Part 3 - Clustering using both image and text

In part 1 and part 2, we have practived the art of clustering text and images separately. However, can we map image and text to the same space? In the Pokemon world, Pokedex catalogs Pokemon's appearances and various metadata. We will build our Pokedex from image dataset link and meta metadata link. Fortunately, ECE 219 Gym kindly provides new Pokemon trainers with the helper code for data preprocessing and inferencing. Please find the code on Bruinlearn modules Week 4.

We will use the pre-trained CLIP [8] to illustrate the idea of multimodal clustering. CLIP (Contrastive Language–Image Pretraining) is an innovative model developed by OpenAI, de- signed to understand and connect concepts from both text and images. CLIP is trained on a vast array of internet-sourced text-image pairs. This extensive training enables the model to understand a broad spectrum of visual concepts and their textual descriptions.

CLIP consists of two primary components: a text encoder and an image encoder. The text encoder processes textual data, converting sentences and phrases into numerical representa- tions. Simultaneously, the image encoder transforms visual inputs into a corresponding set of numerical values. These encoders are trained to map both text and images into a shared embedding space, allowing the model to compare and relate the two different types of data di- rectly. The training employs a contrastive learning approach, where the model learns to match corresponding text and image pairs against numerous non-matching pairs. This approach helps the model in accurately associating images with their relevant textual descriptions and vice versa.

