

Weekly Research Progress Report

Student: 丛 兴 Date: 2023/12/6 - 2023/12/20

List of accomplishments this week

阅读关于稀疏矩阵存储格式的相关论文

学习曙光超算平台的使用方法、学习 slurm 的编写方法

完成算法作业、大规模作业

撰写并行计算机结构结课报告、机器学习结课论文（A Machine Learning-Based Model for Automated Selection of Task Granularity in SpMV Operations）

Paper summary

Name: MSREP: A Fast yet Light Sparse Matrix Framework for Multi-GPU Systems

Motivation: 由于现在计算的数据量超过了单个 GPU 的内存容量和计算能力，因此在将用于传统 CPU 上的 SpMV 计算核转移到 GPU 上去进行计算将无法直接进行运算，为此该团队基于这个 gap，开发了 MSREP 框架。

Solution: 该团队的解决方法是提出一个新型稀疏矩阵表示框架，该框架的大致思想是将原来经典稀疏矩阵的存储格式（CSR、CSC、COO）进行更加细粒度的划分（pCSR、pCSC、pCOO），分成更小的块后，再将这些小块分散到各个 GPU 核心上去进行计算。这样不仅仅可以解决单个 GPU 内存太小的问题，同时由于粒度更小，使得实现各个 GPU 核心的负载均衡更加容易。

Related to us: 为后面设计新的稀疏矩阵存储格式和计算核奠定基础。

Work summary

1. 本周阅读文献总结

（1）更加细粒度的划分

之前的划分方法都是通过行或列来进行划分任务的，现在直接基于非零元来进行划分，可以通过 GPU 的数量以及非零元的数量来更好的实现负载均衡，因为，这样可以使得每个 GPU 核上进行计算的非零元的个数基本上是一致的。

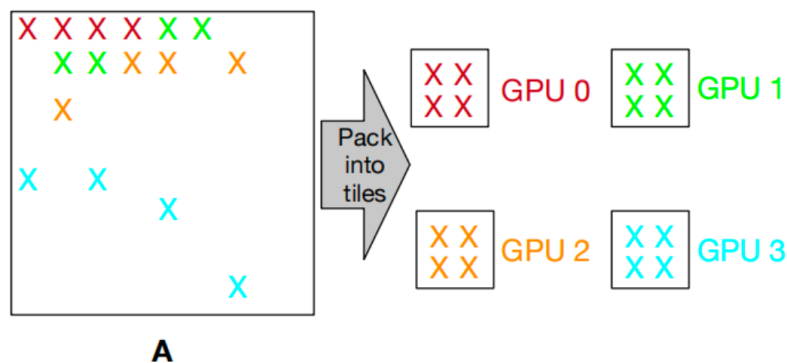


图 1 打破常规行或列的划分形式，进行更加细粒度的划分

(2) 扩展的 3 中稀疏矩阵的存储格式

该论文扩展了三种经典的稀疏矩阵存储格式，分别是 CSR、CSC、COO，扩展成为 pCSR、pCSC、pCOO。

由于 CSR 和 CSC 非常类似，而当 COO 数值排序按照行进行排序或者按照列进行排序时，转化方式甚至比前两者还要简单，因此，这里只对 pCSR 格式进行简单的转换总结。

首先，CSR 格式主要存储了 3 个东西，分别是 value、col_index、row_ptr。value 表示非零元，col_index 表示该非零元所处的行，row_ptr 表示某一个行前面的非零元个数，可以用来变相确定非零元的行数。

