Decentralized Word Count

Overview:

In this assignment, you will simulate a multi-server processing system to allow for the processing of a text file across multiple systems. The resulting cloud software will take as input a single text document, split the data across multiple systems, process the data and return information on the document to the client.

Technology:

Language: Java

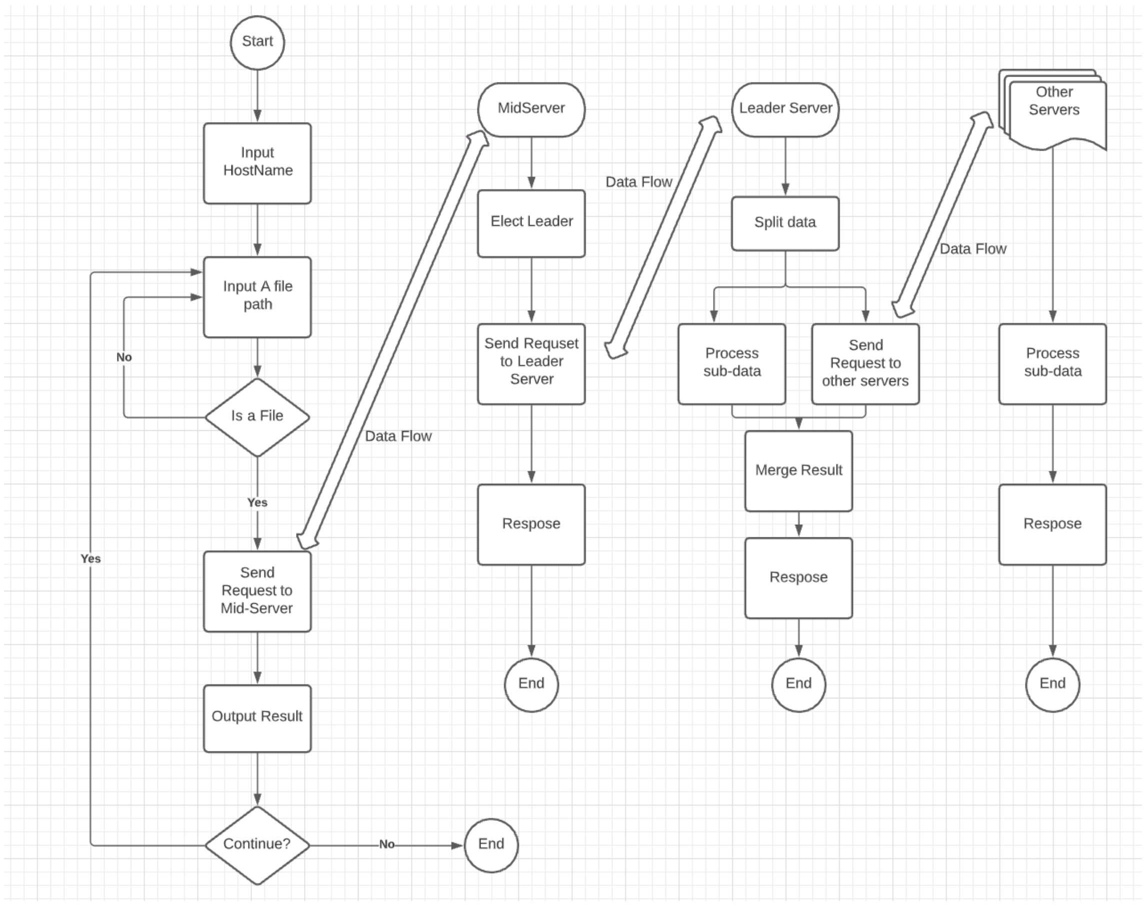
Develop Tool: Intellij IDEA, Docker

Build Tool: Apache maven, Docker-compose

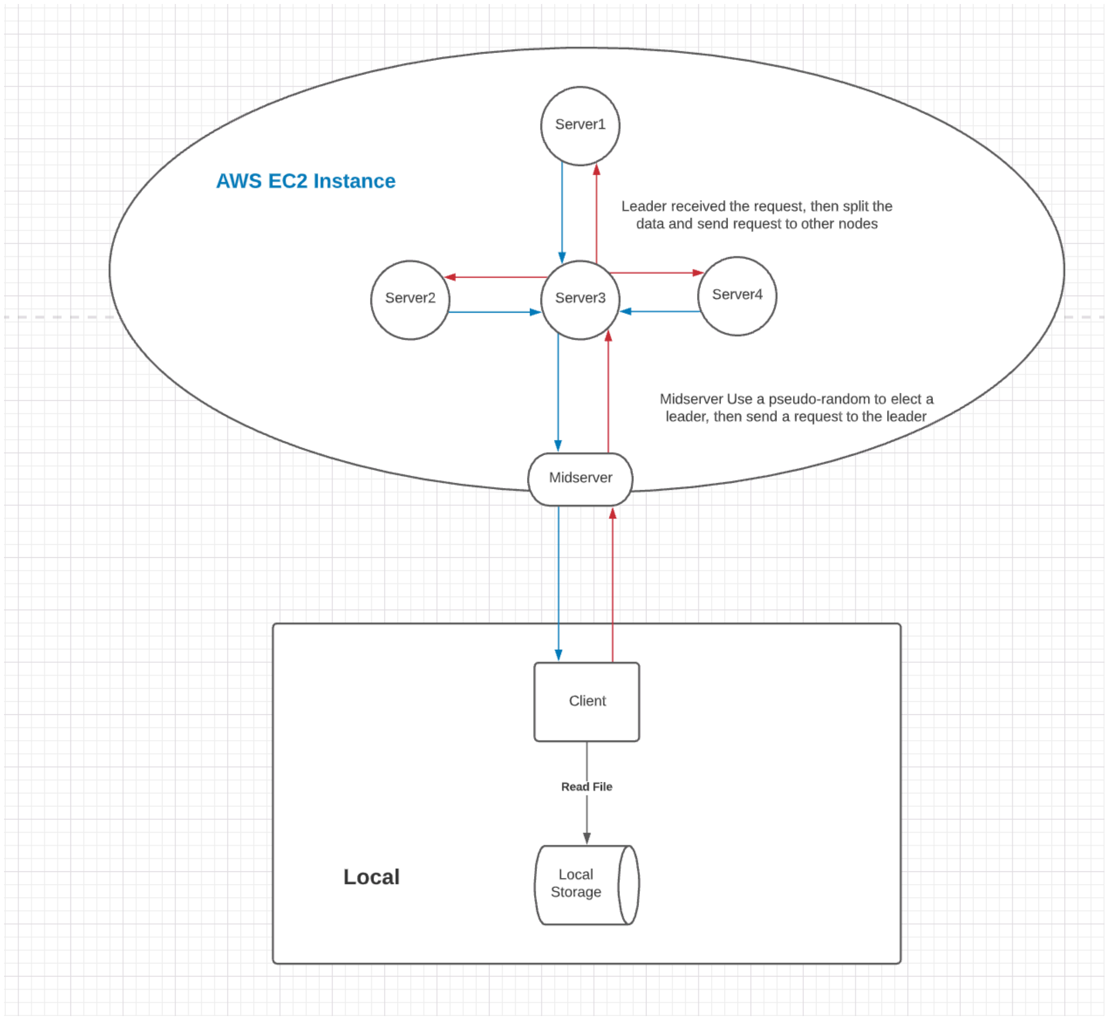
Cloud Service: AWS EC2

Protocol: RPC

Workflow and Data Flow:



Architecture:

