2017101319 한동훈

MapReduce: Simplified Data Processing on Large Cluster

About MapReduce

Mapreduce is software framework developed for processing large clusters of data by Google. MapReduce is model for processing large amount of data. **Mapper** function process key/value pair from input and process them to intermediate key/value. **Reduce** function merges all the intermediate values with same intermediate key. Job is done by large cluster of commodity machine. (parallelized and executed)

Model Explanation

Model is made of three big stages. First is mapper function. Second stage is Shuffling. Third stage is Reducer function. Mapper function은 input 값에서 해당 자료를 식별하기 위한 키 값을 추출한다. 그리고 value 값은 원래 input 값을 넣거나 필요한 값으로 보정하여 넣는다. 이렇게 되면 intermediate key/value가 형성된다. Intermediate 쌍은 Shuffle에 의해 목적에 맞게 grouping, sorting 된다. 그리고 정리된 intermediate key/value 쌍은 Reducer function으로 넘어가서 같은 키 값을 가진 value끼리 묶어내는 과정을 수행한다. 이 때 통사적으로 emit _intermeidate()과 emit() 이 각 함수에서 결과 값을 뽑아내는 역할을 한다.

For your intuition (Examples)

-Distributed Grep: map()->제공된 패턴과 일치하면 emit, reduce()->최종 결과에 제공 받은 정보를 넘겨주는 역할.

-Count of URL Access Frequency: 웹로그를 input으로 받아서 url은 키, 1을 value로 구성. Reduce()는 같은 키값(url)을 모아서 emit()을 실행한다. (최종적으로 <URL, total count>가 결과 변수에 저장된다.)

-Inverted Index: Document들이 input. 이를 map()에서 <word, document ID> pair들을 추출한다. 이를 키 값에 따라 정렬한 후에 reduce()에서 <word, list(doc ID)>페어로 정리한다. 결국 특정 단어를 포함하는 문서를 결과로 얻는 것이다.

++input record에서 key 값을 파싱하고, 등등...

How it actually works

-Master(s) and workers

1. Split input into M pieces. Start up program on cluster of machines. Master machine checks workers' status
2. One of them will be master. Master assign works to workers. Mater assign map task or reduce task to idle workers.
3. Map workers generate intermediate key/value. Result is buffered in memory.
4. Buffer to local disk(map machien의 로컬 저장소). (partitioned in R regions) ~~아마 이게 shuffle, sorting 인 듯~~. Location information passed to master. Master will forward tasks with location to

reduce workers.
5. Reduce workers get location information and use remote procedure call to access remote storage. Then, sort intermediate data with given key (Grouping with key).
6. Iterate through sorted data. Pass key and value to reduce function. Function will append final result to result variable using "emit".
7. After map and reduce function finished, master wakes user program.

Additional Information
-Fault toleration:
Worker failure: If no response from worker, consider mark him as failed. Failed workers are reset to idle state. Failed라고 표시된 기계에서도 task가 정상적으로 수행되기도 하는데 이러한 task는 master가 알 수 없어서 다시 실행된다. (map에서만 그럼, 저장 위치를 몰라서) 이렇게 다시 실행된 map task는 reduce worker들에게 알려지고 이 저장 위치를 remote read 하게 된다.
Master failure: 주기적으로 checkpoint를 남기는데 master가 죽으면 다른 마스터가 이 체크포인트를 이어받아서 계속 실행한다. 마스터가 하나라면 다시 실행해야한다.

-Locality: M, R개로 쪼개서(일반적으로 machine의 갯수보다 많게) 그리고 쪼게진 조각의 크기가 작아서 dynamic load balancing이 효과적이다.
-Task Granularity
-Backup Task

Additional Information
읽어야함

Experiment/Implementation
실험 환경 설명해주기 (1TB input 등등)
Grep: 여러 개의 파일 내에서 특정한 문자열 패턴 추출.
Sort: 텍스트에서 10바이트 키 추출

Related Work