page size' dropdown set to '25'. The table has columns: 'Workspace name', 'Resource group', 'Region', 'Subscription', and 'Created'. The first row shows the workspace 'RobohonTextAlgorithm' in the 'robohon_text_algorithm' resource group, located in the 'eastus' region, under 'Azure subscription 1'. The 'Workspace name' column header and the workspace name itself are highlighted with a red circle and a red arrow.

Workspace name	Resource group	Region	Subscription	Created
RobohonTextAlgorithm	robohon_text_algorithm	eastus	Azure subscription 1	

Step 2: Inside that workspace, click on the “Compute” icon.



Step 3: Click on “inference clusters” and then click on “New”.



The screenshot shows a web interface for managing inference clusters. At the top, there are four tabs: 'Compute instances', 'Compute clusters', 'Inference clusters' (which is selected and underlined), and 'Attached computes'. A red arrow points to the 'Inference clusters' tab. Below the tabs, there is a toolbar with several icons and labels: a plus sign followed by 'New' (with a red arrow pointing to it), 'Refresh', 'Delete', 'Detach', 'Edit columns', and 'Reset view'. Below the toolbar is a search bar with a magnifying glass icon and the word 'Search'. To the right of the search bar are two dropdown menus labeled 'State' and 'Location'. Below these elements is a table with the following columns: 'Name', a star icon, 'State', 'Type', 'Attached/Created', and 'Location'. The table contains one row with the following data: 'auto-compute' (in blue text), a green checkmark icon, 'Succeeded', 'Kubernetes service', 'Created', and 'eastus'.

Name		State	Type	Attached/Created	Location
auto-compute	☆	✓ Succeeded	Kubernetes service	Created	eastus

Step 4: Click on “Create new” and then fill the respective fields and click on next button

☒ Virtual Machine

☐ Advanced Settings

Select the virtual machine size you would like to use for your inference cluster.

Kubernetes Service

☒ Create new ☐ Use existing

Location *

East US

+ Add filter

Showing 550 VM sizes | Current selection: Standard_A2_v2

Name ↑	Category	Available quota
<input checked="" type="radio"/> Standard_A2_v2 2 cores, 4GB RAM, 20GB storage	General purpose	4 cores
<input type="radio"/> Standard_A2m_v2 2 cores, 16GB RAM, 20GB storage	General purpose	4 cores
<input type="radio"/> Standard_A4_v2	General purpose	4 cores

Back

Next

Step 5: Just fill these fields as well and then click on create. This will create the inference cluster.

☒ Virtual Machine

☒ Advanced Settings

Name	Category	Cores	Available quota	RAM	Storage
Standard_A2_v2	General purpose	2	4 cores	4 GB	20 GB

