

공동 활용 GPU 클러스터 시스템의 사용자 작업 통계 분석 기법에 대한 연구

권민우*, 홍태영*

*한국과학기술정보연구원 슈퍼컴퓨팅인프라센터

mwkwon81@kisti.re.kr

A study on the statistical analysis technique of user job of shared utilization GPU cluster system

Min-Woo Kwon*, TaeYoung Hong*

*Dept. of Supercomputing Infrastructure Center, KISTI

요 약

공동 활용 GPU 클러스터 시스템은 PBS, SLURM과 같은 작업 배치 스케줄러 및 Kubernetes, Openstack과 같은 클라우드 플랫폼 소프트웨어를 통해 다수의 사용자에게 공동 활용 서비스를 할 수 있다. 한국과학기술정보연구원(KISTI)에서는 슈퍼컴퓨터 5호기의 보조시스템인 뉴론을 SLURM 배치 스케줄러를 통해 다수의 연구자들에게 서비스하고 있다. 본 논문에서는 작업 배치 스케줄러에서 생성된 과금 통계 데이터를 R의 그래프 함수를 이용해 시각화하여 사용자 작업의 특성을 분석하고 효율적인 계산 자원 관리 정책을 수립할 수 있는 방안에 대해 논의한다.

1. 서론

4차 산업혁명의 핵심동력인 인공지능 기술은 의료, 자동차를 비롯한 산업 전반에서 획기적인 변화를 일으키고 있다. 인공지능 기술의 핵심인 빅데이터, 머신러닝을 활용한 연구를 가속화하기 위해 GPU를 장착한 고성능의 인프라에 대한 필요성이 높다. 이러한 고성능의 인프라는 GPU 뿐만 아니라 고성능의 네트워크 및 공유 파일 시스템 등을 포함하고 있어 높은 구축비용과 함께 일반 연구자들이 운영하기에 큰 부담을 준다.

KISTI에서는 고성능의 AI 인프라에 대한 수요에 대응하기 위해 슈퍼컴퓨터 5호기의 보조시스템인 뉴론을 다수의 연구자들에게 공동 활용 서비스로 제공하고 있다[1]. 연구자들은 뉴론에 접속하여 conda 패키지 설치 및 singularity를 이용한 docker 컨테이너 이미지를 이용하여 AI 소프트웨어를 사용할 수 있다. 최근에는 뉴론 시스템에서 Jupyter 노트북 서비스를 시작하여 GPU가 장착된 서버에서 AI 코드 개발 및 디버깅이 훨씬 수월하게 되었다.

공동 활용 GPU 클러스터 시스템은 PBS, SLURM과 같은 작업 배치 스케줄러나 Kubernetes, Openstack과

같은 클라우드 플랫폼 소프트웨어를 이용해 다수의 사용자에게 서비스할 수 있다. 뉴론 시스템은 SLURM 배치 스케줄러를 통해 다수의 연구자들에게 GPU가 장착된 계산 자원을 효율적으로 제공하고 있다. 작업 배치 스케줄러나 클라우드 플랫폼 소프트웨어들은 자원 사용에 대한 이용요금을 부과하기 위해 과금 로그를 생성해내는데, 이러한 과금 로그는 효율적인 자원 운영을 위한 중요한 데이터이다 [2]. 본 논문에서는 SLURM 배치 스케줄러에서 생성된 사용자의 작업별 과금 로그 데이터를 R의 그래프 함수를 이용하여 분석하여 사용자 작업의 특성을 이해하고 효율적인 자원 관리 정책을 수립할 수 있는 방안에 대해 소개한다.

