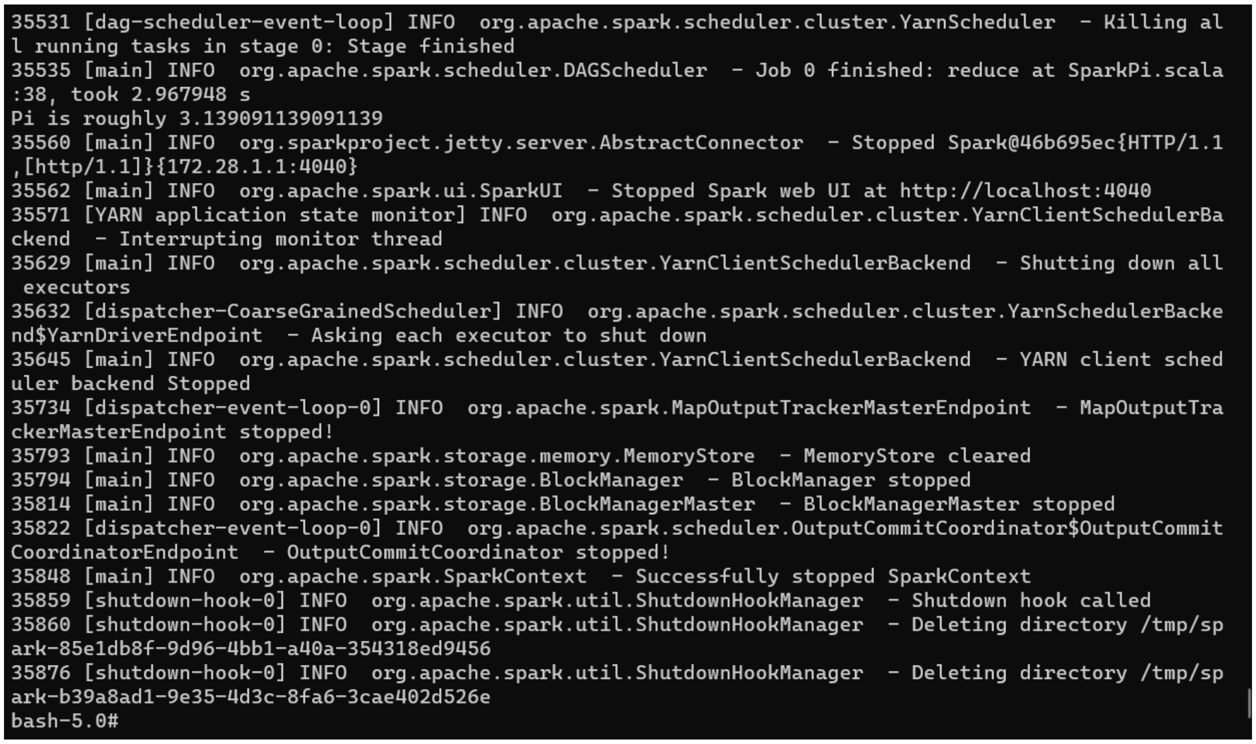
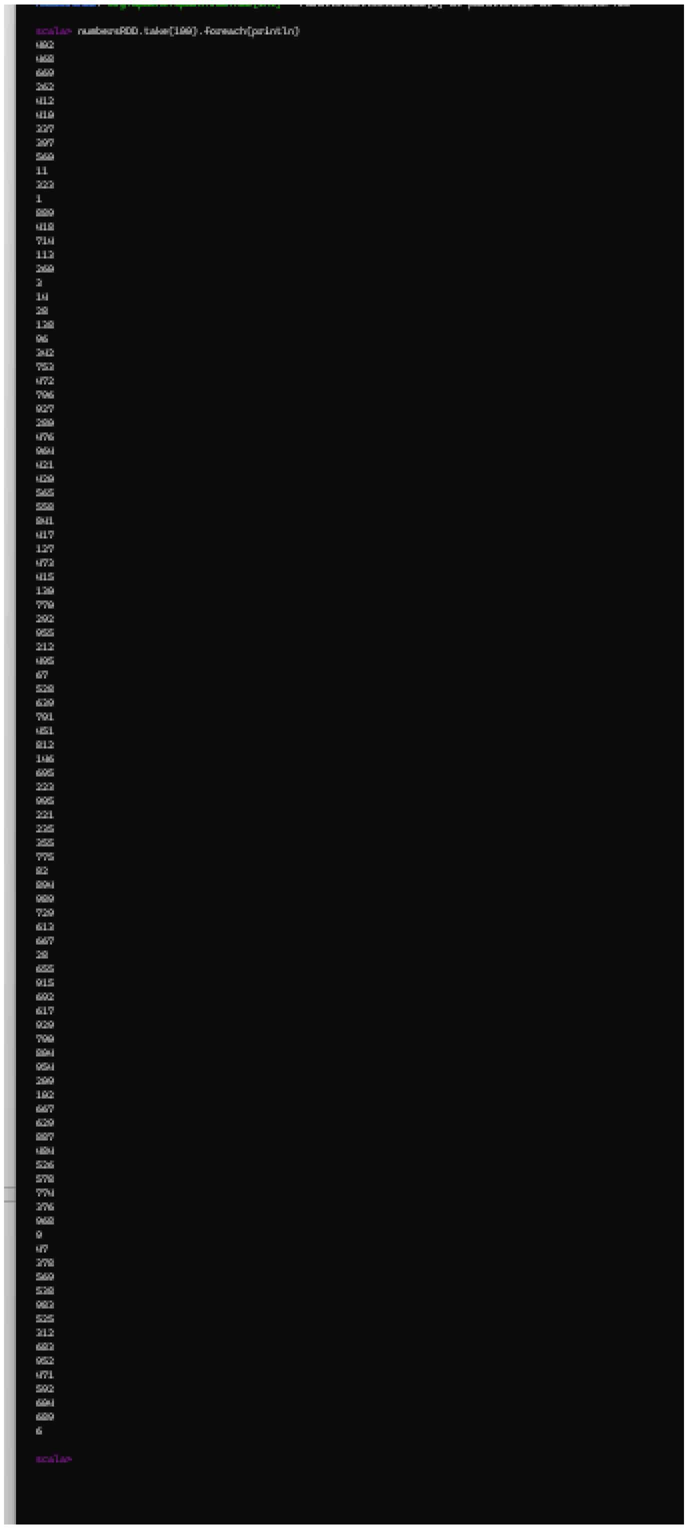
Screenshot of spark Pi output



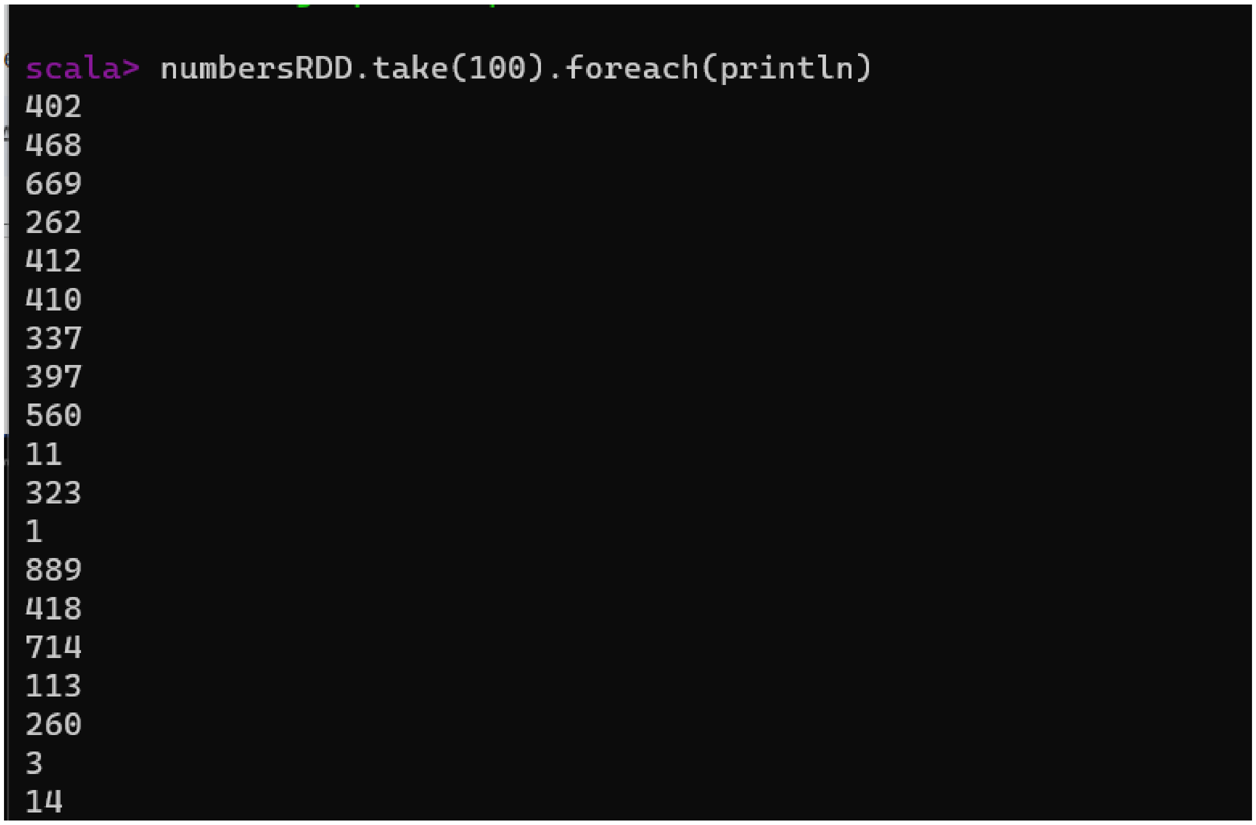
```
35531 [dag-scheduler-event-loop] INFO  org.apache.spark.scheduler.cluster.YarnScheduler  - Killing al
l running tasks in stage 0: Stage finished
35535 [main] INFO  org.apache.spark.scheduler.DAGScheduler  - Job 0 finished: reduce at SparkPi.scala
:38, took 2.967948 s
Pi is roughly 3.139091139091139
35560 [main] INFO  org.sparkproject.jetty.server.AbstractConnector  - Stopped Spark@46b695ec{HTTP/1.1
,[http/1.1]}{172.28.1.1:4040}
