Spark 2.0介绍：Dataset介绍和使用

《Spark 2.0技术预览：更容易、更快速、更智能》文章中简单地介绍了Spark 2.0带来的新技术等。Spark 2.0是Apache Spark的下一个主要版本。此版本在架构抽象、API以及平台的类库方面带来了很大的变化，为该框架明年的发展方向奠定了方向，所以了解Spark 2.0的一些特性对我们能够使用它有着非常重要的作用。本博客将对Spark 2.0进行一序列的介绍（参见Spark 2.0分类），欢迎关注。

Dataset介绍

Dataset是从Spark 1.6开始引入的一个新的抽象，当时还是除于alpha版本；然而在Spark 2.0，它已经变成了稳定版了。下面是DataSet的官方定义：

> A Dataset is a strongly typed collection of domain-specific objects that can be transformed
> in parallel using functional or relational operations. Each Dataset also has an untyped view
> called a DataFrame, which is a Dataset of Row.

Dataset是特定域对象中的强类型集合，它可以使用函数或者相关操作并行地进行转换等操作。每个Dataset都有一个称为DataFrame的非类型化的视图，这个视图是行的数据集。上面的第一看起来和RDD的定义类似，RDD的定义如下：

> RDD represents an immutable,partitioned collection of elements that can be operated on in parallel

RDD也是可以并行化的操作，DataSet和RDD主要的区别是：DataSet是特定域的对象集合；然而RDD是任何对象的集合。DataSet的API总是强类型的；而且可以利用这些模式进行优化，然而RDD却不行。

Dataset的定义中还提到了DataFrame，DataFrame是特殊的Dataset，它在编译时不会对模式进行检
测。在未来版本的
Spark，Dataset将会替代RDD成为我

们开发编程使用的API（注意，RDD并不是会被取消，而是会作为底层的API提供给用户使用）。

上面简单地介绍了Dataset相关的定义，下面让我们来看看如何以编程的角度来使用它。

Dataset Wordcount实例

为了简单起见，我将介绍如何使用DataSet编写WordCount计算程序。

第一步、创建SparkSession

正如我们在《Spark 2.0介绍：SparkSession创建和使用相关API》
中提到的，我们在这里将使用SparkSession作为程序的切入点，并使用它来创建出Dataset：

```
val sparkSession = SparkSession.builder.
    master("local")
    .appName("example")
    .getOrCreate()
```

第二步、读取数据并将它转换成Dataset

我们可以使用read.text
API来读取数据，正如RDD版提供的textFile，as[String]可以为dataset提供相关的模式，如下：

```
import sparkSession.implicits._
val data = sparkSession.read.text("src/main/resources/data.txt").as[String]
```

上面data对象的类型是DataSet[String]，我们需要引入sparkSession.implicits._。

第三步、分割单词并且对单词进行分组

Dataset提供的API和RDD提供的非常类似，所以我们也可以在DataSet对象上使用map,
groupByKey相关的API，如下：

```
val words = data.flatMap(value => value.split("₩₩s+"))
val groupedWords = words.groupByKey(_.toLowerCase)
```

有得同学可能注意到，我们并没有创建出一个key/value键值对，因为DataSet是工作在行级别的抽象，每个值将被看作是带有多列的行数据，而且每个值都可以看作是group的key，正如关系型数据库的group。

第四步、计数

一旦我们有了分组好的数据，我们可以使用count方法对每个单词进行计数，正如在RDD上使用reduceByKey：

```
val counts = groupedWords.count()
```

第五步、打印结果

正如RDD一样，上面的操作都是懒执行的，所以我们需要调用action操作来触发上面的计算。在dataset API中，show函数就是action操作，它会输出前20个结果；如果你需要全部的结果，你可以使用collect操作：

```
counts.show()
```

完整的代码

```
package com.iteblog.spark

import org.apache.spark.sql.SparkSession

/**
  * Created by http://www.iteblog.com
  */
object DataSetWordCount {

  def main(args: Array[String]) {

    val sparkSession = SparkSession.builder.
      master("local")
      .appName("example")
      .getOrCreate()

    import sparkSession.implicits._
    val data = sparkSession.read.text("src/main/resources/data.txt").as[String]
```

```
    val words = data.flatMap(value => value.split("₩₩s+"))

    val groupedWords = words.groupByKey(_.toLowerCase)

    val counts = groupedWords.count()

    counts.show()

  }

}
```