In [ ]:
```
!pip install datasets transformers numpy pandas Pillow matplotlib
!pip install torch tqdm scipy
!pip install git+https://github.com/openai/CLIP.git
!pip install plotly umap-learn
```

```
Collecting datasets
  Downloading datasets-3.2.0-py3-none-any.whl.metadata (20 kB)
Requirement already satisfied: transformers in /usr/local/lib/python3.11/dis
t-packages (4.48.2)
Requirement already satisfied: numpy in /usr/local/lib/python3.11/dist-packa
ges (1.26.4)
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-pack
ages (2.2.2)
Requirement already satisfied: Pillow in /usr/local/lib/python3.11/dist-pack
ages (11.1.0)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.11/dist-
packages (3.10.0)
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-pa
ckages (from datasets) (3.17.0)
Requirement already satisfied: pyarrow>=15.0.0 in /usr/local/lib/python3.11/
dist-packages (from datasets) (17.0.0)
Collecting dill<0.3.9,>=0.3.0 (from datasets)
  Downloading dill-0.3.8-py3-none-any.whl.metadata (10 kB)
Requirement already satisfied: requests>=2.32.2 in /usr/local/lib/python3.1
1/dist-packages (from datasets) (2.32.3)
Requirement already satisfied: tqdm>=4.66.3 in /usr/local/lib/python3.11/dis
t-packages (from datasets) (4.67.1)
Collecting xxhash (from datasets)
  Downloading xxhash-3.5.0-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x
86_64.whl.metadata (12 kB)
Collecting multiprocess<0.70.17 (from datasets)
  Downloading multiprocess-0.70.16-py311-none-any.whl.metadata (7.2 kB)
Collecting fsspec<=2024.9.0,>=2023.1.0 (from fsspec[http]<=2024.9.0,>=2023.
1.0->datasets)
  Downloading fsspec-2024.9.0-py3-none-any.whl.metadata (11 kB)
Requirement already satisfied: aiohttp in /usr/local/lib/python3.11/dist-pac
kages (from datasets) (3.11.11)
Requirement already satisfied: huggingface-hub>=0.23.0 in /usr/local/lib/pyt
hon3.11/dist-packages (from datasets) (0.28.1)
Requirement already satisfied: packaging in /usr/local/lib/python3.11/dist-p
ackages (from datasets) (24.2)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.11/dist
-packages (from datasets) (6.0.2)
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.1
1/dist-packages (from transformers) (2024.11.6)
Requirement already satisfied: tokenizers<0.22,>=0.21 in /usr/local/lib/pyth
on3.11/dist-packages (from transformers) (0.21.0)
Requirement already satisfied: safetensors>=0.4.1 in /usr/local/lib/python3.
11/dist-packages (from transformers) (0.5.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/pyth
on3.11/dist-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dis
t-packages (from pandas) (2025.1)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/d
ist-packages (from pandas) (2025.1)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.1
1/dist-packages (from matplotlib) (1.3.1)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.11/dis
t-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.1
1/dist-packages (from matplotlib) (4.55.8)
```

```
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/python3.1
1/dist-packages (from matplotlib) (1.4.8)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.1
1/dist-packages (from matplotlib) (3.2.1)
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in /usr/local/lib/pyt
hon3.11/dist-packages (from aiohttp->datasets) (2.4.4)
Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.1
1/dist-packages (from aiohttp->datasets) (1.3.2)
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.11/di
st-packages (from aiohttp->datasets) (25.1.0)
Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.1
1/dist-packages (from aiohttp->datasets) (1.5.0)
Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python
3.11/dist-packages (from aiohttp->datasets) (6.1.0)
Requirement already satisfied: propcache>=0.2.0 in /usr/local/lib/python3.1
1/dist-packages (from aiohttp->datasets) (0.2.1)
Requirement already satisfied: yarl<2.0,>=1.17.0 in /usr/local/lib/python3.1
1/dist-packages (from aiohttp->datasets) (1.18.3)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/
python3.11/dist-packages (from huggingface-hub>=0.23.0->datasets) (4.12.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-pa
ckages (from python-dateutil>=2.8.2->pandas) (1.17.0)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/py
thon3.11/dist-packages (from requests>=2.32.2->datasets) (3.4.1)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dis
t-packages (from requests>=2.32.2->datasets) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.
11/dist-packages (from requests>=2.32.2->datasets) (2.3.0)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.
11/dist-packages (from requests>=2.32.2->datasets) (2025.1.31)
Downloading datasets-3.2.0-py3-none-any.whl (480 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 480.6/480.6 kB 7.8 MB/s eta 0:0
0:00
Downloading dill-0.3.8-py3-none-any.whl (116 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 116.3/116.3 kB 9.7 MB/s eta 0:0
0:00
Downloading fsspec-2024.9.0-py3-none-any.whl (179 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 179.3/179.3 kB 13.7 MB/s eta 0:0
0:00
Downloading multiprocess-0.70.16-py311-none-any.whl (143 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 143.5/143.5 kB 5.2 MB/s eta 0:0
0:00
Downloading xxhash-3.5.0-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86
_64.whl (194 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 194.8/194.8 kB 7.3 MB/s eta 0:0
0:00
Installing collected packages: xxhash, fsspec, dill, multiprocess, datasets
  Attempting uninstall: fsspec
    Found existing installation: fsspec 2024.10.0
    Uninstalling fsspec-2024.10.0:
      Successfully uninstalled fsspec-2024.10.0
ERROR: pip's dependency resolver does not currently take into account all th
e packages that are installed. This behaviour is the source of the following
dependency conflicts.
gcsfs 2024.10.0 requires fsspec==2024.10.0, but you have fsspec 2024.9.0 whi
ch is incompatible.
```

```
torch 2.5.1+cu124 requires nvidia-cublas-cu12==12.4.5.8; platform_system ==
"Linux" and platform_machine == "x86_64", but you have nvidia-cublas-cu12 1
2.5.3.2 which is incompatible.
torch 2.5.1+cu124 requires nvidia-cuda-cupti-cu12==12.4.127; platform_system
== "Linux" and platform_machine == "x86_64", but you have nvidia-cuda-cupti-
cu12 12.5.82 which is incompatible.
torch 2.5.1+cu124 requires nvidia-cuda-nvrtc-cu12==12.4.127; platform_system
== "Linux" and platform_machine == "x86_64", but you have nvidia-cuda-nvrtc-
cu12 12.5.82 which is incompatible.
torch 2.5.1+cu124 requires nvidia-cuda-runtime-cu12==12.4.127; platform_syst
em == "Linux" and platform_machine == "x86_64", but you have nvidia-cuda-run
time-cu12 12.5.82 which is incompatible.
torch 2.5.1+cu124 requires nvidia-cudnn-cu12==9.1.0.70; platform_system ==
"Linux" and platform_machine == "x86_64", but you have nvidia-cudnn-cu12 9.
3.0.75 which is incompatible.
torch 2.5.1+cu124 requires nvidia-cufft-cu12==11.2.1.3; platform_system ==
"Linux" and platform_machine == "x86_64", but you have nvidia-cufft-cu12 11.
2.3.61 which is incompatible.
torch 2.5.1+cu124 requires nvidia-curand-cu12==10.3.5.147; platform_system =
= "Linux" and platform_machine == "x86_64", but you have nvidia-curand-cu12
10.3.6.82 which is incompatible.
torch 2.5.1+cu124 requires nvidia-cusolver-cu12==11.6.1.9; platform_system =
= "Linux" and platform_machine == "x86_64", but you have nvidia-cusolver-cu1