35562 [main] INFO  org.apache.spark.ui.SparkUI  - Stopped Spark web UI at http://localhost:4040
35571 [YARN application state monitor] INFO  org.apache.spark.scheduler.cluster.YarnClientSchedulerBa
ckend  - Interrupting monitor thread
35629 [main] INFO  org.apache.spark.scheduler.cluster.YarnClientSchedulerBackend  - Shutting down all
 executors
35632 [dispatcher-CoarseGrainedScheduler] INFO  org.apache.spark.scheduler.cluster.YarnSchedulerBacke
nd$YarnDriverEndpoint  - Asking each executor to shut down
35645 [main] INFO  org.apache.spark.scheduler.cluster.YarnClientSchedulerBackend  - YARN client sched
uler backend Stopped
35734 [dispatcher-event-loop-0] INFO  org.apache.spark.MapOutputTrackerMasterEndpoint  - MapOutputTra
ckerMasterEndpoint stopped!
35793 [main] INFO  org.apache.spark.storage.memory.MemoryStore  - MemoryStore cleared
35794 [main] INFO  org.apache.spark.storage.BlockManager  - BlockManager stopped
35814 [main] INFO  org.apache.spark.storage.BlockManagerMaster  - BlockManagerMaster stopped
35822 [dispatcher-event-loop-0] INFO  org.apache.spark.scheduler.OutputCommitCoordinator$OutputCommit
CoordinatorEndpoint  - OutputCommitCoordinator stopped!
