ASO LAB Seminar
#week 5
Triton Server - nsight

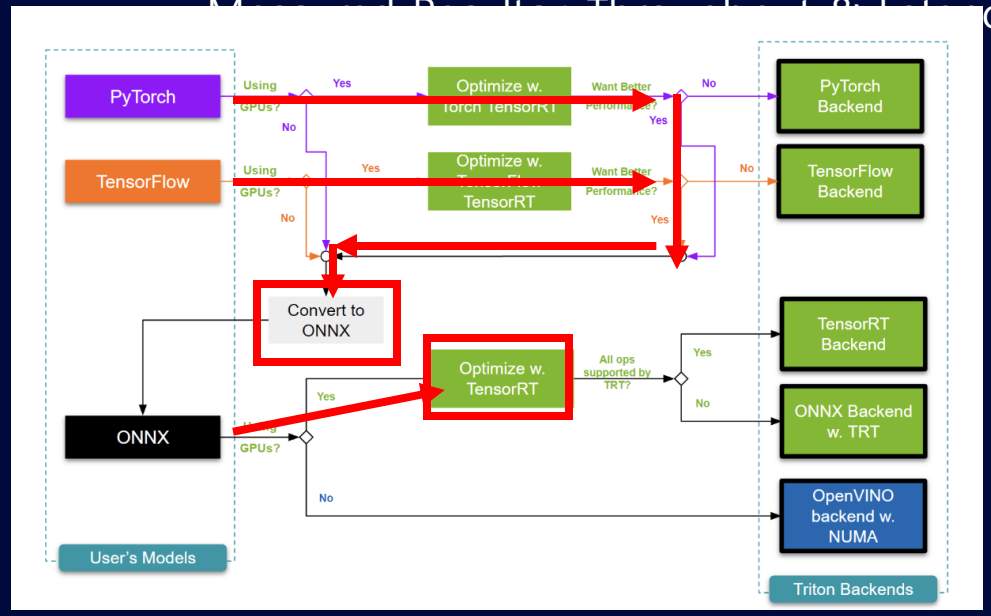
엄소은

CONTENTS

1. Previous Research

Research on Triton Server

- **A Deployment Scheme of YOLOv5 with Inference Optimizations Based on the Triton Inference Server**, 2021 IEEE 6th International Conference on Cloud Computing and Big Data Analytics
  - Deployment scheme of YOLOv5 with Triton Server
  - Model framework : Pytorch -> ONNX -> TensorRT
  - Measured Results : Throughput & Latency



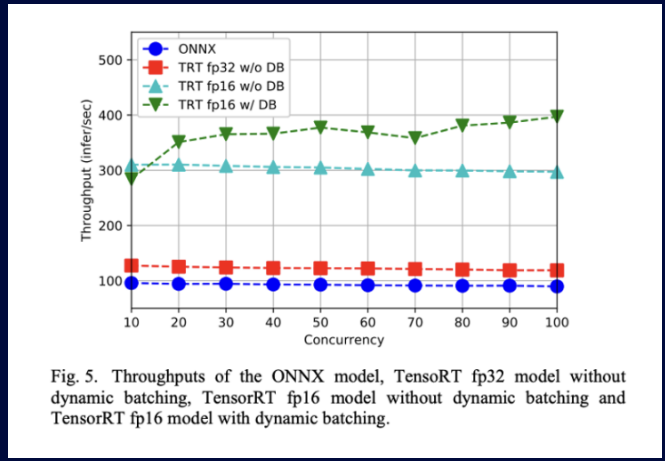


Fig. 5. Throughputs of the ONNX model, TensoRT fp32 model without dynamic batching, TensorRT fp16 model without dynamic batching and TensorRT fp16 model with dynamic batching.

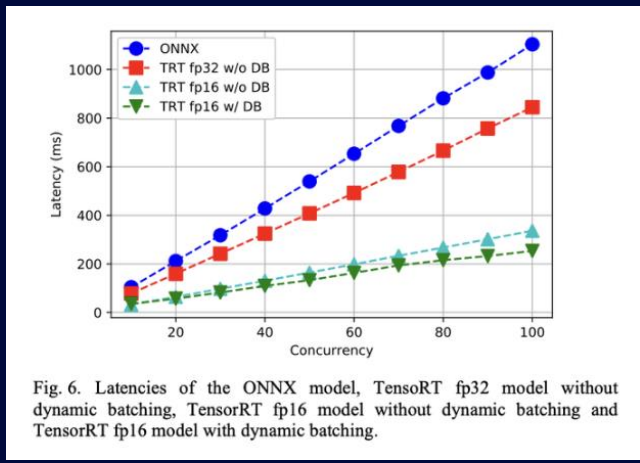


Fig. 6. Latencies of the ONNX model, TensoRT fp32 model without dynamic batching, TensorRT fp16 model without dynamic batching and TensorRT fp16 model with dynamic batching.