Compute name * ⓘ

name-of-compute

Cluster purpose

☒ Production ☐ Dev-test

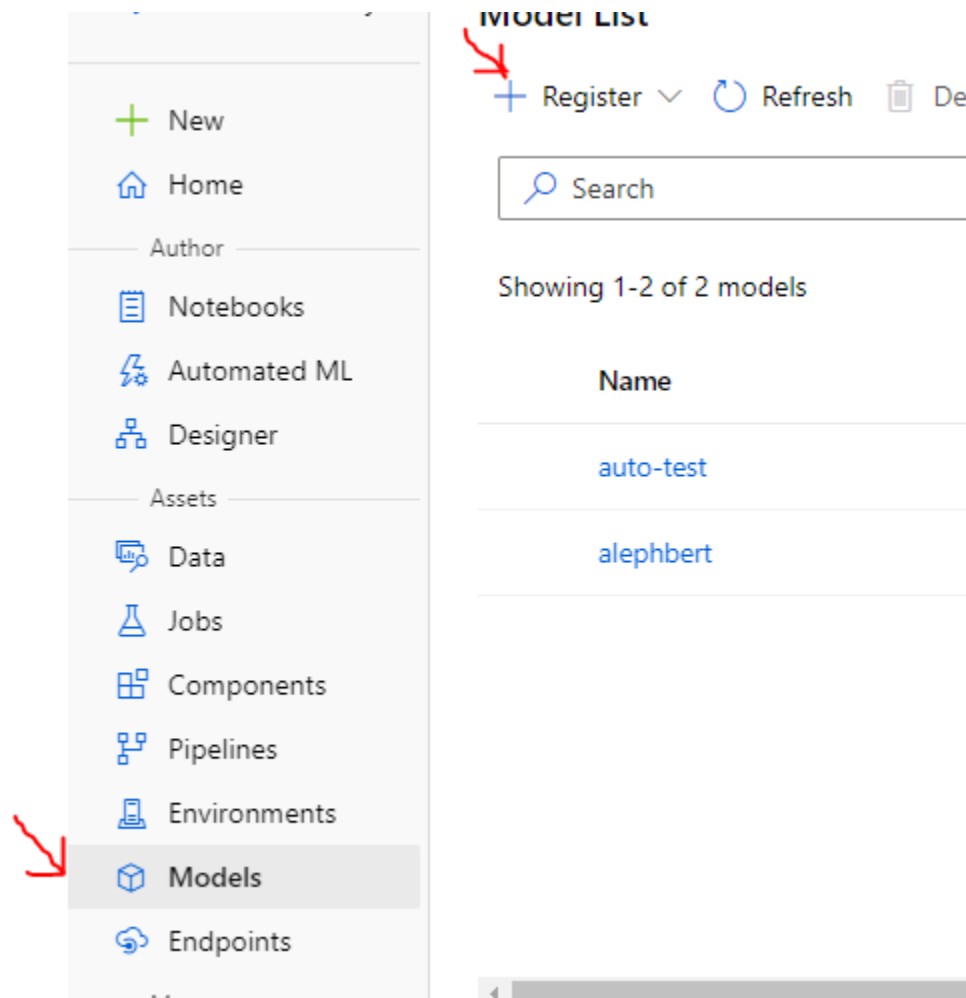
Number of nodes * ⓘ

3

Network configuration ⓘ

☒ Basic ☐ Advanced

Step 6: After that, click on the “Model” and then “Register” to register the trained model.




Step 7: Click the option directed by the red arrow.

The screenshot shows the 'Default Directory' sidebar on the left with navigation options: Default Directory, New, Home, Author, Notebooks, Automated ML, Designer, Assets, Data, Jobs, and Components. The main content area is titled 'Model List' and shows a breadcrumb path: Default Directory > RobohonTextAlgorithm > Models. Above the model list are buttons for Register, Refresh, Delete, Archive, and Deploy. The 'Register' dropdown menu is open, showing four options: 'From local files', 'From a job output', 'From datastore', and 'From local files (based on framework)'. A red arrow points to the last option. Below the dropdown is a table with columns for model name, Version, and Experiment.


	Version	Experiment
auto-test	1	
alephbert	1	

Step 8: Fill the respective fields as directed

Name * 

unique-name-for-model

Description

optional description 

Model framework *


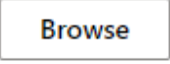
TensorFlow

Framework version *

2.9.2


Model file or folder *

☐ Upload file ☒ Upload folder

File or folder is required

Tags

Name	:	Value	Add
 No tags			

Step 9: When the model is registered successfully. This will take some time (depends on the model size)


Model List

+ Register ▾ Refresh Delete Archive Deploy ▾ Compare (preview) ▾ Edit columns Reset view | ...