2. SLURM 배치 스케줄러 과금 데이터 추출

SLURM 배치 스케줄러는 모든 사용자의 과금 데이터를 MariaDB에 저장한다[2]. 그림 1과 같이 sacct 커맨드를 이용하여 JobID, JobName, Start, End, State 정보들을 데이터베이스에서 추출할 수 있다. '-o' 옵션 뒤에 추가적인 필드를 입력하여 원하는 정보를 추출할 수 있다.

```

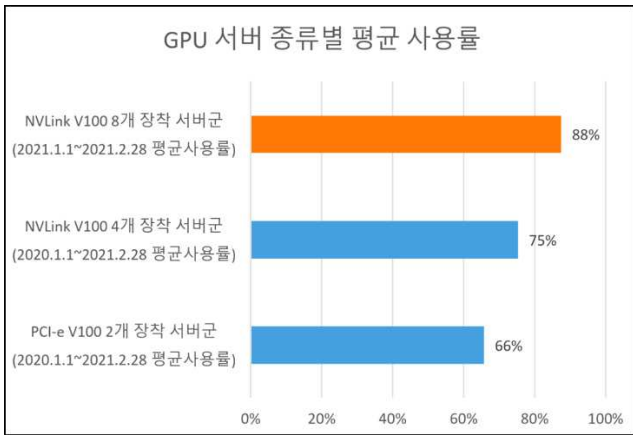
$ sacct -T -X -ojobid,jobname,start,end,state
-----
JobID   JobName      Start              End                State
-----
61393   bash         2021-03-16T09:42:51 2021-03-16T10:35:55 FAILED
61413   bash         2021-03-16T10:36:21 2021-03-16T10:38:13 FAILED
61416   bash         2021-03-16T10:38:16 2021-03-16T10:54:10 FAILED
61419   bash         2021-03-16T11:01:12 2021-03-16T13:04:55 COMPLETED
    
```

(그림 1) 응용SW별 사용량

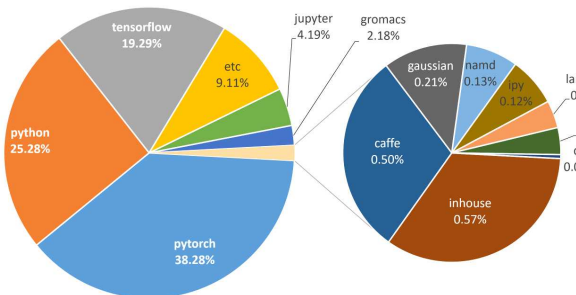
뉴론 시스템에서는 매일 Perl 스크립트를 이용하여 MariaDB에 직접 SQL 쿼리문을 전송하여 과금 정보를 테이블 형태로 저장하고 있다. 추가적으로 SLURM 배치 스케줄러에서 제공해주지 않는 사용자 작업별 GPU 사용 통계 정보를 Perl 스크립트로 계산하여 과금 테이블에 함께 저장하고 있다[3].

3. 뉴론 시스템 사용량 통계

그림 2는 2020년 1월1일부터 2021년 2월 28일까지 최근에 구축된 GPU 서버 종류별 사용률을 보여준다. 그림 3은 이 기간 뉴론 시스템에서 수행된 사용자 작업의 응용SW별 사용량(작업 수행시간 기준)을 보여준다. 해당 기간 동안 수행된 전체 작업 개수는 39,470개이며, 483명의 연구자들의 의해 작업이 수행되었다.