1. Previous Research

Research on Triton Server

- MLHarness: A Scalable Benchmarking System for MLCommons, BenchCouncil Transactions on Benchmarks, Standards and Evaluations
  - Triton Cold start problem - worse than other frameworks

The final Zhang's test was performed only with two software – TensorFlow Serving and NVIDIA Triton Inference Server. The researchers identified that Triton has a longer starting time in the case of the "cold start" test scenario, rather than TensorFlow Serving (see Figure 8). "Even for a small image classification model, it needs more than 10 seconds to prepare" [23]. According to the experiment, a long start time may challenge machine resource provisioning.

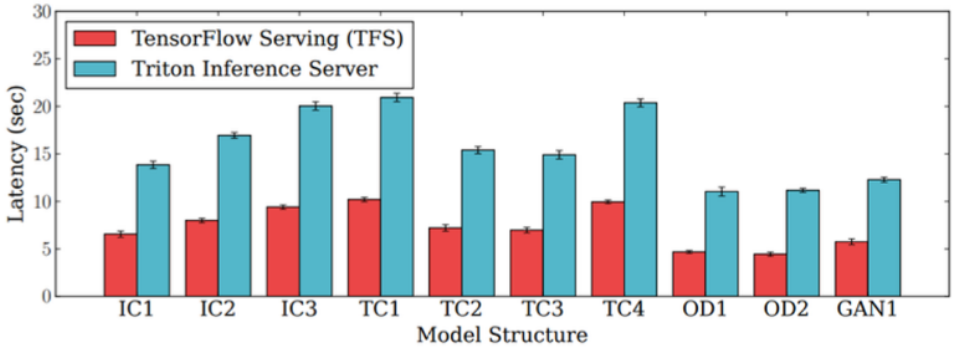
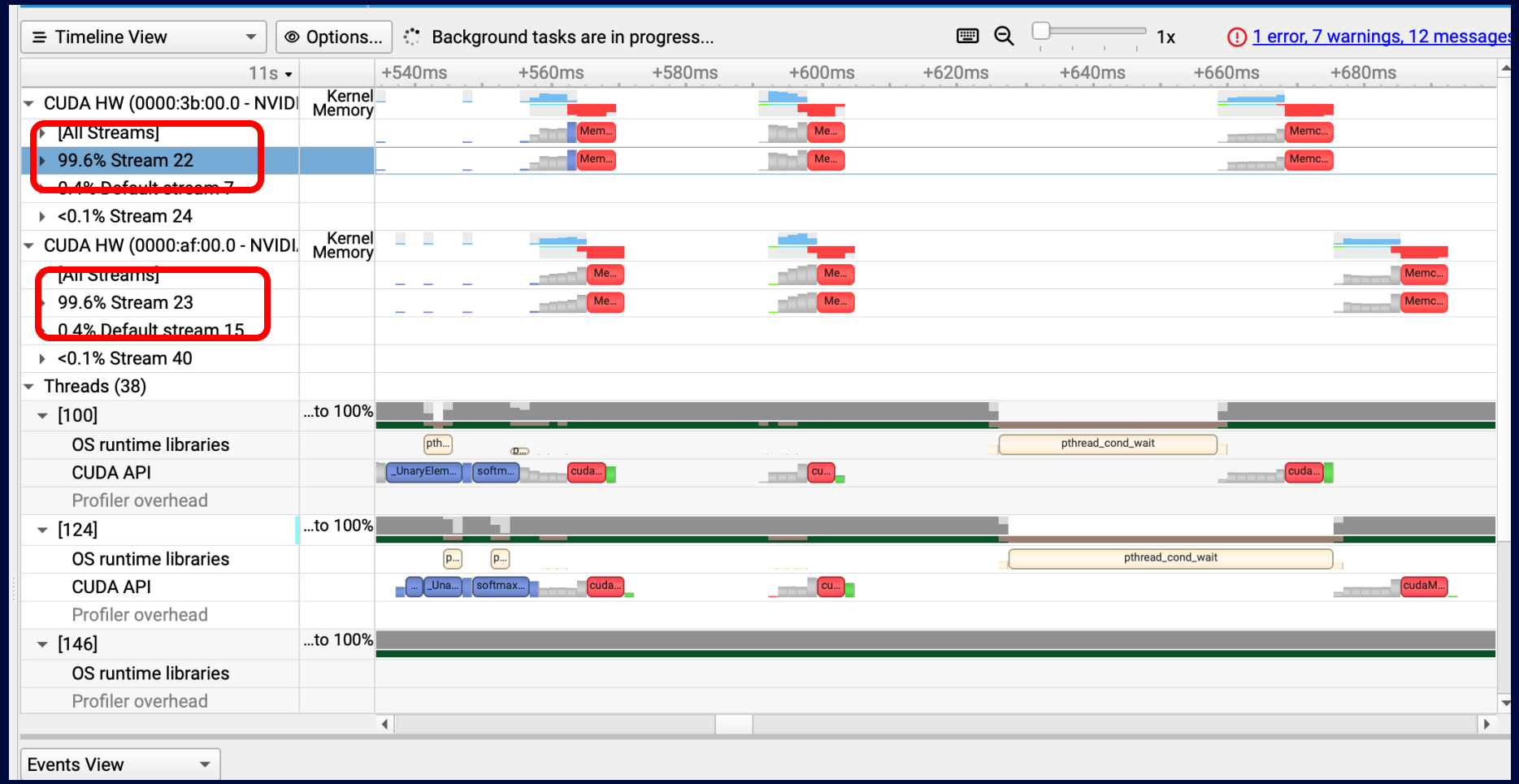