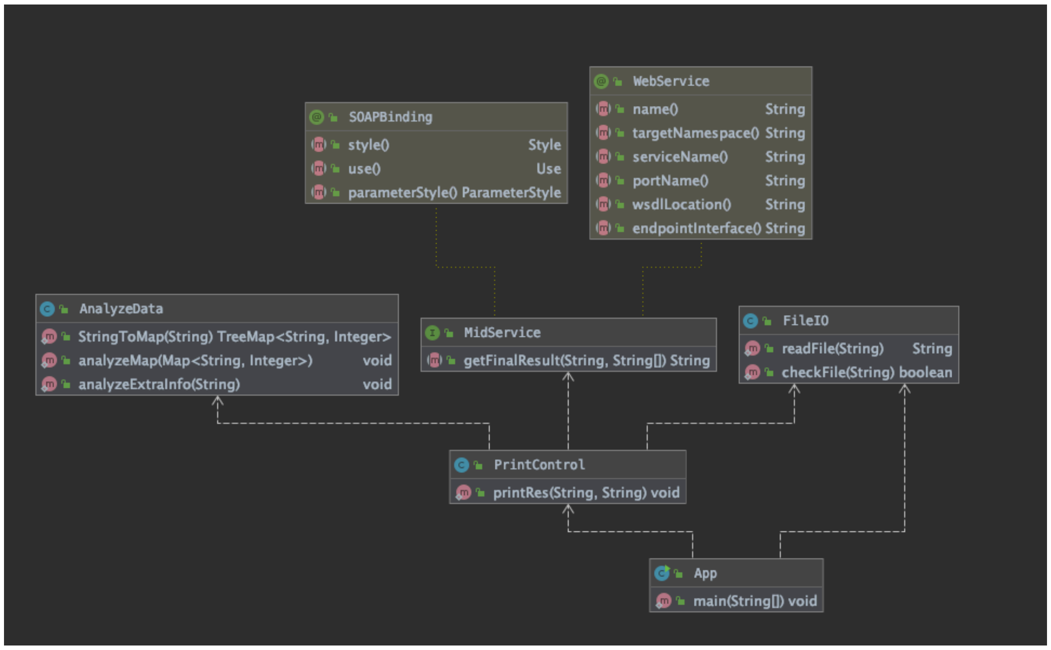


Server1
AWS EC2 Instance
Leader received the request, then split the data and send request to other nodes
Server2
Server3
Server4
Midserver Use a pseudo-random to elect a leader, then send a request to the leader
Midserver
Client
Read File
Local
Local Storage

1. In the cloud part, I use an EC2 instance to run my server program. I create a cluster that includes 4 nodes. Every node can communicate with each other. Every node can become a leader.
2. The mid server is the only node to provide web services to the external network. The mid-server just does one thing to elect a leader and send a request to the leader server. Mid-server doesn't process data.
3. When the leader receives the request, it will split data and distribute the sub-task to other nodes.
4. When other nodes receive the request from the leader, they will process the data, and return the result to the leader.
5. After all nodes respond, the leader will merge all results, and calculate the final result.
6. All servers and mid-server run in different docker containers.
7. Client read local file and send file content to mid-server.
8. After mid-server get result from leader server, mid-server send result to client.