2 11.6.3.83 which is incompatible.
torch 2.5.1+cu124 requires nvidia-cusparse-cu12==12.3.1.170; platform_system
== "Linux" and platform_machine == "x86_64", but you have nvidia-cusparse-cu
12 12.5.1.3 which is incompatible.
torch 2.5.1+cu124 requires nvidia-nvjitlink-cu12==12.4.127; platform_system
== "Linux" and platform_machine == "x86_64", but you have nvidia-nvjitlink-c
u12 12.5.82 which is incompatible.
Successfully installed datasets-3.2.0 dill-0.3.8 fsspec-2024.9.0 multiproces
s-0.70.16 xxhash-3.5.0
Requirement already satisfied: torch in /usr/local/lib/python3.11/dist-packa
ges (2.5.1+cu124)
Requirement already satisfied: tqdm in /usr/local/lib/python3.11/dist-packag
es (4.67.1)
Requirement already satisfied: scipy in /usr/local/lib/python3.11/dist-packa
ges (1.13.1)
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-pa
ckages (from torch) (3.17.0)
Requirement already satisfied: typing-extensions>=4.8.0 in /usr/local/lib/py
thon3.11/dist-packages (from torch) (4.12.2)
Requirement already satisfied: networkx in /usr/local/lib/python3.11/dist-pa
ckages (from torch) (3.4.2)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.11/dist-pack
ages (from torch) (3.1.5)
Requirement already satisfied: fsspec in /usr/local/lib/python3.11/dist-pack
ages (from torch) (2024.9.0)
Collecting nvidia-cuda-nvrtc-cu12==12.4.127 (from torch)
  Downloading nvidia_cuda_nvrtc_cu12-12.4.127-py3-none-manylinux2014_x86_64.
whl.metadata (1.5 kB)
Collecting nvidia-cuda-runtime-cu12==12.4.127 (from torch)
  Downloading nvidia_cuda_runtime_cu12-12.4.127-py3-none-manylinux2014_x86_6
4.whl.metadata (1.5 kB)
Collecting nvidia-cuda-cupti-cu12==12.4.127 (from torch)
  Downloading nvidia_cuda_cupti_cu12-12.4.127-py3-none-manylinux2014_x86_64.
```

```
whl.metadata (1.6 kB)
Collecting nvidia-cudnn-cu12==9.1.0.70 (from torch)
  Downloading nvidia_cudnn_cu12-9.1.0.70-py3-none-manylinux2014_x86_64.whl.m
etadata (1.6 kB)
Collecting nvidia-cublas-cu12==12.4.5.8 (from torch)
  Downloading nvidia_cublas_cu12-12.4.5.8-py3-none-manylinux2014_x86_64.whl.
metadata (1.5 kB)
Collecting nvidia-cufft-cu12==11.2.1.3 (from torch)
  Downloading nvidia_cufft_cu12-11.2.1.3-py3-none-manylinux2014_x86_64.whl.m
etadata (1.5 kB)
Collecting nvidia-curand-cu12==10.3.5.147 (from torch)
  Downloading nvidia_curand_cu12-10.3.5.147-py3-none-manylinux2014_x86_64.wh
l.metadata (1.5 kB)
Collecting nvidia-cusolver-cu12==11.6.1.9 (from torch)
  Downloading nvidia_cusolver_cu12-11.6.1.9-py3-none-manylinux2014_x86_64.wh
l.metadata (1.6 kB)
Collecting nvidia-cusparse-cu12==12.3.1.170 (from torch)
  Downloading nvidia_cusparse_cu12-12.3.1.170-py3-none-manylinux2014_x86_64.
whl.metadata (1.6 kB)
Requirement already satisfied: nvidia-nccl-cu12==2.21.5 in /usr/local/lib/py
thon3.11/dist-packages (from torch) (2.21.5)
Requirement already satisfied: nvidia-nvtx-cu12==12.4.127 in /usr/local/lib/
python3.11/dist-packages (from torch) (12.4.127)
Collecting nvidia-nvjitlink-cu12==12.4.127 (from torch)
  Downloading nvidia_nvjitlink_cu12-12.4.127-py3-none-manylinux2014_x86_64.w
hl.metadata (1.5 kB)
Requirement already satisfied: triton==3.1.0 in /usr/local/lib/python3.11/di
st-packages (from torch) (3.1.0)
Requirement already satisfied: sympy==1.13.1 in /usr/local/lib/python3.11/di
st-packages (from torch) (1.13.1)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.
11/dist-packages (from sympy==1.13.1->torch) (1.3.0)
Requirement already satisfied: numpy<2.3,>=1.22.4 in /usr/local/lib/python3.
11/dist-packages (from scipy) (1.26.4)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.11/
dist-packages (from jinja2->torch) (3.0.2)
Downloading nvidia_cublas_cu12-12.4.5.8-py3-none-manylinux2014_x86_64.whl (3
63.4 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 363.4/363.4 MB 4.3 MB/s eta 0:0
0:00
Downloading nvidia_cuda_cupti_cu12-12.4.127-py3-none-manylinux2014_x86_64.wh
l (13.8 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 13.8/13.8 MB 92.4 MB/s eta 0:00:
00
Downloading nvidia_cuda_nvrtc_cu12-12.4.127-py3-none-manylinux2014_x86_64.wh
l (24.6 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 24.6/24.6 MB 67.7 MB/s eta 0:00:
00
Downloading nvidia_cuda_runtime_cu12-12.4.127-py3-none-manylinux2014_x86_64.
whl (883 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 883.7/883.7 kB 51.8 MB/s eta 0:0
0:00
Downloading nvidia_cudnn_cu12-9.1.0.70-py3-none-manylinux2014_x86_64.whl (66
4.8 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 664.8/664.8 MB 2.2 MB/s eta 0:0
0:00
```

```
Downloading nvidia_cufft_cu12-11.2.1.3-py3-none-manylinux2014_x86_64.whl (21
1.5 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 211.5/211.5 MB 6.0 MB/s eta 0:0
0:00
Downloading nvidia_curand_cu12-10.3.5.147-py3-none-manylinux2014_x86_64.whl
(56.3 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 56.3/56.3 MB 11.3 MB/s eta 0:00:
00
Downloading nvidia_cusolver_cu12-11.6.1.9-py3-none-manylinux2014_x86_64.whl
(127.9 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 127.9/127.9 MB 7.9 MB/s eta 0:0
0:00
Downloading nvidia_cusparse_cu12-12.3.1.170-py3-none-manylinux2014_x86_64.wh
l (207.5 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 207.5/207.5 MB 5.6 MB/s eta 0:0
0:00
Downloading nvidia_nvjitlink_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl
(21.1 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 21.1/21.1 MB 59.0 MB/s eta 0:00:
00
Installing collected packages: nvidia-nvjitlink-cu12, nvidia-curand-cu12, nv
idia-cufft-cu12, nvidia-cuda-runtime-cu12, nvidia-cuda-nvrtc-cu12, nvidia-cu
da-cupti-cu12, nvidia-cublas-cu12, nvidia-cusparse-cu12, nvidia-cudnn-cu12,
nvidia-cusolver-cu12
  Attempting uninstall: nvidia-nvjitlink-cu12
    Found existing installation: nvidia-nvjitlink-cu12 12.5.82
    Uninstalling nvidia-nvjitlink-cu12-12.5.82:
      Successfully uninstalled nvidia-nvjitlink-cu12-12.5.82
  Attempting uninstall: nvidia-curand-cu12
    Found existing installation: nvidia-curand-cu12 10.3.6.82
    Uninstalling nvidia-curand-cu12-10.3.6.82:
      Successfully uninstalled nvidia-curand-cu12-10.3.6.82
  Attempting uninstall: nvidia-cufft-cu12
    Found existing installation: nvidia-cufft-cu12 11.2.3.61
    Uninstalling nvidia-cufft-cu12-11.2.3.61:
      Successfully uninstalled nvidia-cufft-cu12-11.2.3.61
  Attempting uninstall: nvidia-cuda-runtime-cu12
    Found existing installation: nvidia-cuda-runtime-cu12 12.5.82
    Uninstalling nvidia-cuda-runtime-cu12-12.5.82:
      Successfully uninstalled nvidia-cuda-runtime-cu12-12.5.82
  Attempting uninstall: nvidia-cuda-nvrtc-cu12
    Found existing installation: nvidia-cuda-nvrtc-cu12 12.5.82
    Uninstalling nvidia-cuda-nvrtc-cu12-12.5.82:
      Successfully uninstalled nvidia-cuda-nvrtc-cu12-12.5.82
  Attempting uninstall: nvidia-cuda-cupti-cu12
    Found existing installation: nvidia-cuda-cupti-cu12 12.5.82
    Uninstalling nvidia-cuda-cupti-cu12-12.5.82:
      Successfully uninstalled nvidia-cuda-cupti-cu12-12.5.82
  Attempting uninstall: nvidia-cublas-cu12
    Found existing installation: nvidia-cublas-cu12 12.5.3.2
    Uninstalling nvidia-cublas-cu12-12.5.3.2:
      Successfully uninstalled nvidia-cublas-cu12-12.5.3.2
  Attempting uninstall: nvidia-cusparse-cu12
    Found existing installation: nvidia-cusparse-cu12 12.5.1.3
    Uninstalling nvidia-cusparse-cu12-12.5.1.3:
      Successfully uninstalled nvidia-cusparse-cu12-12.5.1.3
```

