Import dataset to Vertex AI

I have a tar file in my gcs. It is image dataset with this folder structure having train, eval and test 3 folders. How can I import it as dataset in vertex ai dataset. It has approximately 13 lac files containing of img and their labelling xml.

To import your image dataset (with **`train`**, **`eval`**, and **`test`** folders containing **`.jpg`** images and **`.xml`** Pascal VOC annotations) into **Vertex AI Dataset**, you need to follow these steps:

✅ 1. Prepare Your Dataset for Vertex AI

Since your dataset uses **Pascal VOC format**, and you want to use **Vertex AI Custom Object Detection**, convert it to **COCO JSON** or **Vertex-compatible CSV format**. Google's AutoML requires a specific format:

Option A: Convert Pascal VOC to CSV (Vertex AI Format)

CSV Format for object detection in Vertex AI:

```
csv


SET,IMAGE_URL,LABEL,XMIN,YMIN,XMAX,YMAX
TRAIN,gs://your-bucket/images/train1.jpg,Car,0.1,0.2,0.5,0.6
```

Normalize bounding box values between 0 and 1 (relative to width and height).

You can use a Python script like:

```
python


import os
import xml.etree.ElementTree as ET
import pandas as pd

def parse_voc_annotation(xml_path, image_gcs_prefix, set_type):
    tree = ET.parse(xml_path)
    root = tree.getroot()
    filename = root.find("filename").text
    width = int(root.find("size/width").text)
    height = int(root.find("size/height").text)
    rows = []

    for obj in root.findall("object"):
        name = obj.find("name").text
        bbox = obj.find("bndbox")
        xmin = int(bbox.find("xmin").text) / width
        ymin = int(bbox.find("ymin").text) / height
        xmax = int(bbox.find("xmax").text) / width
        ymax = int(bbox.find("ymax").text) / height

        image_uri = os.path.join(image_gcs_prefix, filename)
        rows.append([set_type, image_uri, name, xmin, ymin, xmax, ymax])
    return rows

# Example usage:
```

```
all_rows = []
xml_dir = "/path/to/train/xmls"
image_gcs_prefix = "gs://your-bucket/dataset/train"

for xml_file in os.listdir(xml_dir):
    if xml_file.endswith(".xml"):
        all_rows.extend(parse_voc_annotation(os.path.join(xml_dir, xml_file), image_gcs_prefix,
"TRAIN"))

df = pd.DataFrame(all_rows, columns=["SET", "IMAGE_URL", "LABEL", "XMIN", "YMIN", "XMAX",
"YMAX"])
df.to_csv("vertex_dataset.csv", index=False)
```

✅ 2. Upload Your Images and CSV to GCS

Make sure your tar is extracted and all image files are in a **publicly accessible GCS path** or authenticated access is configured:

```
bash


gsutil cp -r train/ gs://your-bucket/dataset/train/
gsutil cp vertex_dataset.csv gs://your-bucket/dataset/
```

✅ 3. Import Dataset in Vertex AI

1. Open **Vertex AI** in Google Cloud Console.
2. Go to **Datasets > Create**.
3. Choose **`Image Object Detection`**.
4. Name the dataset.
5. Choose **Import Data > From CSV**.
6. Paste your **`gs://.../vertex_dataset.csv`** path.

✅ 4. (Optional) Handling Test Set

Since test images have **no annotations**, you don't need to import them for training. But you can still upload them to GCS for manual evaluation later.

✅ Summary Table

Task	Tool/Command
Convert VOC XML to CSV	Python (custom script)
Upload data to GCS	**`gsutil cp -r train/ gs://bucket/...`**
Format accepted by Vertex AI	CSV with normalized coordinates
Import into Vertex AI	Console UI → Datasets → Create → Image Object Detection → Import from CSV

Would you like me to generate the VOC-to-CSV converter as a complete runnable Python script for Colab or local execution?