

File name: configs.py

`def Benchmarking`: Read the json file of NN description which includes FM shapes, kernel size, block size and the pruning ratio.

File name: traceGen.py

`class DataBlock`: The corresponding input and output channels of each data block.

`class Layer`: The description of one layer in NN. Also, the description of block and channels` order will be included.

`def SparseDataGen`: Fill in the description of each block.

File name: reorderEngine.py

`def AdrGen`:

1. Generate the bank id for each element in FM and weights.

Input FM: Firstly, gives all data in input FM with an id through ascending order. Then calculate the remainder (mod numBank) of each data, which equals to the bank id. Finally, the data are reshaped into the form according to GEMM.

Output FM: Same with input FM but do not need to reshape into GEMM form.

2. Get the input FM and output FM's bank id for each block.

File name: scheduler.py

`class PartialLayer`: Data volume and address for each partial layer.

`def Partition`: Find the source of input FM and the destination of output FM for each partial layer.

File name: perfModel.py

`def PerfSRAM`: Calculate the energy and latency under ideal or real data loading and writing.