而 pCSR 在此基础上，主要是新增加了 4 个东西，如下图所示：

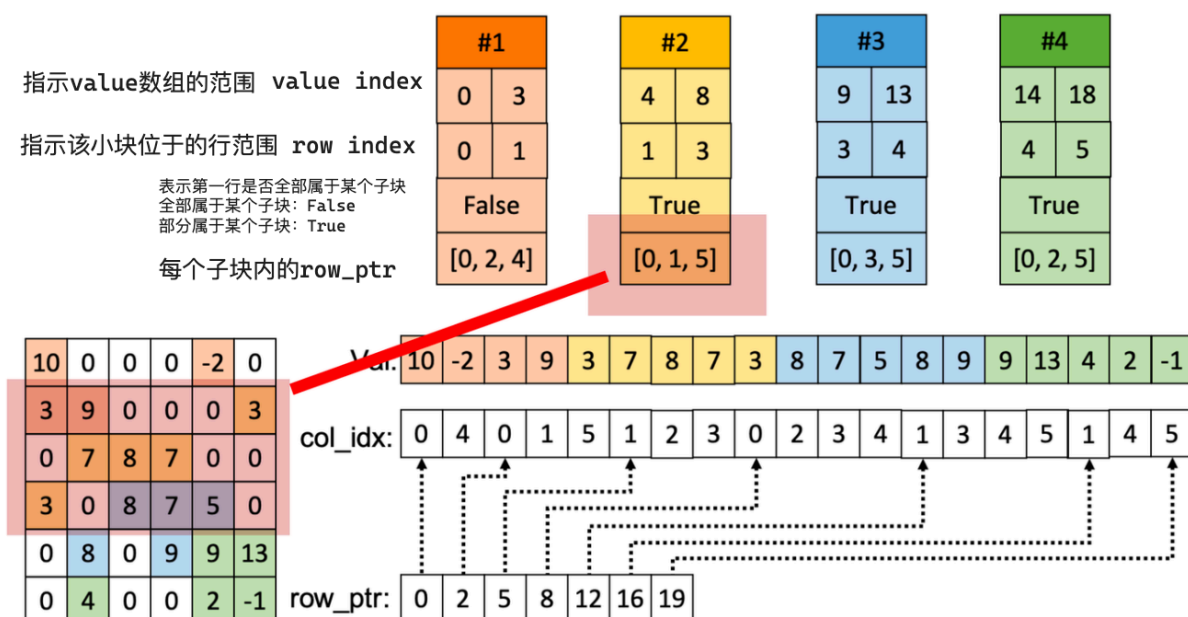


图 2 pCSR 存储转换示例

转化和合并实现方法如下图所示：

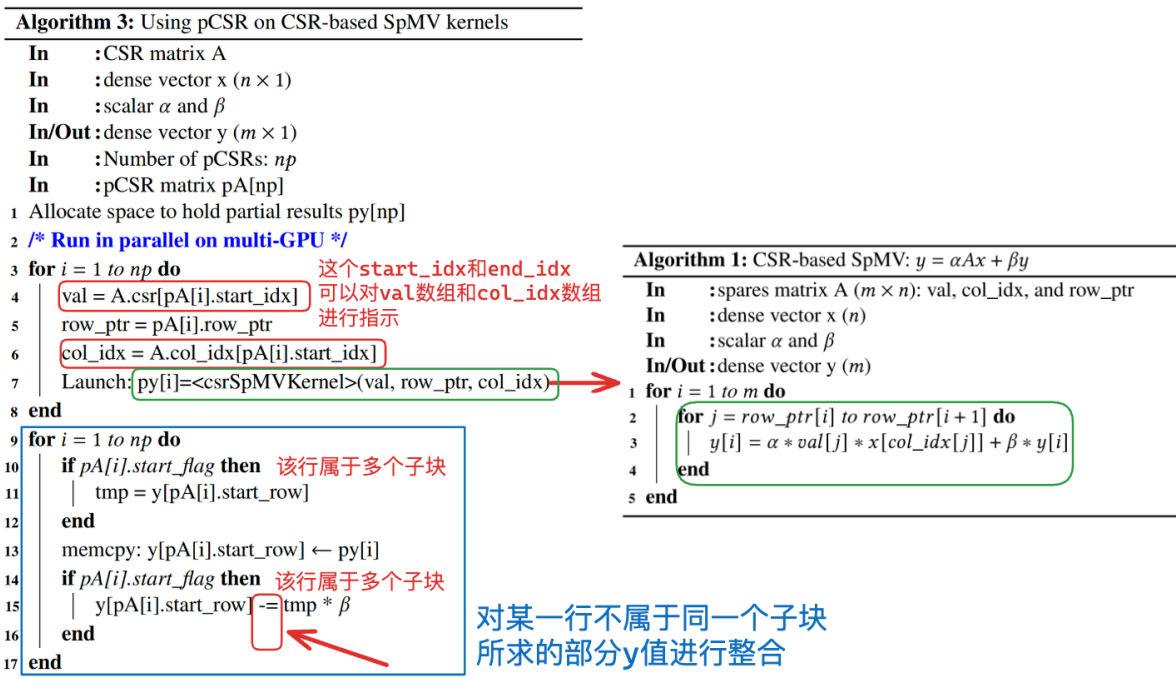


图 3 pCSR 转化的代码实现

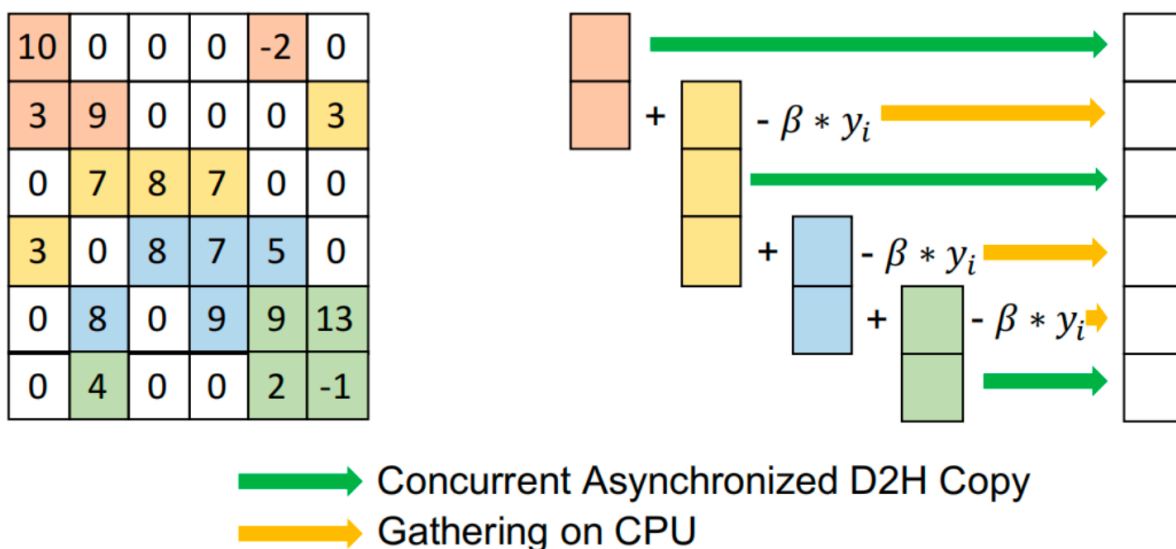


图 4 pCSR 的合并过程

2. 机器学习涉及到的论文

(1) 使用曙光超算平台、slurm 调度

可以先使用可视化界面进行提交，提交之后作业目录下会自动生成 slurm 脚本，直接在该脚本的基础上进行一定的修改即可。

调了一夜的 bug，一直报错，原来在多节点上使用 python 的 mpi 需要加一些命令，这和单节点上直接执行是不行的。修改后的命令如下：

```
mpirun -hosts=$SLURM_NODELIST -n $x python mpi-ml.py $id
```

其中，关键的是 `-hosts=$SLURM_NODELIST`，这个选项指定了要在哪些主机上运行程

序。`$SLURM_NODELIST` 是一个环境变量，通常由 SLURM 设置，包含了为当前作业分配的节点列表。

(2) 该论文的主要思路

通过超算平台进行计算，获得数千个稀疏矩阵的最优进程数量，同时提取出数千个矩阵的特征值，以矩阵的多个特征值以及所获得的最优进程数量结果进行机器学习训练，得出准确率较高的模型，想要达到的效果是，等下次运行某个矩阵时，直接将某个矩阵的特征输入到该模型中，该模型可以自动推测出取得最优性能的进程数量，也就是任务分配，相当于每个进程所计算的稀疏矩阵的行数。