Figure 8. Cold start latency of different models with two serving software

2. Nsight Profile

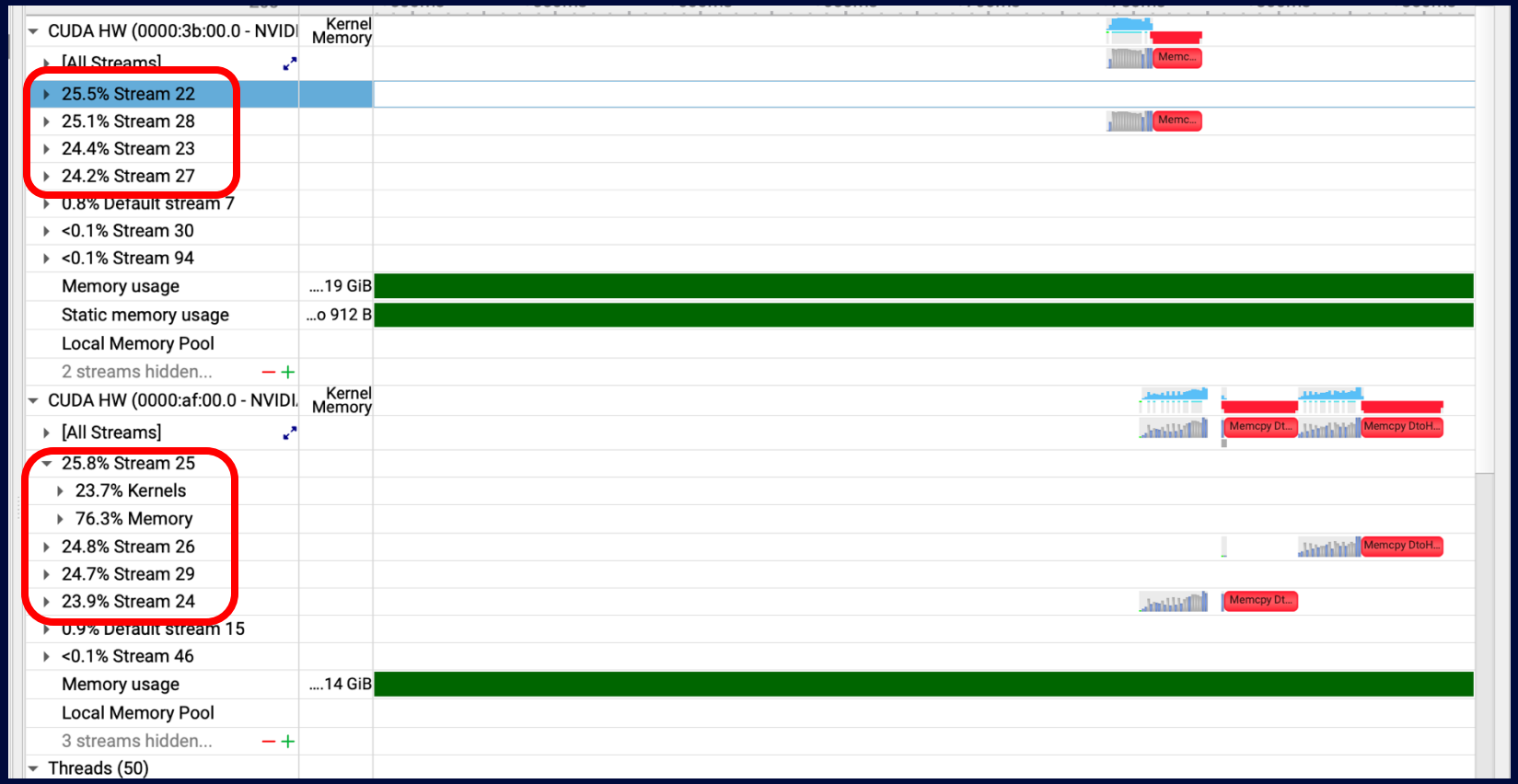
기본 세팅 (no optimization)