```
  Attempting uninstall: nvidia-cudnn-cu12
    Found existing installation: nvidia-cudnn-cu12 9.3.0.75
    Uninstalling nvidia-cudnn-cu12-9.3.0.75:
      Successfully uninstalled nvidia-cudnn-cu12-9.3.0.75
  Attempting uninstall: nvidia-cusolver-cu12
    Found existing installation: nvidia-cusolver-cu12 11.6.3.83
    Uninstalling nvidia-cusolver-cu12-11.6.3.83:
      Successfully uninstalled nvidia-cusolver-cu12-11.6.3.83
Successfully installed nvidia-cublas-cu12-12.4.5.8 nvidia-cuda-cupti-cu12-1
2.4.127 nvidia-cuda-nvrtc-cu12-12.4.127 nvidia-cuda-runtime-cu12-12.4.127 nv
idia-cudnn-cu12-9.1.0.70 nvidia-cufft-cu12-11.2.1.3 nvidia-curand-cu12-10.3.
5.147 nvidia-cusolver-cu12-11.6.1.9 nvidia-cusparse-cu12-12.3.1.170 nvidia-n
vjitlink-cu12-12.4.127
Collecting git+https://github.com/openai/CLIP.git
  Cloning https://github.com/openai/CLIP.git to /tmp/pip-req-build-b1l2ltc_
  Running command git clone --filter=blob:none --quiet https://github.com/op
enai/CLIP.git /tmp/pip-req-build-b1l2ltc_
  Resolved https://github.com/openai/CLIP.git to commit dcba3cb2e2827b402d27
01e7e1c7d9fed8a20ef1
  Preparing metadata (setup.py) ... done
Collecting ftfy (from clip==1.0)
  Downloading ftfy-6.3.1-py3-none-any.whl.metadata (7.3 kB)
Requirement already satisfied: packaging in /usr/local/lib/python3.11/dist-p
ackages (from clip==1.0) (24.2)
Requirement already satisfied: regex in /usr/local/lib/python3.11/dist-packa
ges (from clip==1.0) (2024.11.6)
Requirement already satisfied: tqdm in /usr/local/lib/python3.11/dist-packag
es (from clip==1.0) (4.67.1)
Requirement already satisfied: torch in /usr/local/lib/python3.11/dist-packa
ges (from clip==1.0) (2.5.1+cu124)
Requirement already satisfied: torchvision in /usr/local/lib/python3.11/dist
-packages (from clip==1.0) (0.20.1+cu124)
Requirement already satisfied: wcwidth in /usr/local/lib/python3.11/dist-pac
kages (from ftfy->clip==1.0) (0.2.13)
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-pa
ckages (from torch->clip==1.0) (3.17.0)
Requirement already satisfied: typing-extensions>=4.8.0 in /usr/local/lib/py
thon3.11/dist-packages (from torch->clip==1.0) (4.12.2)
Requirement already satisfied: networkx in /usr/local/lib/python3.11/dist-pa
ckages (from torch->clip==1.0) (3.4.2)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.11/dist-pack
ages (from torch->clip==1.0) (3.1.5)
Requirement already satisfied: fsspec in /usr/local/lib/python3.11/dist-pack
ages (from torch->clip==1.0) (2024.9.0)
Requirement already satisfied: nvidia-cuda-nvrtc-cu12==12.4.127 in /usr/loca
l/lib/python3.11/dist-packages (from torch->clip==1.0) (12.4.127)
Requirement already satisfied: nvidia-cuda-runtime-cu12==12.4.127 in /usr/lo
cal/lib/python3.11/dist-packages (from torch->clip==1.0) (12.4.127)
Requirement already satisfied: nvidia-cuda-cupti-cu12==12.4.127 in /usr/loca
l/lib/python3.11/dist-packages (from torch->clip==1.0) (12.4.127)
Requirement already satisfied: nvidia-cudnn-cu12==9.1.0.70 in /usr/local/li
b/python3.11/dist-packages (from torch->clip==1.0) (9.1.0.70)
Requirement already satisfied: nvidia-cublas-cu12==12.4.5.8 in /usr/local/li
b/python3.11/dist-packages (from torch->clip==1.0) (12.4.5.8)
Requirement already satisfied: nvidia-cufft-cu12==11.2.1.3 in /usr/local/li
b/python3.11/dist-packages (from torch->clip==1.0) (11.2.1.3)
```

```
Requirement already satisfied: nvidia-curand-cu12==10.3.5.147 in /usr/local/
lib/python3.11/dist-packages (from torch->clip==1.0) (10.3.5.147)
Requirement already satisfied: nvidia-cusolver-cu12==11.6.1.9 in /usr/local/
lib/python3.11/dist-packages (from torch->clip==1.0) (11.6.1.9)
Requirement already satisfied: nvidia-cusparse-cu12==12.3.1.170 in /usr/loca
l/lib/python3.11/dist-packages (from torch->clip==1.0) (12.3.1.170)
Requirement already satisfied: nvidia-nccl-cu12==2.21.5 in /usr/local/lib/py
thon3.11/dist-packages (from torch->clip==1.0) (2.21.5)
Requirement already satisfied: nvidia-nvtx-cu12==12.4.127 in /usr/local/lib/
python3.11/dist-packages (from torch->clip==1.0) (12.4.127)
Requirement already satisfied: nvidia-nvjitlink-cu12==12.4.127 in /usr/loca
l/lib/python3.11/dist-packages (from torch->clip==1.0) (12.4.127)
Requirement already satisfied: triton==3.1.0 in /usr/local/lib/python3.11/di
st-packages (from torch->clip==1.0) (3.1.0)
Requirement already satisfied: sympy==1.13.1 in /usr/local/lib/python3.11/di
st-packages (from torch->clip==1.0) (1.13.1)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.
11/dist-packages (from sympy==1.13.1->torch->clip==1.0) (1.3.0)
Requirement already satisfied: numpy in /usr/local/lib/python3.11/dist-packa
ges (from torchvision->clip==1.0) (1.26.4)
Requirement already satisfied: pillow!=8.3.*,>=5.3.0 in /usr/local/lib/pytho
n3.11/dist-packages (from torchvision->clip==1.0) (11.1.0)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.11/
dist-packages (from jinja2->torch->clip==1.0) (3.0.2)
Downloading ftfy-6.3.1-py3-none-any.whl (44 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 44.8/44.8 kB 2.9 MB/s eta 0:00:0
0
Building wheels for collected packages: clip
  Building wheel for clip (setup.py) ... done
  Created wheel for clip: filename=clip-1.0-py3-none-any.whl size=1369489 sh
a256=d1468ef5816fc7f208b62055e2778f0f05c84faa93889c615c5f0decca3faedc
  Stored in directory: /tmp/pip-ephem-wheel-cache-zj7e6hlh/wheels/3f/7c/a4/9
b490845988bf7a4db33674d52f709f088f64392063872eb9a
Successfully built clip
Installing collected packages: ftfy, clip
Successfully installed clip-1.0 ftfy-6.3.1
Requirement already satisfied: plotly in /usr/local/lib/python3.11/dist-pack
ages (5.24.1)
Collecting umap-learn
  Downloading umap_learn-0.5.7-py3-none-any.whl.metadata (21 kB)
Requirement already satisfied: tenacity>=6.2.0 in /usr/local/lib/python3.11/
dist-packages (from plotly) (9.0.0)
Requirement already satisfied: packaging in /usr/local/lib/python3.11/dist-p
ackages (from plotly) (24.2)
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.11/dist
-packages (from umap-learn) (1.26.4)
Requirement already satisfied: scipy>=1.3.1 in /usr/local/lib/python3.11/dis
t-packages (from umap-learn) (1.13.1)
Requirement already satisfied: scikit-learn>=0.22 in /usr/local/lib/python3.
11/dist-packages (from umap-learn) (1.6.1)
Requirement already satisfied: numba>=0.51.2 in /usr/local/lib/python3.11/di
st-packages (from umap-learn) (0.61.0)
Collecting pynndescent>=0.5 (from umap-learn)
  Downloading pynndescent-0.5.13-py3-none-any.whl.metadata (6.8 kB)
Requirement already satisfied: tqdm in /usr/local/lib/python3.11/dist-packag
es (from umap-learn) (4.67.1)
```

```
Requirement already satisfied: llvmlite<0.45,>=0.44.0dev0 in /usr/local/lib/
python3.11/dist-packages (from numba>=0.51.2->umap-learn) (0.44.0)
Requirement already satisfied: joblib>=0.11 in /usr/local/lib/python3.11/dis
t-packages (from pynndescent>=0.5->umap-learn) (1.4.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in /usr/local/lib/python
3.11/dist-packages (from scikit-learn>=0.22->umap-learn) (3.5.0)
Downloading umap_learn-0.5.7-py3-none-any.whl (88 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 88.8/88.8 kB 5.4 MB/s eta 0:00:0
0
Downloading pynndescent-0.5.13-py3-none-any.whl (56 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 56.9/56.9 kB 5.1 MB/s eta 0:00:0
0
Installing collected packages: pynndescent, umap-learn
Successfully installed pynndescent-0.5.13 umap-learn-0.5.7
```