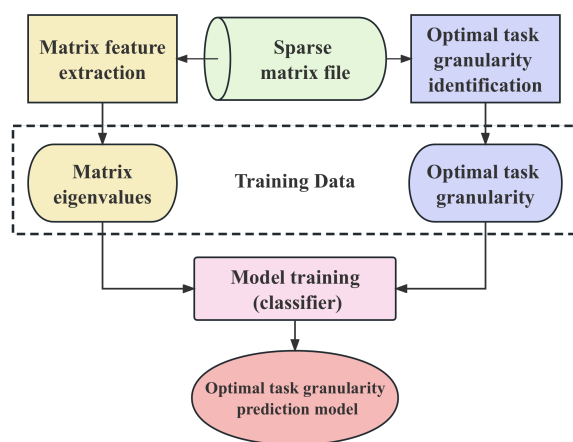


图 5 最优预测模型实现流程图

(3) 通过超算平台计算获得的结果

本次计算采用了 1627 个稀疏矩阵样本进行实验。这些样本的非零元素的数量范围为 0-100000。

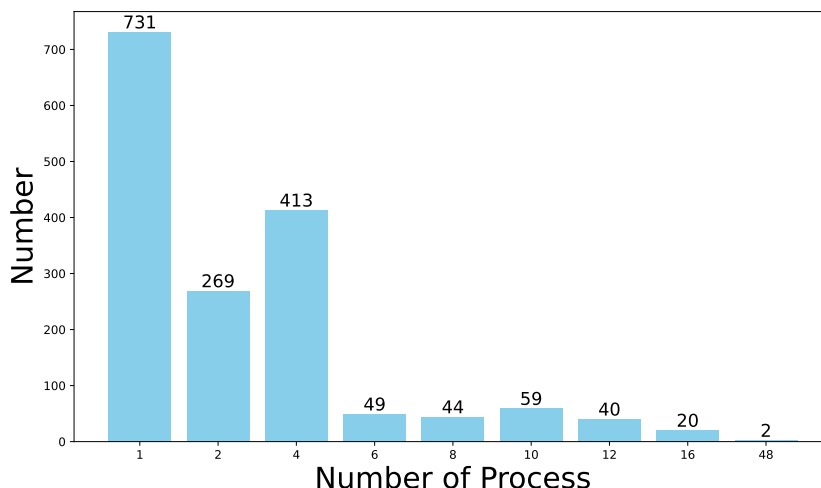


图 6 每个进程取得最佳性能，每种进程数量对应的稀疏矩阵数量

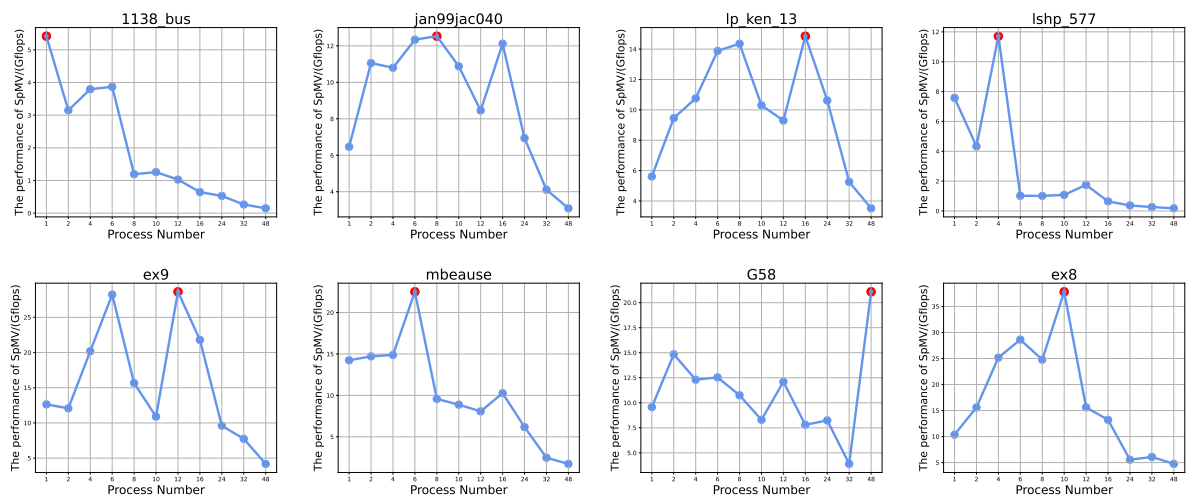


图 7 不同矩阵对应不同的任务粒度而达到最优

(3) 采用的矩阵特征

Feature	Meaning
n_rows	Number of rows in sparse matrix
n_cols	Number of columns in sparse matrix
nnz_frac	Fraction of non-zero elements
nnz_min	Minimum number of nnz elements
nnz_max	Maximum number of nnz elements
nnz_avg	Average number of nnz elements
nnz_std	Standard deviation of nnz elements

图 8 提取的矩阵的 7 个特征

(4) 采用的训练方法

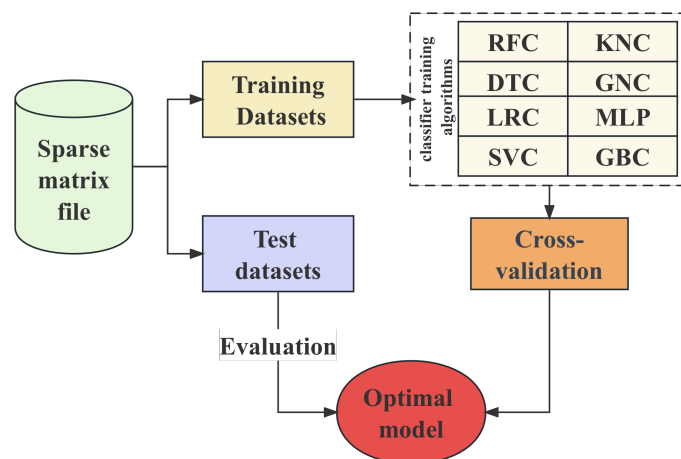


图 9 预测模型选择和评估流程图

(5) 模型表现

准确率不佳，最高的是决策树模型，可以达到 65% 左右。

Next

阅读 HiCOO 篇论文