- BERT-base-uncased model, 110M params, onnx backend

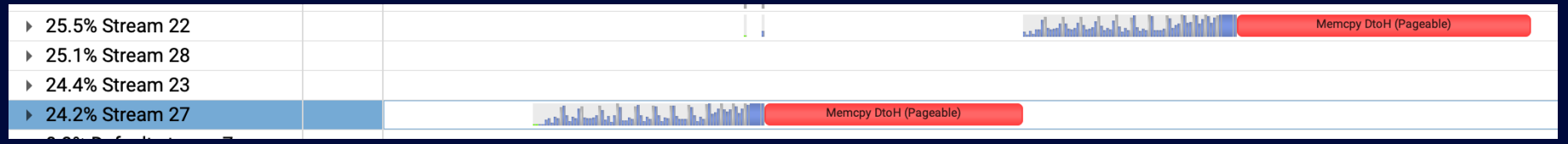


2. Nsight Profile

Multiple instances (instance = 4)



<- 각 GPU에 4개의 stream



<- stream 내에서 cuda 연산이 동시에 실행되지는 않음 ...

3. Optimization Analysis

Cache

- Inference request is the model name, model version, and input tensors (name, shape, datatype and tensor data) that make up a request submitted to Triton.
- Inference result is the output tensors (name, shape, datatype and tensor data) produced by an inference execution.
- response cache is used by Triton to hold inference results generated for previous executed inference requests
- Triton accesses the response cache with a hash of the inference request that includes the model name, model version and model inputs.

3. Optimization Analysis

Cache

```
void
DynamicBatchScheduler::CacheLookUp(
    std::unique_ptr<InferenceRequest>& request,
    std::unique_ptr<InferenceResponse>& cached_response)
{
  Status status;
  auto cache = model_->Server()->CacheManager()->Cache();
  std::unique_ptr<InferenceResponse> local_response;
  request->ResponseFactory()->CreateResponse(&local_response);
  // Hash request into cache key
  std::string key = "";
  if (!request->CacheKeyIsSet()) {
    status = cache->Hash(*request, &key);
    if (!status.IsOk()) {
      LOG_ERROR << "Failed to hash request: " << status.Message();
      return;
    }
    request->SetCacheKey(key);
  } else {
    key = request->CacheKey();
  }

  // Lookup and capture timestamps
  {
    request->CaptureCacheLookupStartNs();
    status = cache->Lookup(local_response.get(), key);
    request->CaptureCacheLookupEndNs();
  }

  if (status.IsOk() && (local_response != nullptr)) {
    cached_response = std::move(local_response);
#ifdef TRITON_ENABLE_STATS
    // Update model metrics/stats on cache hits
    // Backends will update metrics as normal on cache misses
    request->ReportStatisticsCacheHit(model_->MetricReporter().get());
#endif  // TRITON_ENABLE_STATS
  }
}
```

3. Optimization Analysis

Cache

```
python client.py

input_ids shape :  (2, 12) , attention_mask shape :  (2, 12) , token_type_ids shape :  (2, 12)
Results Logits shape: (2, 12, 30522)
Text: Skiing in a [MASK] is my favorite winter sport. -> Predicted token: blizzard
Text: The capital of [MASK] is beijing. -> Predicted token: china
Inference time: 0.242176 seconds

python client.py

input_ids shape :  (2, 12) , attention_mask shape :  (2, 12) , token_type_ids shape :  (2, 12)
Results Logits shape: (2, 12, 30522)
Text: Swimming in a [MASK] is my favorite summer sport. -> Predicted token: pool
Text: The best city to travel in [MASK] is seoul. -> Predicted token: korea
Inference time: 0.043731 seconds
```

- 전체 response cache 를 저장하고 따로 Model parameter 도 warm up 되어서 저장되는 것 같지만..

4. To-Do

- 연구 방향성 ?
  - 1) Cold start 문제 조사
  - 2) nsight 분석 방향성