Optimizing Recommendation System Inference Performance Based on GPU

Xiaowei Shen, Junrui Zhou, Kan Liu, Lingling Jin, Pengfei Fan, Wei Zhang
Alibaba Inc
{leyu.sxw, junrui.zjr, liukan.lk, l.jin, qiheng.fpf, wz.ww}@alibaba-inc.com

Jun Yang
University of Pittsburgh
juy9@pitt.edu

Motivation

WDL (wide and deep learning) based recommendation systems are widely adopted in E-commerce vendors' websites, such as Amazon, Taobao, Tmall to support billions of users. The system needs significant computing hardware to meet the requirement of query per second (QPS) and latency. Therefore, the efficiency of the system plays an important role on the cost of hardware. As more and more products and users the model needs to rank, the feature length and batch size (similar products to rank for a single user) of the model increase drastically so that traditional CPU implementation cannot achieve high QPS or even real-time query processing speed. Typical recommendation system receives thousands of user queries per second for billions of users. However CPU implementation for this model can only provide <10 QPS. GPU especially the latest T4 provides very high peak performance (8.1TFLOPS for FP32 and 65TFLOPS for FP16) for deep learning. We develop a CPU-GPU heterogeneous system to boost the inference performance of recommendation system based on GPU architecture. By node placement, model quantization and graph transformation, we can achieve **2.9x performance speedup** when compared with a baseline GPU implementation.

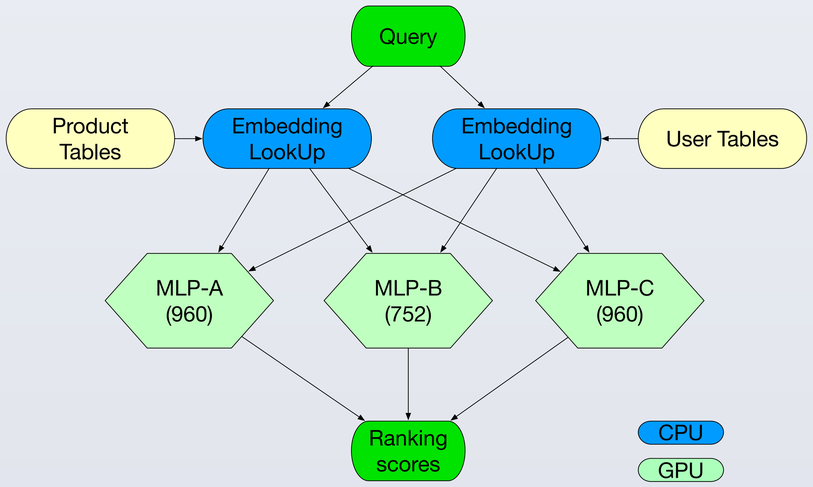


Figure 1. Recommendation system model

Node placement: embedding lookup on CPU and MLPs on GPU

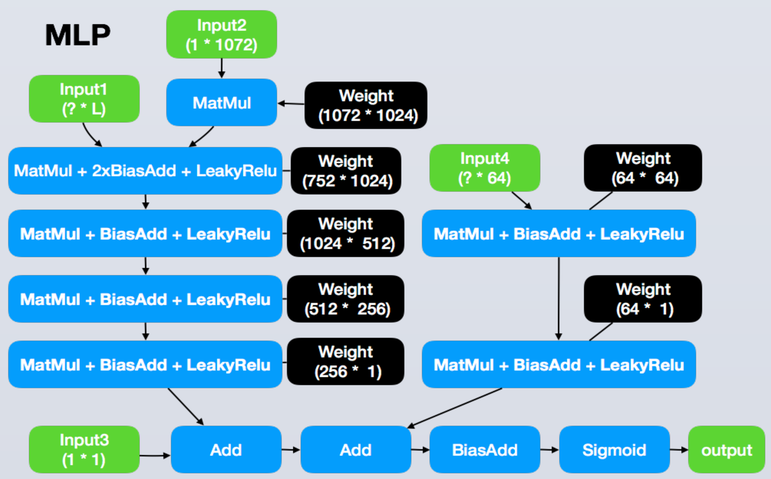


Figure 2. Model architecture of MLP

Model quantization: inputs and weights of each MatMul, biasadd and leakyrelu in each MLP are in FP16.