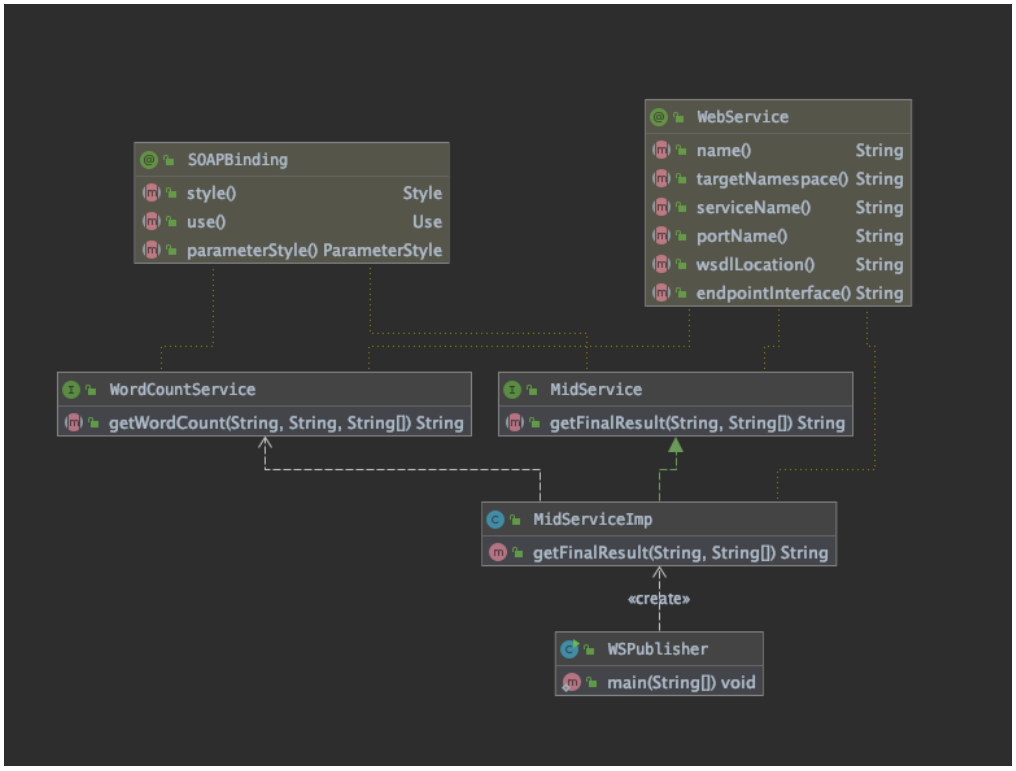
UML:

Client UML:

