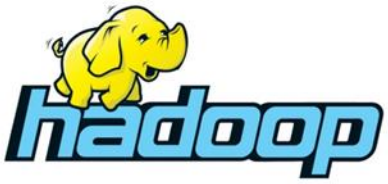
تمرین دوم درس تحلیل ها و سیستم های داده های حجیم – استادگرامی جناب آقای دکتر خواسته

نگاشت کاهش در هدوپ



دستیاران: آذرکسب- زکی زاده

ترم اول سال تحصیلی ۰۲-۱۴۰۱

مقدمه:

در این تمرین ما می خواهیم به صورت عملی با Hadoop به عنوان یک پروژه متن باز برای محاسبات توزیع شده، مقیاس پذیر و ذخیره داده ها آشنایی پیدا کنیم. هدوپ امکان ذخیره و تجزیه و تحلیل، کلان داده را با زمان و هزینه قابل قبول فراهم می کند. برای استفاده از پردازش موازی در هدوپ ابتدا باید پرس و جو را با مدل برنامه نویسی map-reduce بیان کنیم. وظیفه اصلی برنامه نویس در هدوپ نوشتن دو تابع mapو reduce است. در ادامه به اهمیت طراحی جفت-کلید مقدار برای مسائل مختلف در مدل نگاشت-کاهش آشنا خواهیم شد. همچنین نحوه نوشتن تابع reduce ، map و combine برای اجرا درهدوپ یادگرفته می شود. به عنوان راهنمایی به طور مثال چنانچه محیط کاری تان لینوکس debian 11 است. برای آماده سازی سیستم، ابتدا مراحل زیر می بایست انجام شود:

۱- هدوپ رو دانلود می کنیم:

```
sudo wget https://dlcdn.apache.org/hadoop/common/stable2/hadoop-2.10.1.tar.gz
```

۲- فایل دانلود شده رو اکسترکت می کنیم:

```
sudo tar -xvzf hadoop-2.10.1.tar.gz
```

۳- در فایل اکسترکت شده به مسیر etc/hadoop می رویم و فایل hadoop-env.sh رو ادیت می کنیم و در خط JAVA_HOMEآدرس $JAVA_HOME سیستم خودمون را قرار می دهیم.

۴- حالا از مسیر bin در فایل اکسترکت شده دستور hadoop رو اجرا می کنید تا ببینیم اوکی است یا نه؟

نتیجه اجرای دستور hadoop جهت تست کردن نصب درست برنامه در شکل ۱ نشان داده شده است.

```
mostafa@amiri:/media/mostafa/CE/BD/exercise2$ ../hadoop/hadoop/hadoop-2.10.1/bin/hadoop
Usage: hadoop [--config confdir] [COMMAND | CLASSNAME]
  CLASSNAME            run the class named CLASSNAME
 or
  where COMMAND is one of:
  fs                   run a generic filesystem user client
  version              print the version
  jar <jar>            run a jar file
                       note: please use "yarn jar" to launch
                             YARN applications, not this command.
  checknative [-a|-h]  check native hadoop and compression libraries availability
  distcp <srcurl> <desturl> copy file or directories recursively
  archive -archiveName NAME -p <parent path> <src>* <dest> create a hadoop archive
  classpath            prints the class path needed to get the
                       Hadoop jar and the required libraries
  credential           interact with credential providers
  daemonlog            get/set the log level for each daemon
  trace                view and modify Hadoop tracing settings

Most commands print help when invoked w/o parameters.
```

شکل ۱: نتیجه اجرای دستور hadoop جهت تست کردن نصب درست برنامه

در ادامه باید توابع map و reduce را بنویسیم ساختار کلی به این صورت است که یک کلاس تعریف می کنیم و در آن دو کلاس که یکی کلاس

org.apache.hadoop.mapreduce.Mapper

و دیگری کلاس

org.apache.hadoop.mapreduce.Reducer

را پیاده‌سازی می کنند تعریف می کنیم.