Methodology

- **Node placement**
  - based on node computing features and device features
- **Model quantization**:
  - based on GPU tensorcore support operations
- **Graph transformation**
  - to reduce GPU kernel launch time

Node Placement

- Embedding lookup on CPUs
- MLPs on GPUs

Model Quantization

- MatMuls, BiasAdd and LeakyRelu in FP16

Graph Transformation

- MLP batching
- Node fusing

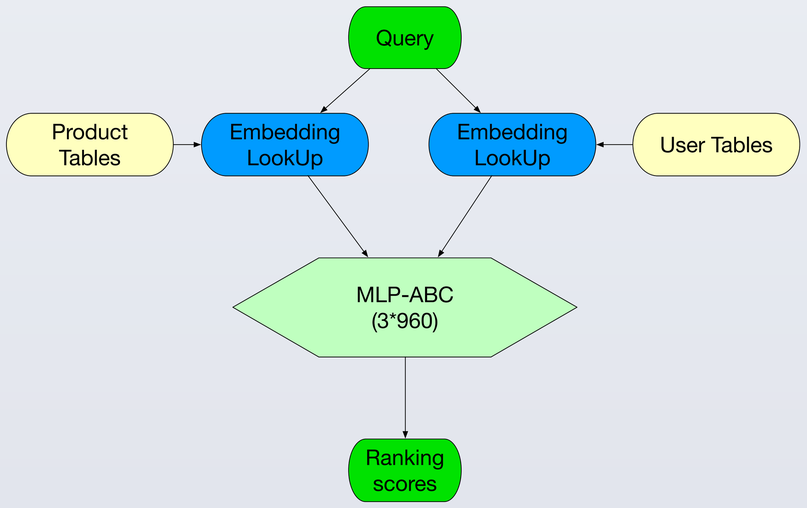


Figure 3. Recommendation system model after batching MLP

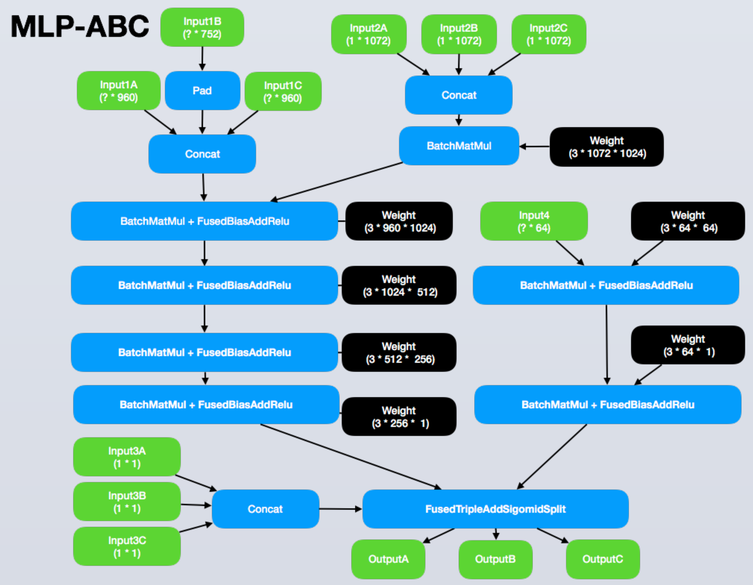
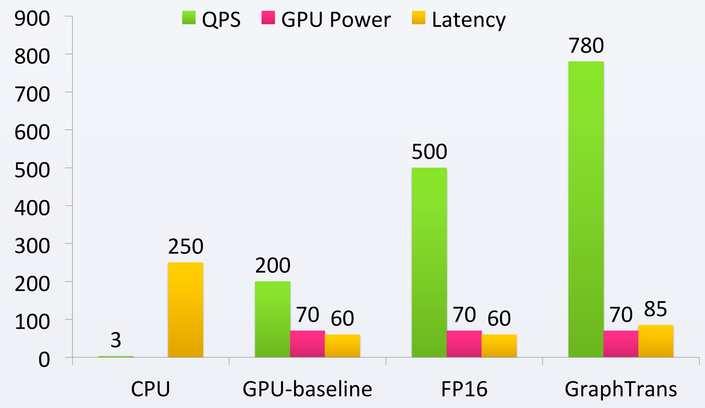


Figure 4. Three MLP architecture after batching and fusing

Experiments and Results



- **Performance (QPS)**
  - Over 200x speedup compared with CPU
  - 2.9x speedup compared with GPU baseline
- **Power**
  - Reached 70W peak power after using GPU before and after optimization
- **Latency**
  - Reduced by 66% to a tolerable latency compared with CPU
  - Increased by 41.7% but also a tolerable latency compared with GPU baseline
- **GPU nodes**
  - Reduced by 75% by graph transformation

Conclusions

Traditional implementation of recommendation system inference on CPU is becoming more difficult to meet the QPS and latency requirement of recommendation tasks because of the increasing of products and users. We develop a GPU based system to speedup recommendation system inference performance. First, we reduce the model precision from FP32 to FP16 to adopt the high peak performance of Tensor cores on GPUs without decreasing the prediction accuracy. Second, we place the model graph nodes to CPU and GPU based on their computing features and decrease the kernel launch overheads by batching and fusing the sub-graph on GPU. On the optimized CPU-GPU platform, we can achieve 2.9x performance speedup when compared with a baseline GPU implementation respectively. The system has been deployed in Alibaba to serve billions of users and reduces the hardware costs by 75% for the recommendation system.