(그림 2) GPU 서버 종류별 사용량

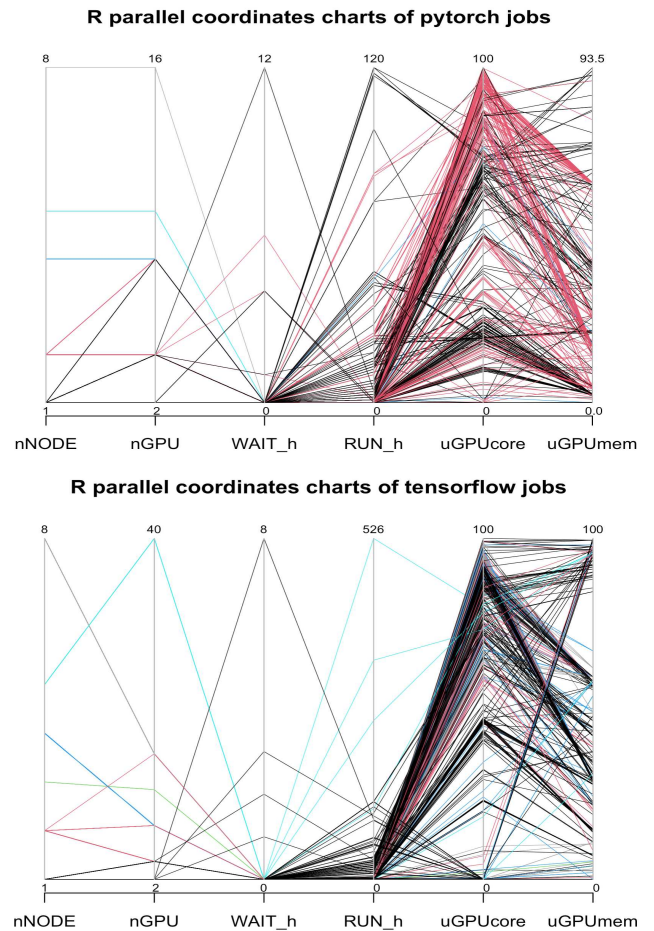


(그림 3) 응용SW별 시스템 사용량

응용SW별 사용량은 pytorch 38.28%, python 25.28%, tensorflow 19.29%, caffe 0.50%로 뉴론에서 수행된 사용자의 작업 중에서 AI 관련 작업이 80% 이상을 차지하는 것을 알 수 있다.

4. R을 이용한 사용자 작업 통계 분석

그림 4는 R의 parallel coordinates 그래프(이하 PCC)를 이용하여 AI 관련 작업 중에서 사용률이 높은 pytorch, tensorflow의 2021년 1월1일부터 2021년 2월 28일까지 수행된 모든 작업에 대한 작업별 사용 패턴을 시각적으로 표현한 것이다[4].



(그림 4) R parallel coordinates 그래프

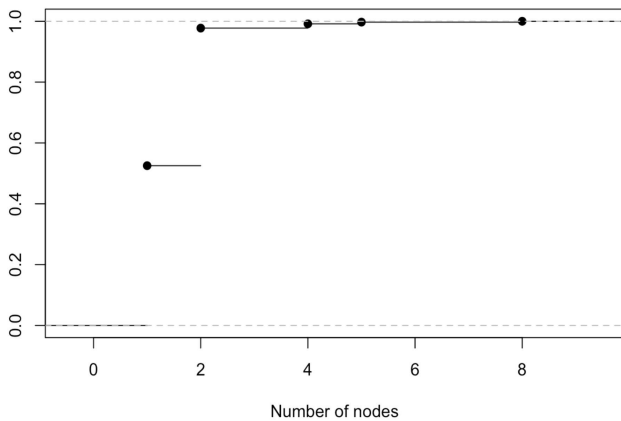
그래프의 y축의 값의 의미는 표1과 같다. 이 중에서 uGPUcore, uGPUmem은 작업의 수행시간 동안 축적된 nvidia-smi 커맨드에서 출력되는 core, memory의 평균값이다[3]. 이러한 통계 정보를 통해 뉴론 시스템에서 수행 중인 pytorch, tensorflow 작업이 얼마나 GPU 카드를 효율적으로 사용하고 있는지를 확인할 수 있다.

<표 1> R parallel coordinates 그래프 y축 값의 의미

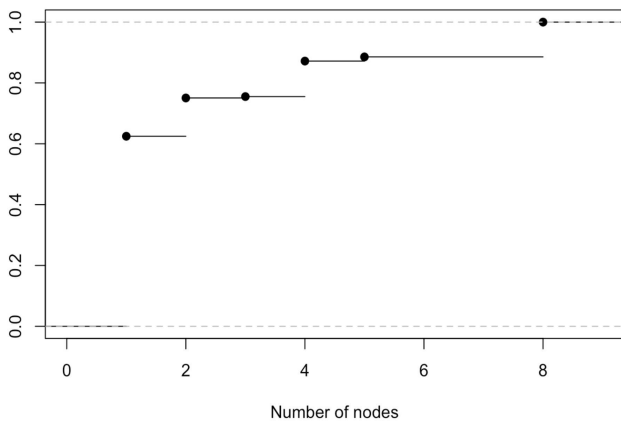
명칭	정보
nNODE	작업이 사용한 노드개수
nGPU	작업이 사용한 GPU개수
WAIT_h	작업의 대기시간(hour)
RUN_h	작업의 수행시간(hour)
uGPUcore	작업의 GPU core 사용률
uGPUmem	작업의 GPU memory 사용률