In [ ]:
```
### QUESTION 28:
In the first and second question, we investigated how CLIP creates 'clusters
mapping images and texts of various Pokemon into a high-dimensional space ar
hood of these items in this space. For this question, please use t-SNE to vi
specifically for Pokemon types Bug, Fire, and Grass. You can use scatter plo
plotly. For the visualization, color-code each point based on its first type
argument, and label each point with the Pokemon's name and types using 'hove
will enable you to identify each Pokemon represented in your visualization.
visualization, analyze it and discuss whether the clustering of Pokemon type
```

```
Mounted at /content/drive
```

In [ ]:
```
# Code for Q28
```

In [ ]:
```
After completing the visualization, analyze it and discuss whether the clust
```

Answer:

In [ ]:
```
# Code for Q26
# Load the Pokedex data
pokedex = construct_pokedex()
```

In [ ]:
```
# Define type queries
type_queries = {
    "Bug": "A Pokémon of Bug type",
    "Fire": "A Pokémon of Fire type",
    "Grass": "A Pokémon of Grass type",
    "Dark": "A Pokémon of Dark type",
    "Dragon": "A Pokémon of Dragon type"
}
```

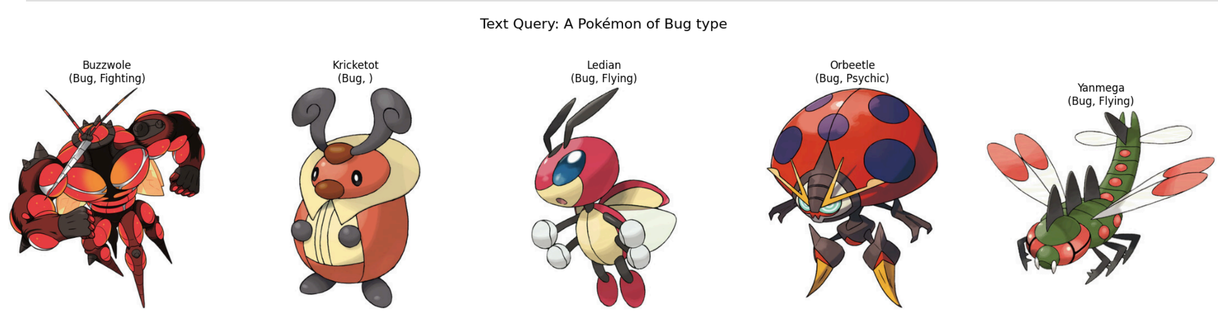
In [ ]:
```
import numpy as np

# Load CLIP model
model, preprocess, device = load_clip_model()

# Encode text queries
text_embeddings = clip_inference_text(model, preprocess, list(type_queries.v
```