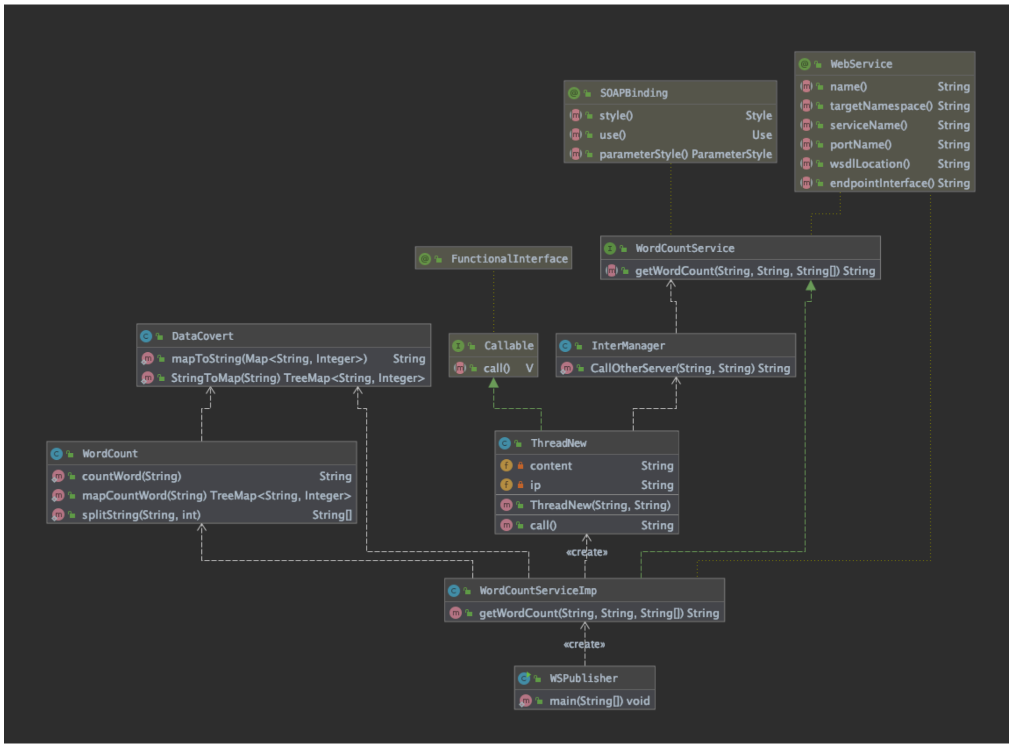


SOAPBinding
style() Style
use() Use
parameterStyle() ParameterStyle
WebService
name() String
targetNamespace() String
serviceName() String
portName() String
wsdlLocation() String
endpointInterface() String
AnalyzeData
StringToMap(String) TreeMap<String, Integer>
analyzeMap(Map<String, Integer>) void
analyzeExtraInfo(String) void
MidService
getFinalResult(String, String[]) String
FileIO
readFile(String) String
checkFile(String) boolean
PrintControl
printRes(String, String) void
App
main(String[]) void

Mid-Server UML:



Server UML:


WebService
name() String
targetNamespace() String
serviceName() String
portName() String
wsdlLocation() String
endpointInterface() String
SOAPBinding
style() Style
use() Use
parameterStyle() ParameterStyle
FunctionalInterface
WordCountService
getWordCount(String, String, String[]) String
DataCovert
mapToString(Map<String, Integer>) String
StringToMap(String) TreeMap<String, Integer>
Callable
call() V
InterManager
CallOtherServer(String, String) String
WordCount
countWord(String) String
mapCountWord(String) TreeMap<String, Integer>
splitString(String, int) String[]
ThreadNew
content String
ip String
ThreadNew(String, String)
call() String
«create»
WordCountServiceImp
getWordCount(String, String, String[]) String
«create»
WSPublisher
main(String[]) void

How to Implement:

Server:

1. Server has four functions: split data, call web services, data process, merge result. When the server is the leader, it should do all the functions. When the server is accepter, it just does the data process.
2. Use RPC to realize the communication between nodes. Each server will publish a web service.
3. All the servers run in a sub-net, so the external network must through the mid server to access them.
4. I use TreeMap to record the number of each word.
5. Webservice doesn't support transport Map object. So I convert TreeMap to String and return to the caller.
6. When Leader server get response, it should convert String to TreeMap, then merge TreeMap.
7. I use Multi-threaded concurrency to send requests to other nodes. In this way, I don't waste time to wait the node to response.

Mid Server:

1. The mid server uses pseudo-random to elect leader.
2. Mid server publishes a web service that is used for the client.
3. Mid Server also runs in the sub-net, but it maps to TCP port 3000. So, the client can access it from external network.

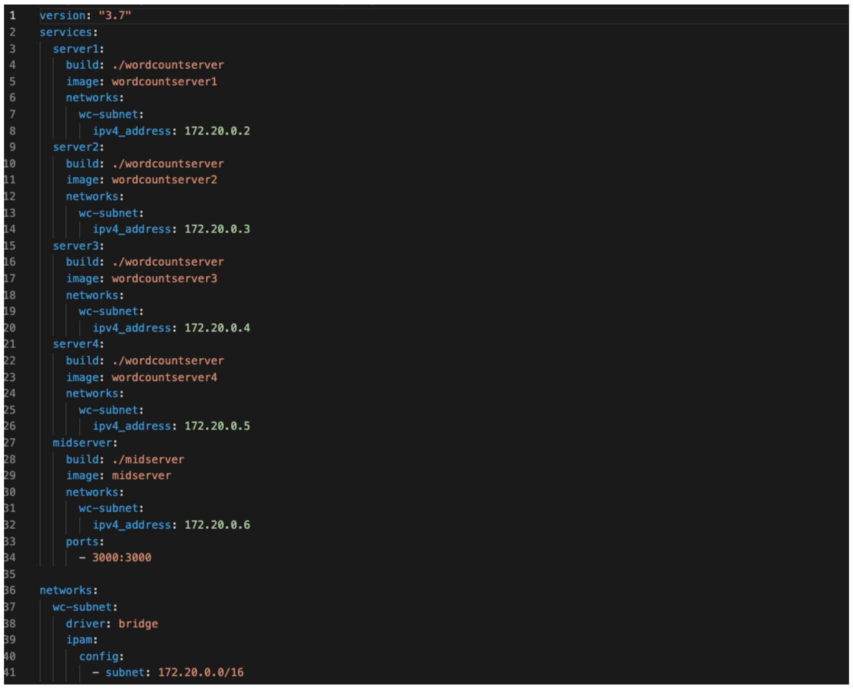
Client:

1. Client has tree functions: read file, call web service and analyze data.
2. My client application needs users to enter the hostname of the web service, I think it is a little complicated for users. But each time I start the ec2 instance that will be allocated a new IP address. Unless I don't stop my ec2 instance, I can't config the hostname for the client app.

3. I try to use AWS java JDK to get ec2 instance information in order to dynamic config hostname. But this way is too complicated for a client app.

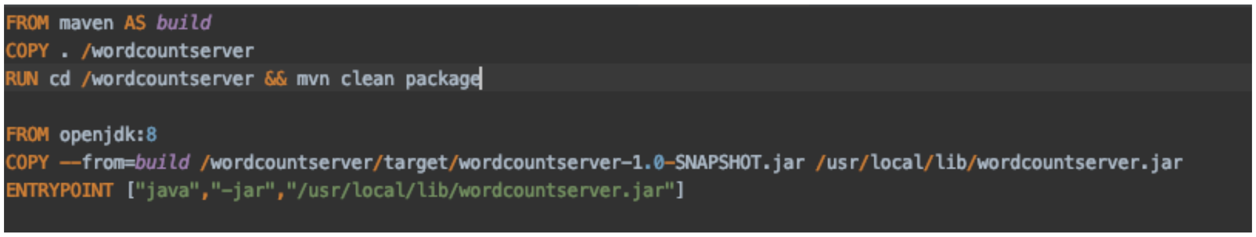
About Docker:

1. I use docker-compose.yml to deploy my application to container.



```yaml
version: "3.7"
services:
  server1:
    build: ./wordcountserver
    image: wordcountserver1
    networks:
      wc-subnet:
        ipv4_address: 172.20.0.2
  server2:
    build: ./wordcountserver
    image: wordcountserver2
    networks:
      wc-subnet:
        ipv4_address: 172.20.0.3
  server3:
    build: ./wordcountserver
    image: wordcountserver3
    networks:
      wc-subnet:
        ipv4_address: 172.20.0.4
  server4:
    build: ./wordcountserver
    image: wordcountserver4
    networks:
      wc-subnet:
        ipv4_address: 172.20.0.5
  midserver:
    build: ./midserver
    image: midserver
    networks:
      wc-subnet:
        ipv4_address: 172.20.0.6
    ports:
      - 3000:3000

networks:
  wc-subnet:
    driver: bridge
    ipam:
      config:
        - subnet: 172.20.0.0/16
```