کلاس اول یک تابع map و کلاس دوم یک تابع reduce دارد. بنابراین ساختار کلی به این صورت شکل ۲ خواهد بود:

```
public static class MyMapper
extends Mapper<Object, Text, Text, Text>{

public void map(Object key, Text value, Context context
) throws IOException, InterruptedException {
//some map action
context.write(key, value);
}
}
}

public static class MyReducer
extends Reducer<Text,Text,Text,Text> {
//some reduce action
context.write(key, result);
}

}

public static void main(String[] args) throws Exception {
Configuration conf = new Configuration();
Job job = Job.getInstance(conf, "example");
job.setJarByClass(Example.class);
job.setMapperClass(MyMapper.class);
job.setCombinerClass(MyReducer.class);
job.setReducerClass(MyReducer.class);
job.setOutputKeyClass(Text.class);
job.setOutputValueClass(Text.class);
FileInputFormat.addInputPath(job, new Path(args[0]));
FileOutputFormat.setOutputPath(job, new Path(args[1]));
System.exit(job.waitForCompletion(true) ? 0 : 1);
}
}
```

شکل ۲: ساختار کلی توابع مپ و ردیوس

ضمنا یک تابع main هم داریم که تنظیمات مربوط به فریمورک را در آن انجام می دهیم و تابع مپ و ردیوس و در صورت نیاز کامباینر را به آن معرفی می کنیم و بعد از نوشتن این کلاس توسط دستور زیر آن را کامپایل می کنیم:

```
hadoop com.sun.tools.javac.Main Example.java
```

و بعد از آن کلاس ها را در یک فایل jar توسط دستور زیر قرار می دهیم:

```
jar cf exmpl.jar Exampe*.class
```

در ادامه فایل یا فایلهای ورودی را در یک فولدر مثلا به نام input قرار می دهیم و با دستور زیر برنامه را اجرا می کنیم:

```
hadoop jar exmpl.jar Example input output
```

و نهایتا نتایج در دایرکتوری output قابل مشاهده خواهد بود.

نکات:

- **ضمن توجه و تاکید بر نکات عنوان شده در تمرین اول به خصوص موضوع کپی!، به اطلاع دوستان می رسانیم که ارائه این تکلیف نیز به صورت گروههای ۲ نفره امکان پذیر می باشد.**
- **ارائه گزارش کامل و مصور به همراه توضیح مراحل انجام کار، بسیار حائز اهمیت است و در مقدار نمره دهی بسیار تاثیر گذار است.**
- **تثبیت نمره تکلیف، از طریق ارائه توضیحات به صورت حضوری توسط هر دو عضو گروه، انجام می شود که زمان آن متعاقبا اعلام خواهد شد.**
- **دوستان، در ادامه لطفا با استفاده از مجموعه داده تکلیف اول، و روش نگاشت-کاهش به سوالات زیر پاسخ دهید.**

مجموعه سوالات اول:

۱) تعداد سفارش های هر پیتزا به تفکیک نوع و اندازه را بدست آورید و به صورت نزولی مرتب کنید.

۲) مجموع کل قیمت پیتزا به تفکیک نوع و اندازه را بدست آورید و به صورت صعودی مرتب کنید.

۳) نوع و سایز پیتزا با بیشترین قیمت و کمترین کیفیت را پیدا کنید.

٤) در کدام ماه (بدون در نظر گرفتن سال) بیشترین تعداد سفارش را داشته ایم؟

٥) میانگین قیمت پیتزا در طول هر سال را محاسبه کنید.

مجموعه سوالات دوم:

۱) کاربرانی که به ٤ نوع ورزش علاقه مند هستند را بیابید و به صورت صعودی مرتب کنید.

۲) کمترین و بیشترین حروفی که در نام ها شرکت کنندگان وجود دارند را بیابید.

۳) میانگین طول نام های افراد را بیابید.

٤) محبوب ترین ورزش کدام است؟

٥) در چه سالی بیشترین تعداد ثبت نام را داشته ایم؟

با تشکر و آرزوی سلامتی برای تمام دوستان و عزیزان