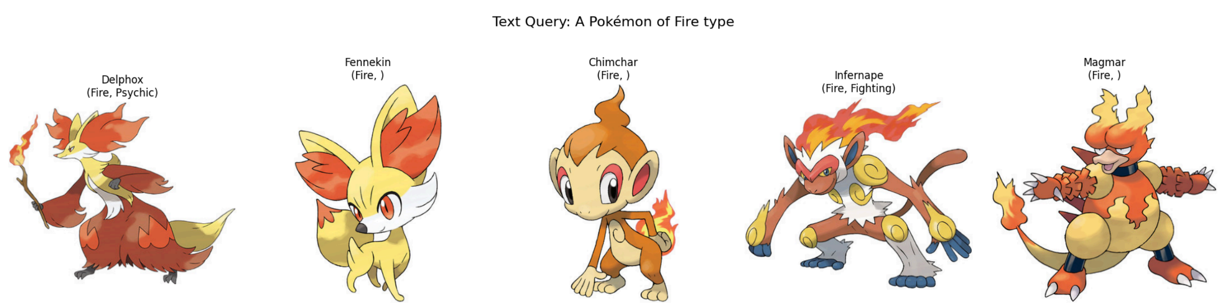
```
# Encode Pokémon images
image_embeddings = clip_inference_image(model, preprocess, pokedex["image_pa

# Compute similarity between text queries and images
similarities = compute_similarity_text_to_image(image_embeddings, text_embec

# Get the top 5 most relevant Pokémon for each type
top_k_indices = np.argsort(similarities, axis=0)[-5:][::-1]
```

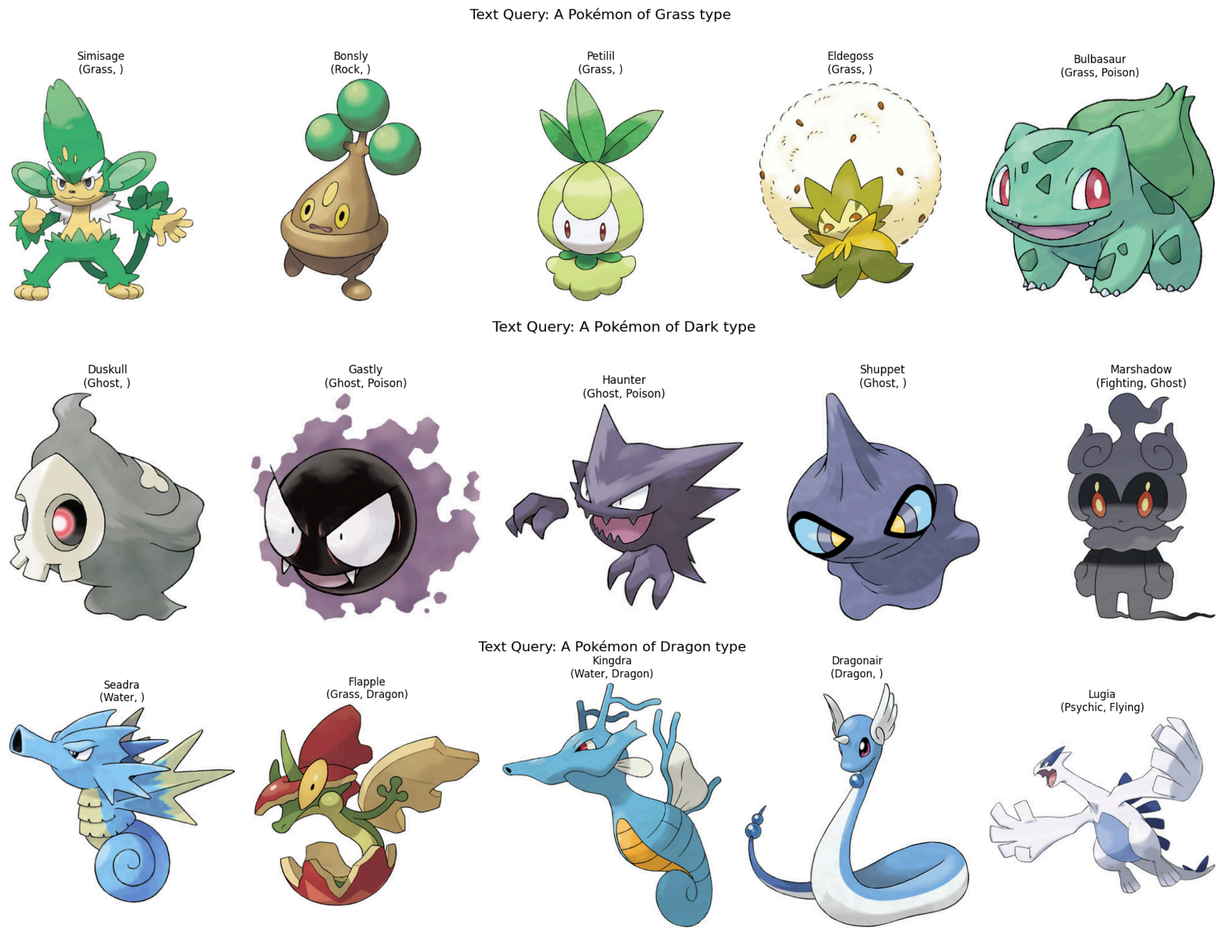
```
100%|██████████████████████████████████████| 890M/890M [00:21<00:00, 43.6Mi
B/s]
100%|██████████| 754/754 [05:25<00:00,  2.31it/s]
```

In [ ]:
```
# Function to plot top Pokémon results
def plot_top_pokemon(top_indices, query, type_name):
    fig, axes = plt.subplots(1, 5, figsize=(20, 5))
    fig.suptitle(f"Text Query: {query}", fontsize=16)

    for i, idx in enumerate(top_indices):
        pokemon_name = pokedex.iloc[idx]["Name"]
        pokemon_type1 = pokedex.iloc[idx]["Type1"]
        pokemon_type2 = pokedex.iloc[idx]["Type2"]
        img_path = pokedex.iloc[idx]["image_path"]

        # Load and display the Pokémon image
        image = Image.open(img_path)
        axes[i].imshow(image)
        axes[i].axis("off")
        axes[i].set_title(f"{pokemon_name}\n({pokemon_type1}, {pokemon_type2

    plt.tight_layout(pad=2.0)
    plt.show()
```

