Report: Include a list of all kernels that collectively consume more than 90% of the program time.

1. [CUDA memcpy HtoD]
2. Volta_scudnn_128x32_relu_interior_nn_v1
3. void cudnn::detail::implicit_convolve_sgemm<float, float, int=1024, int=5, int=5, int=3, int=3, int=3, int=1, bool=1, bool=0, bool=1>(int, int, int, float const *, int, float*,cudnn::detail::implicit_convolve_sgemm<float, float, int=1024, int=5, int=5, int=3, int=3, int=3, int=1, bool=1, bool=0, bool=1>*, kernel_conv_params, int, float, float, int, float, float, int, int)
4. void cudnn::detail::activation_fw_4d_kernel<float, float, int=128, int=1, int=4, cudnn::detail::tanh_func<float>>(cudnnTensorStruct, float const *, cudnn::detail::activation_fw_4d_kernel<float, float, int=128, int=1, int=4, cudnn::detail::tanh_func<float>>, cudnnTensorStruct*, float, cudnnTensorStruct*, int, cudnnTensorStruct*)
5. Volta_sgemm_128x128_tn
6. void cudnn::detail::pooling_fw_4d_kernel<float, float, cudnn::detail::maxpooling_func<float, cudnnNanPropagation_t=0>, int=0, bool=0>(cudnnTensorStruct, float const *, cudnn::detail::pooling_fw_4d_kernel<float, float, cudnn::detail::maxpooling_func<float, cudnnNanPropagation_t=0>, int=0, bool=0>, cudnnTensorStruct*, cudnnPoolingStruct, float, cudnnPoolingStruct, int, cudnn::reduced_divisor, float)
7. void mshadow::cuda::MapPlanLargeKernel<mshadow::sv::saveto, int=8, int=1024, mshadow::expr::Plan<mshadow::Tensor<mshadow::gpu, int=2, float>, float>, mshadow::expr::Plan<mshadow::expr::ScalarExp<float>, float>>(mshadow::gpu, unsigned int, mshadow::Shape<int=2>, int=2, int)
8. void mshadow::cuda::SoftmaxKernel<int=8, float, mshadow::expr::Plan<mshadow::Tensor<mshadow::gpu, int=2, float>, float>, mshadow::expr::Plan<mshadow::Tensor<mshadow::gpu, int=2, float>, float>>(mshadow::gpu, int=2, unsigned int)
9. void mshadow::cuda::MapPlanKernel<mshadow::sv::saveto, int=8, mshadow::expr::Plan<mshadow::Tensor<mshadow::gpu, int=2, float>, float>, mshadow::expr::Plan<mshadow::expr::ScalarExp<float>, float>>(mshadow::gpu, unsigned int, mshadow::Shape<int=2>, int=2)
10. volta_sgemm_32x32_sliced1x4_tn

Report: Include a list of all CUDA API calls that collectively consume more than 90% of the program time.

1. cudaStreamCreateWithFlags
2. cudaMemGetInfo
3. cudaFree
4. cudaFuncSetAttribute

5. cudaMemcpy2DAsync
6. cudaStreamSynchronize
7. cudaEventCreateWithFlags
8. cudaMalloc
9. cudaGetDeviceProperties
10. cudaMemcpy

Report: Include an explanation of the difference between kernels and API calls
Kernels are executed on the device in parallel by different CUDA threads.

Kernels is built on top of a lower-level API calls, which is also accessible by the application. Kernels provides an additional level of control by exposing lower-level concepts such as CUDA contexts - the analogue of host processes for the device - and CUDA modules - the analogue of dynamically loaded libraries for the device.

Report: Show output of rai running MXNet on the CPU

```
Loading fashion-mnist data... done
Loading model... done
New Inference
EvalMetric: {'accuracy': 0.8177}
19.33user 3.85system 0:13.18elapsed 175%CPU (0avgtext+0avgdata
5956088maxresident)k
0inputs+2856outputs (0major+1584783minor)pagefaults
0swaps
```

Report: List program run time
13.18s

Report: Show output of rai running MXNet on the GPU

```
Loading fashion-mnist data... done
Loading model... done
New Inference
EvalMetric: {'accuracy': 0.8177}
4.49user 2.54system 0:05.01elapsed 140%CPU (0avgtext+0avgdata
2840048maxresident)k
0inputs+4568outputs (0major+704659minor)pagefaults 0swaps
```

Report: List program run time
5.01s