将Flink DataSet中的数据写入到ElasticSearch(低级篇)

Flink内置提供了将DataStream中的数据写入到ElasticSearch中的Connector(flink-connector-elasticsearch2_2.10)，但是并没有提供将DateSet的数据写入到ElasticSearch。本文介绍如何通过自定义OutputFormat将Flink DateSet里面的数据写入到ElasticSearch。

如果需要将DateSet中的数据写入到外部存储系统（比如HDFS），我们可以通过writeAsText、writeAsCsv、write等内置的方法；这里我们需要将数据写入到ElasticSearch中，然而Flink内置并没有提供相应的实现类。而且我们知道如果需要将数据写到外部系统，我们可以定义一个我们自己的OutputFormat类，内置的writeAsText、writeAsCsv、write分别对应于TextOutputFormat、ScalaCsvOutputFormat以及FileOutputFormat其最终都是实现OutputFormat类，所以如果我们需要将数据写入到ElasticSearch，我们可以定义一个ElasticSearchOutputFormat类，实现如下：

```
package com.iteblog.es

import java.net.InetAddress

import org.apache.flink.api.common.io.OutputFormat
import org.apache.flink.configuration.Configuration
import org.elasticsearch.client.Requests
import org.elasticsearch.client.transport.TransportClient
import org.elasticsearch.common.settings.Settings
import org.elasticsearch.common.transport.InetSocketTransportAddress

import scala.collection.JavaConversions

//////////////////////////////////////////////////////////////////////
 User: 过往记忆
 Date: 2016-10-11
 Time: 23:35
 bolg: https://www.iteblog.com
 本文地址：https://www.iteblog.com/archives/1825
 过往记忆博客，专注于hadoop、hive、spark、shark、flume的技术博客，大量的干货
 过往记忆博客微信公共帐号：iteblog_hadoop
//////////////////////////////////////////////////////////////////////
class ElasticSearchOutputFormat extends OutputFormat[String] {
  var client: TransportClient = _

  override def configure(configuration: Configuration): Unit = {
    val settings = Settings.settingsBuilder().put("cluster.name", "iteblog").build()
    val nodes = "www.iteblog.com"
    val transportAddress = nodes.split(",").map(node =>
```

```
        new InetSocketTransportAddress(InetAddress.getByName(node), 9003))
    client = TransportClient.builder().settings(settings).build()
        .addTransportAddresses(transportAddress: _*)
  }

  override def close(): Unit = client.close()

  override def writeRecord(element: String): Unit = {
    val indexRequest = Requests.indexRequest.index("iteblog").`type`("iteblog").source(elemen
t)
    client.index(indexRequest).actionGet()
  }

  override def open(taskNumber: Int, numTasks: Int): Unit = {
    //super.open(taskNumber, numTasks)
  }
}
```

我们的ElasticSearchOutputFormat类继承自OutputFormat，然后实现configure方法，在其中我们主要是初始化TransportClient。数据写入到ElasticSearch主要是通过writeRecord函数实现，这里我们的index和type都是iteblog，element是json格式的数据，当然你也可以写入一个Map，这里就不介绍；close方法将在数据写完之后关闭TransportClient。现在我们来使用这个ElasticSearchOutputFormat，如下：

```
val path = "data.txt"

val env = ExecutionEnvironment.getExecutionEnvironment

val data = env.readTextFile(path).filter(_.nonEmpty).map { line =>
  val Array(registerTime, mobileEncrypt, mobile, uid, userName, lastUpdateTime) = line.split("
₩₩t")
  val millis = DateTime.parse(lastUpdateTime, DateTimeFormat.forPattern("yyyy-MM-
dd HH:mm:ss")).getMillis
  val map = Map("regtime" -> registerTime, "uid" -> uid.toInt, "mobile_encrypt" -> mobileEncry
pt,
    "execute_time" -> millis, "mobile" -> mobile, "username" -> userName)
  scala.util.parsing.json.JSONObject(map).toString()
}

data.output(new ElasticSearchOutputFormat)
```

data.txt文件的数据格式如下：

1303692738 186iGAB3350 18621603350 2000011 lyla422 2016-02-11 23:38:12
1303692770 189c9Xp7138 18999137138 2000012 oylan21 2015-05-20 10:56:30
1303692773 153Md_L8077 15352028077 2000013 m028077 2015-05-20 10:56:30
1303692774 138Pb=w7180 13808137180 2000014 711065 2016-08-29 23:04:48
1303692776 138G1D07802 13851487802 2000015 alexqiao 2016-06-17 14:31:36
1303692791 132mtXg9996 13267699996 2000016 linzhuoyi 2015-05-20 10:56:30
1303692806 1382uHm7699 13810657699 2000017 liyanjie86 2015-12-12 15:41:45

我们测试了1,172,235条数据，使用上面ElasticSearchOutputFormat全部写完使用了40多分钟！原因是ElasticSearchOutputFormat每次只向ElasticSearch请求一条记录，这效率肯定很低，所以生产环境下是不会使用这中方法的。明天我将介绍一种更高效的方法将Flink DateSet数据写入到ElasticSearch，敬请关注。