In [ ]:
```
# Loop through all types and plot results
for type_name, query in type_queries.items():
    top_indices = top_k_indices[:, list(type_queries.keys()).index(type_name
    plot_top_pokemon(top_indices, query, type_name)
```







Answer

The results of the queries can be seen above. Judging by these results, the queries were actually very effective. In almost every single case the query resulted in pokemen from the rolodex that perfectly matched the query type. There are a few off cases where the pokemon is not of the correct type. For example, pokemon of bug, and fire were 100% accurate. This is most likwly because it prioritized the respective bright green and red colors which are. abit easier to isolate within images. One case where it didnt perform as well is for dark type. The queue seems to be . mit more ambiguous and the results reflect this. The pokemon are . bit darker in hugh but not always the correct type. This shows how ambiguity in queue and uniqueness can result in worst accuracy and results overall.

QUESTION 27:

Randomly select 10 Pokemon images from the dataset and use CLIP to find the most relevant types (use your preferred template, e.g "type: Bug"). For each selected Pokemon, please plot it and indicate:

- its name and first and second type;
- the five most relevant types predicted by CLIP and their predicted probabilities.

In [ ]:
```
# Code for Q27
# Extract unique types
unique_types = pd.unique(pd.concat([pokedex['Type1'], pokedex['Type2']])).to

# Construct query dictionary
type_queries = {t: f"type: {t}" for t in unique_types if pd.notna(t)}

# Load CLIP model
model, preprocess, device = load_clip_model()

# Encode type queries using CLIP
query_embeddings = clip_inference_text(model, preprocess, list(type_queries.

# Select 10 random Pokémon (random seed = 108)
selected_pokemon = pokedex.sample(n=10, random_state=108).reset_index(drop=T
```

In [ ]:
```
# Function to predict top 5 Pokémon types using CLIP
def predict_types(image_paths):
    # Encode Pokémon images using CLIP
    image_embeddings = clip_inference_image(model, preprocess, image_paths,

    # Compute similarity scores (text-to-image)
    similarity_scores = compute_similarity_text_to_image(image_embeddings, q

    # Get top 5 predicted types for each image
    predicted_types = []
    for scores in similarity_scores:
        top_indices = np.argsort(scores)[::-1][:5]
        relevant_types = [(list(type_queries.keys())[i], scores[i]) for i in
        predicted_types.append(relevant_types)

    return predicted_types

# Function to plot Pokémon with actual and predicted types
def plot_pokemon_with_predictions(pokemon_name, type1, type2, image_path, pr
    plt.figure(figsize=(6, 6))

    # Load and display
    image = Image.open(image_path)
    plt.imshow(image)
    plt.axis("off")

    # Title
    plt.title(f"{pokemon_name}\n Type 1: {type1}, Type 2: {type2}", fontsize

    # Display predicted types with probabilities
    caption = "\n".join([f"{ptype}: {prob:.4f}" for ptype, prob in predicted
    plt.figtext(0.5, 0.01, caption, wrap=True, horizontalalignment='center',

    plt.show()
```

In [ ]:
```
# Process each Pokémon and display results
for _, row in selected_pokemon.iterrows():
    pokemon_name = row['Name']
    type1 = row['Type1']
```

```
    type2 = row['Type2'] if pd.notna(row['Type2']) else "None"
    image_path = row['image_path']

    predicted_types = predict_types([image_path])[0]  # Get predictions for 
    plot_pokemon_with_predictions(pokemon_name, type1, type2, image_path, pr
```

```
100%|██████████| 1/1 [00:00<00:00, 32.84it/s]
```


```
100%|██████████| 1/1 [00:00<00:00, 27.22it/s]
```


```
100%|██████████| 1/1 [00:00<00:00, 26.55it/s]
```


```
100%|██████████| 1/1 [00:00<00:00, 28.09it/s]
```


```
100%|██████████| 1/1 [00:00<00:00, 22.80it/s]
```


```
100%|██████████| 1/1 [00:00<00:00, 27.96it/s]
```