PCC를 통해 사용자 작업의 분포를 한눈에 확인할 수 있는데, pytorch, tensorflow 작업들의 경우, 사용한 노드 개수는 1~8개, GPU개수는 2~16, 2~40 개 사이에 분포되어 있음을 알 수 있다. 대기시간은 각각 0~12시간, 0~8시간 사이에 분포되어 있으며, 대다수의 작업이 0시간, 즉 대기시간 없이 즉시 실행되었음을 알 수 있다. 실행시간은 각각 0~120시간, 0~526시간 사이에 분포되어 있다. 그림 4의 PCC 그래프는 사용한 노드 개수별 선의 색상을 달리하여 노드 개수별 작업 패턴을 한눈에 확인할 수 있다.

R cumulative distribution function of pytorch jobs



R cumulative distribution function of tensorflow jobs



(그림 5) R 누적분포함수 그래프

GPU core, memory usage의 경우 pytorch에 비해 tensorflow 작업들이 50% 이상 지점에 다수 분포되어 있어 좀 더 효율적으로 GPU를 사용하고 있음을 알 수 있다. PCC의 경우, 작업패턴이 겹치는 nNODE와 같은 요소는 단일선으로 표기되므로 추가적으로 그림 5와 같은 누적분포함수를 통해 작업 분포를 확인할 수 있다[5]. pytorch의 경우 대략 47%의 작업이 멀티노드(2노드 이상)를 사용하여 작업을 수행하였으며, 이는 tensorflow가 멀티노드를 사용한 작업이 38%인 것에 비해 높은 것으로 멀티노드를 활용한 작업 수행이 활발하게 이루어지고 있음을 알 수 있다.

3. 결론 및 향후 연구 방향

뉴론 시스템은 R의 PCC 그래프와 누적분포함수를 주기적으로 생성하여 노드 및 GPU의 활용률, 대기시간 등을 확인하여 사용자가 제출 및 실행할 수 있는 최대 작업의 개수 및 노드와 GPU 점유 개수 등을 유동적으로 조정하여 사용자들이 오랜 대기 시간 없이, 작업 수행 목적에 맞게 시스템을 원활하게 사용할 수 있도록 서비스를 제공하고 있다. 뉴론 시스템과 같은 작업 스케줄러를 사용하지 않는 일반적인 클러스터 시스템에서도 별도의 로그 데이터를 생성하여 본 논문에서 제안하는 분석 기법을 활용해 효율적인 시스템 운영이 가능할 것으로 기대한다. 향후에는 과금 데이터를 활용하여 대기 시간에 영향을 주는 요소를 분석하여 사용자가 제출한 작업의 대기 시간을 예측하고 작업 배치 스케줄러에서 이러한 요소에 능동적으로 대응할 수 있는 자동화된 기술에 대한 연구를 수행할 예정이다.

참고문헌

[1] National Supercomputing Center, KISTI 뉴론 시스템, <https://www.ksc.re.kr/ggspect/neuron>
 [2] SLURM workload manager, SLURM accounting, <https://slurm.schedmd.com/accounting.html>
 [3] 권민우, 윤준원, 홍태영 “클러스터 시스템에서 GPU 사용 통계정보 획득 방안에 대한 연구” 2018년 춘계학술발표대회, 476-477, 2018.
 [4] R Graph Gallery, R parallel coordinates chart, <https://www.r-graph-gallery.com/parallel-plot.html>
 [5] RPubs by RStudio, R cumulative distribution function, https://rpubs.com/tgjohnst/cumulative_plotting