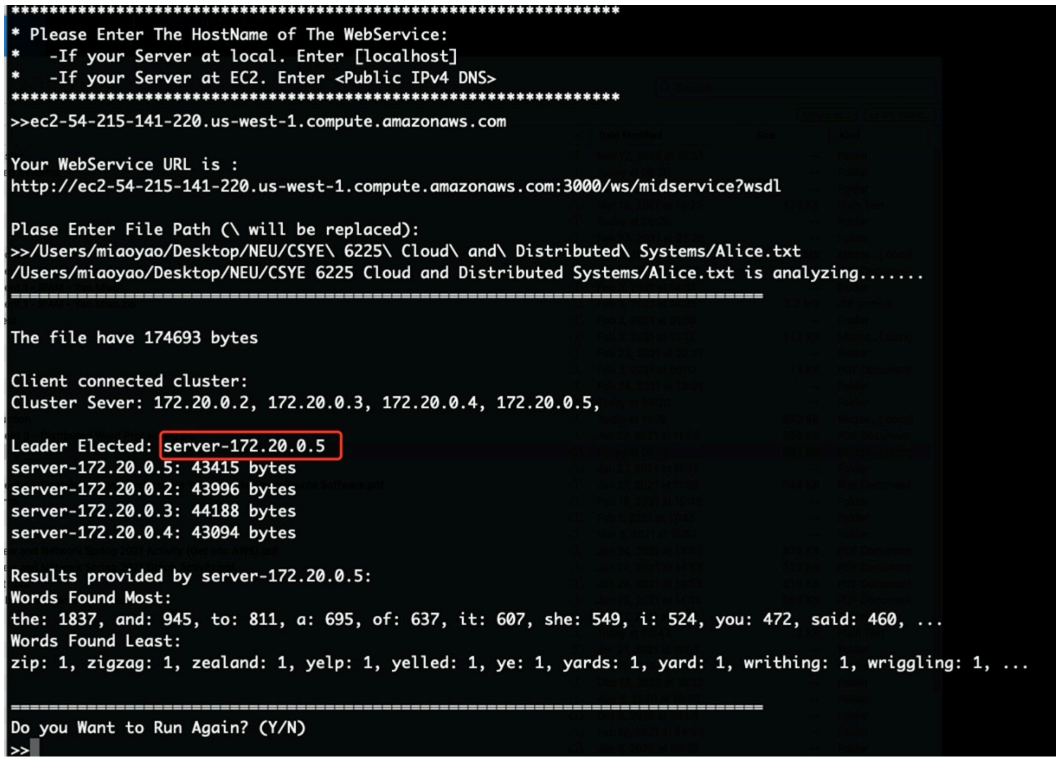
2. My Dockerfile



```dockerfile
FROM maven AS build
COPY . /wordcountserver
RUN cd /wordcountserver && mvn clean package

FROM openjdk:8
COPY --from=build /wordcountserver/target/wordcountserver-1.0-SNAPSHOT.jar /usr/local/lib/wordcountserver.jar
ENTRYPOINT ["java","-jar","/usr/local/lib/wordcountserver.jar"]
```

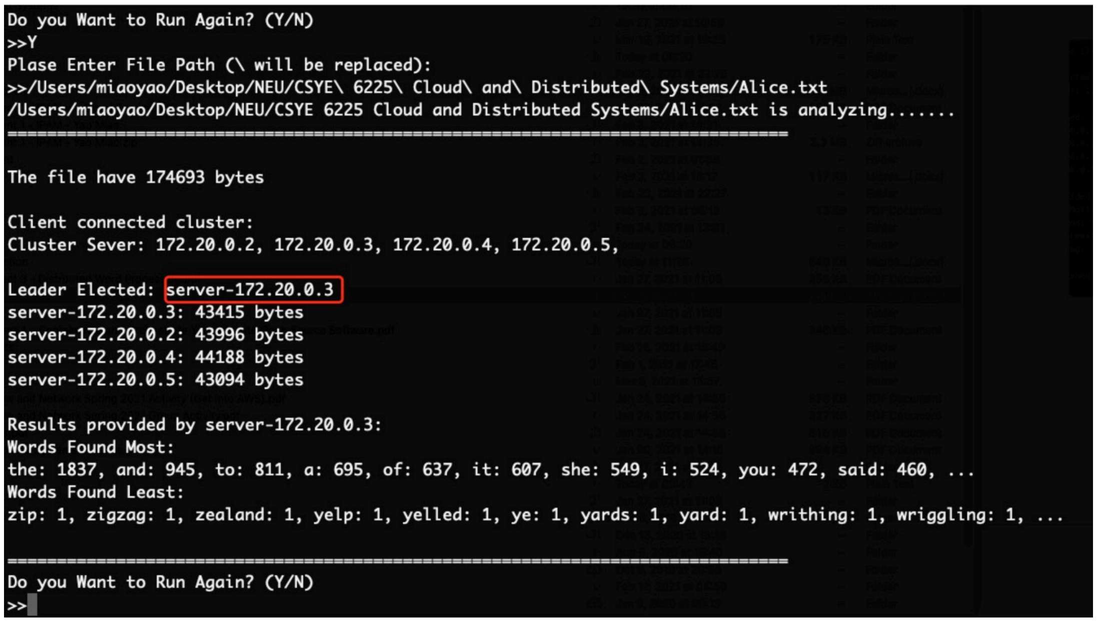
Running Result:

1. First time: Leader is 172.20.0.5



```
****************************************************************
* Please Enter The HostName of The WebService:
*   -If your Server at local. Enter [localhost]
*   -If your Server at EC2. Enter <Public IPv4 DNS>
****************************************************************
>>ec2-54-215-141-220.us-west-1.compute.amazonaws.com

Your WebService URL is :
http://ec2-54-215-141-220.us-west-1.compute.amazonaws.com:3000/ws/midservice?wsdl

Plase Enter File Path (\ will be replaced):
>>/Users/miaoyao/Desktop/NEU/CSYE\ 6225\ Cloud\ and\ Distributed\ Systems/Alice.txt
/Users/miaoyao/Desktop/NEU/CSYE 6225 Cloud and Distributed Systems/Alice.txt is analyzing.......
===============================================================================

The file have 174693 bytes

Client connected cluster:
Cluster Sever: 172.20.0.2, 172.20.0.3, 172.20.0.4, 172.20.0.5,

Leader Elected: server-172.20.0.5
server-172.20.0.5: 43415 bytes
server-172.20.0.2: 43996 bytes
server-172.20.0.3: 44188 bytes
server-172.20.0.4: 43094 bytes

Results provided by server-172.20.0.5:
Words Found Most:
the: 1837, and: 945, to: 811, a: 695, of: 637, it: 607, she: 549, i: 524, you: 472, said: 460, ...
Words Found Least:
zip: 1, zigzag: 1, zealand: 1, yelp: 1, yelled: 1, ye: 1, yards: 1, yard: 1, writhing: 1, wriggling: 1, ...

===============================================================================
Do you Want to Run Again? (Y/N)
>>
```

2. Second time: Leader is 172.20.0.3



```
Do you Want to Run Again? (Y/N)
>>Y
Plase Enter File Path (\ will be replaced):
>>/Users/miaoyao/Desktop/NEU/CSYE\ 6225\ Cloud\ and\ Distributed\ Systems/Alice.txt
/Users/miaoyao/Desktop/NEU/CSYE 6225 Cloud and Distributed Systems/Alice.txt is analyzing.......
===============================================================================

The file have 174693 bytes

Client connected cluster:
Cluster Sever: 172.20.0.2, 172.20.0.3, 172.20.0.4, 172.20.0.5,

Leader Elected: server-172.20.0.3
server-172.20.0.3: 43415 bytes
server-172.20.0.2: 43996 bytes
server-172.20.0.4: 44188 bytes
server-172.20.0.5: 43094 bytes

Results provided by server-172.20.0.3:
Words Found Most:
the: 1837, and: 945, to: 811, a: 695, of: 637, it: 607, she: 549, i: 524, you: 472, said: 460, ...
Words Found Least:
zip: 1, zigzag: 1, zealand: 1, yelp: 1, yelled: 1, ye: 1, yards: 1, yard: 1, writhing: 1, wriggling: 1, ...

===============================================================================
Do you Want to Run Again? (Y/N)
>>
```