Search

Created on ▾ Created by ▾ Tags ▾ All filter

Showing 1-1 of 1 models

Name	☆	Version	Experiment	Job (Run ID)	Created on ↓
 auto-test		1			Nov 1, 2022 4:28 PM

Step 10: Click on Deploy and then Deploy to Web Services as directed by the arrow.

auto-test:1 ☆

Details Versions Artifacts Endpoints Jobs Data Responsible AI Explanations (preview) Fair

Refresh Archive Deploy ▾ Download all Share model

Attributes

Name
auto-test

Version
1

Created on
Nov 1, 2022 4:28 PM

Created by
saifullahnust3711

Type
CUSTOM

Created by job

Deploy to real-time endpoint
Deploy the model using the new real-time endpoint wizard

Deploy to batch endpoint
Deploy the model using the new batch endpoint wizard


Deploy to web service
Deploy to a web service (only for models based on frameworks)

Tags
No tags

Properties
No properties

Description
testing auto models

Step 11: Fill the respective fields for the model deployment for real-time inference.

Name * ⓘ 

name-of-deployment

Description ⓘ

any descriptioon 1

Compute type * ⓘ

Azure Kubernetes Service

Compute name * ⓘ

Select or search by name

auto-compute

Models: auto-test:1

Step 12: Here insert the model scoring script script.py and conda_dependencies.yml. Both of these are available on the Git Hub.

Use custom deployment assets

☒ Use custom deployment assets

Entry script file * ⓘ

Select entry script file *


Conda dependencies file * ⓘ

Select conda dependencies file *

Dependencies

> Advanced

Step 13: Click on the Deploy button. Wait till the model is deployed.

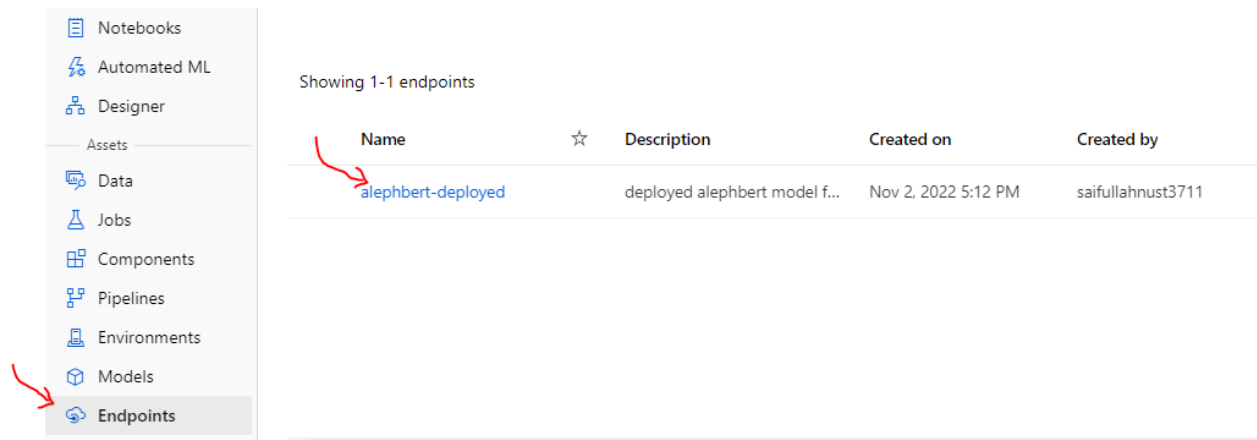
Conda dependencies file * 

*

Dependencies

[> Advanced](#)

Step 14: Check the endpoints for the deployed model endpoints. Click on the deployed model.



The screenshot displays the Azure ML interface. On the left sidebar, the 'Endpoints' menu item is highlighted with a red arrow. The main area shows a table of endpoints with the following data:

Name	Description	Created on	Created by
alephbert-deployed	deployed alephbert model f...	Nov 2, 2022 5:12 PM	saifullahnust3711

A red arrow points to the 'alephbert-deployed' endpoint in the table.

Step 15: URL is here which can be used for real-time inferencing.

Details

Test

Consume

Deployment logs

Model ID

auto-test:1

Created on

Nov 2, 2022 5:12 PM

Last updated on

Nov 2, 2022 5:12 PM

Compute target

auto-compute

Image ID

--

REST endpoint

http://20.62.153.81:80/api/v1/service/alephbert-deployed/score

Key-based authentication enabled

false

Token-based authentication enabled

false

Swagger URI