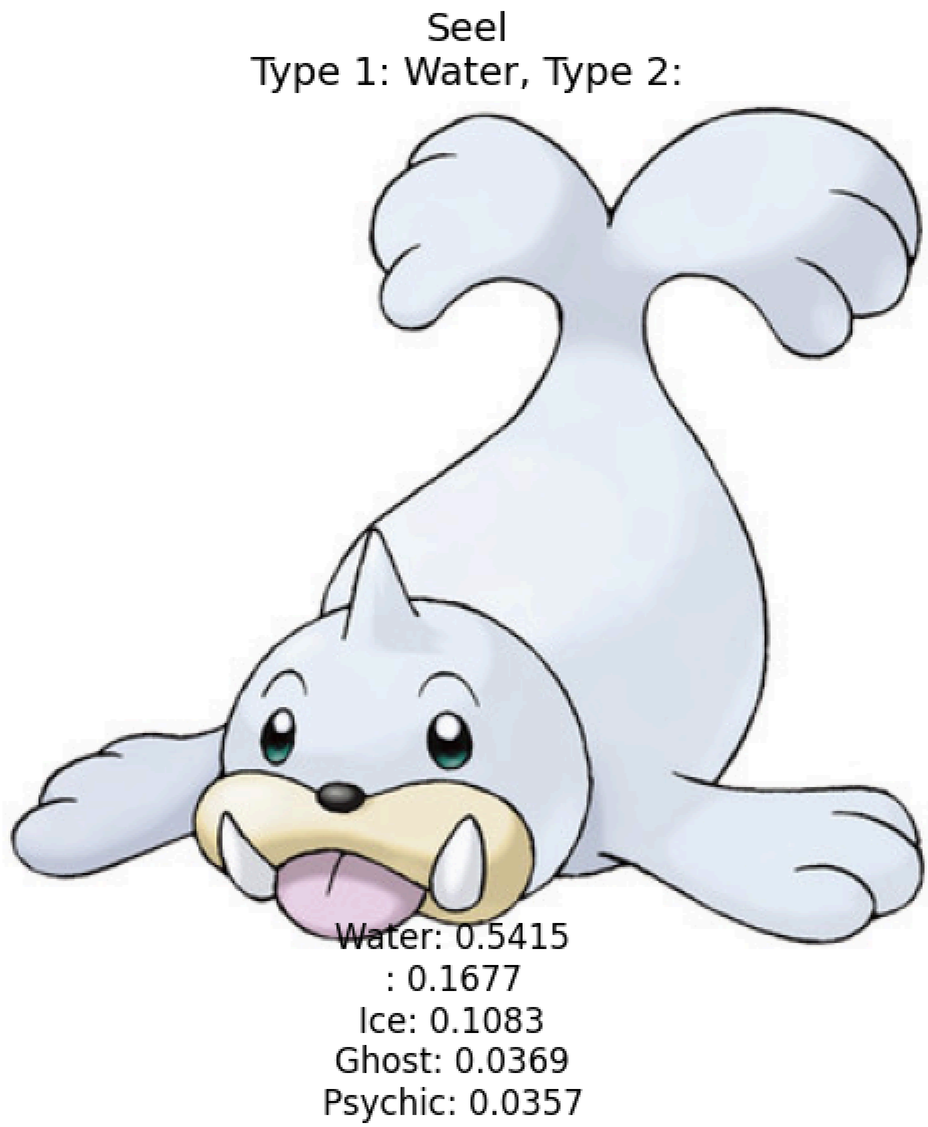

```
100%|██████████| 1/1 [00:00<00:00, 30.56it/s]
```

Lugia
Type 1: Psychic, Type 2: Flying

Water: 0.3005
Ice: 0.2693
Flying: 0.1334
Dragon: 0.0861
: 0.0530

```
100%|██████████| 1/1 [00:00<00:00, 30.62it/s]
```


```
100%|██████████| 1/1 [00:00<00:00, 27.43it/s]
```




```
100%|██████████| 1/1 [00:00<00:00, 25.65it/s]
```

QUESTION 28:

In the first and second question, we investigated how CLIP creates 'clusters' by mapping images and texts of various Pokemon into a high-dimensional space and explored neighbor- hood of these items in this space. For this question, please use t-SNE to visualize image clusters, specifically for Pokemon types Bug, Fire, and Grass. You can use scatter plot from python package plotly. For the visualization, color-code each point based on its first type type 1 using the 'color' argument, and label each point with the Pokemon's name and types using 'hover name'. This will enable you to identify each Pokemon represented in your visualization. After completing the visualization, analyze it and discuss whether the clustering of Pokemon types make sense to you.

In [ ]:

```
# Code for Q28
# Select Pokémon of types Bug, Fire, and Grass
selected_types = ["Bug", "Fire", "Grass"]
filtered_pokedex = pokedex[pokedex["Type1"].isin(selected_types)].reset_inde
```

```python
# Load CLIP model
model, preprocess, device = load_clip_model()

# Compute image embeddings using CLIP
image_embeddings = clip_inference_image(model, preprocess, filtered_pokedex[

# Reduce dimensions using t-SNE
tsne = TSNE(n_components=2, perplexity=15, random_state=42, metric="cosine")
tsne_results = tsne.fit_transform(image_embeddings)

plot_data = pd.DataFrame({
    "x": tsne_results[:, 0],
    "y": tsne_results[:, 1],
    "Name": filtered_pokedex["Name"],
    "Type1": filtered_pokedex["Type1"],
    "Type2": filtered_pokedex["Type2"]
})

# Create plot
fig = px.scatter(
    plot_data,
    x="x",
    y="y",
    color="Type1",
    hover_data={"Name": True, "Type1": True, "Type2": True}
)

# Show plot
fig.show()
```

```
100%|██████████| 192/192 [00:04<00:00, 42.88it/s]
```

Code for Q26

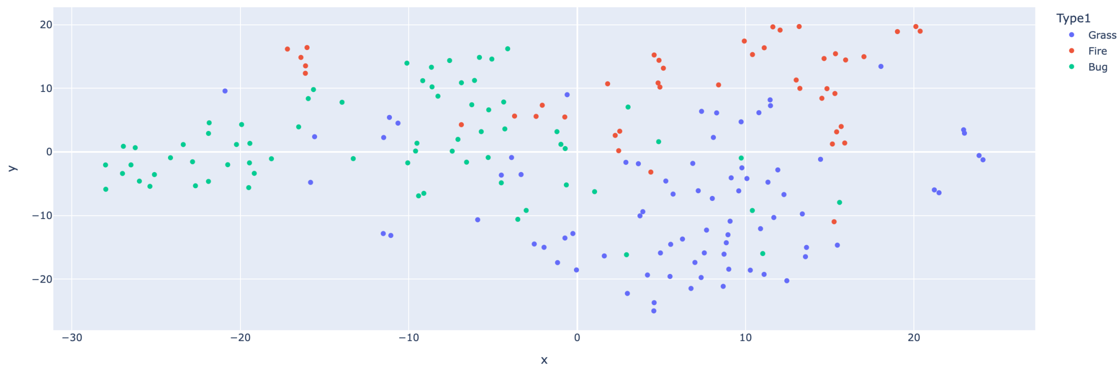
Load the Pokedex data

pokedex = construct_pokedex()

QUESTION 27:

Randomly select 10 Pokemon images from the dataset and use CLIP to find the most relevant types (use your preferred template, e.g "type: Bug"). For each selected Pokemon, please plot it and indicate:

- its name and first and second type;
- the five most relevant types predicted by CLIP and their predicted probabilities.


Type1
Grass
Fire
Bug
y
x
20
10
0
−10
−20
−30
−20
−10
0
10
20