This project includes a series of scripts designed to process PGM database. The primary objective is to enumerate all metadata in PGM instances and generate enumerated labels for each instance, which are then saved to a pkl file. Subsequently, you can use these labels to run the corresponding training code. If you are not using the SBR algorithm to optimize the Valen model, there is no need to specifically generate the enumerated labels.

Running the Metadata Enumeration Script

First, you need to run the read_tokens_pgm_pkl.py script to enumerate the metadata in all instances of the PGM and generate a pkl file with enumerated labels. This script will iterate through the PGM instances, extract relevant metadata information, and assign unique enumerated labels to each.

bash

```
python read_tokens_pgm_pkl.py
```

Distributed Training with PyTorch's torch.distributed

Here is a simplified example of how to use PyTorch's torch.distributed module to run distributed training:

bash

```
python -m torch.distributed.launch --nproc_per_node=4 Train_Funny_Valen_SBR.py
```

Or

```
python -m torch.distributed.launch --nproc_per_node=4 Train_Valen_Tine.py
```

Assuming you have 4 GPUs available, this command will start 4 processes on the current node, each using one GPU.