35848 [main] INFO  org.apache.spark.SparkContext  - Successfully stopped SparkContext
35859 [shutdown-hook-0] INFO  org.apache.spark.util.ShutdownHookManager  - Shutdown hook called
35860 [shutdown-hook-0] INFO  org.apache.spark.util.ShutdownHookManager  - Deleting directory /tmp/sp
ark-85e1db8f-9d96-4bb1-a40a-354318ed9456
35876 [shutdown-hook-0] INFO  org.apache.spark.util.ShutdownHookManager  - Deleting directory /tmp/sp
ark-b39a8ad1-9e35-4d3c-8fa6-3cae402d526e
bash-5.0#
```

Screenshot of the 100 random generated numbers:



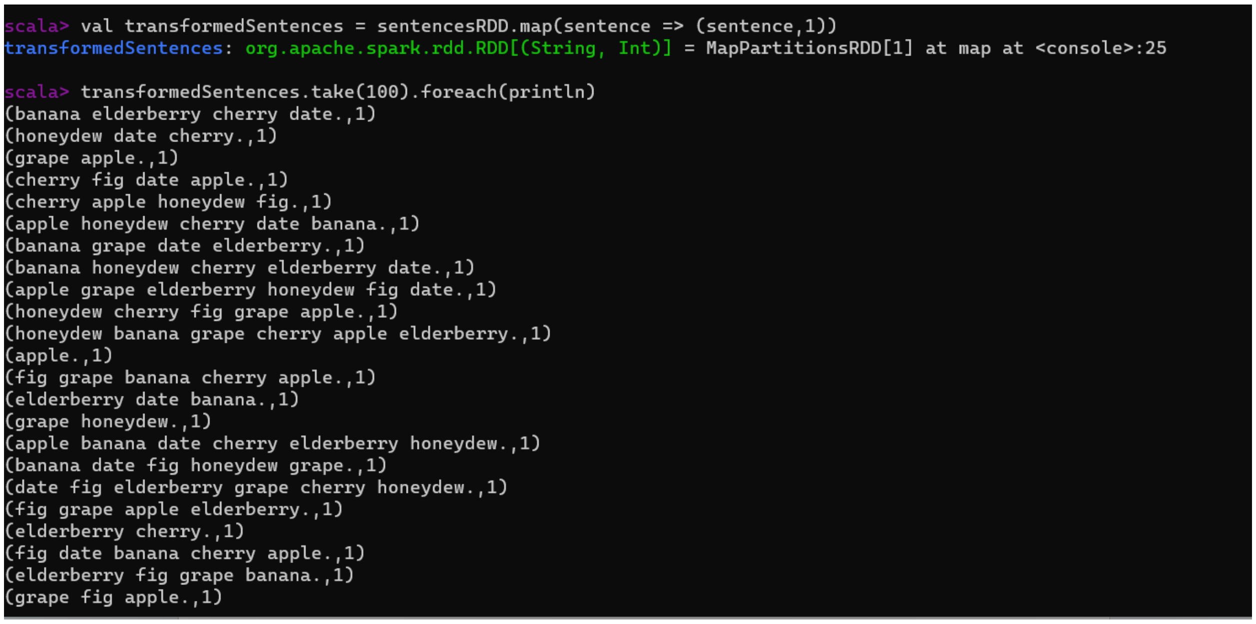
```
[illegible]

scala> numbersRDD.take(100).foreach(println)
482
488
669
262
412
419
337
397
569
11
323
1
889
418
714
112
269
3
14
28
138
96
342
753
472
796
927
389
476
964
421
439
565
558
841
417
127
473
415
139
779
392
955
212
495
67
528
639
791
451
812
146
695
323
995
221
235
355
775
82
894
989
729
613
667
28
655
915
692
617
939
799
894
954
399
192
667
639
897
494
536
578
774
376
968
9
47
378
569
538
983
525
312
683
952
471
592
694
689
6

scala>
```

This is an up close look at the command and numbers



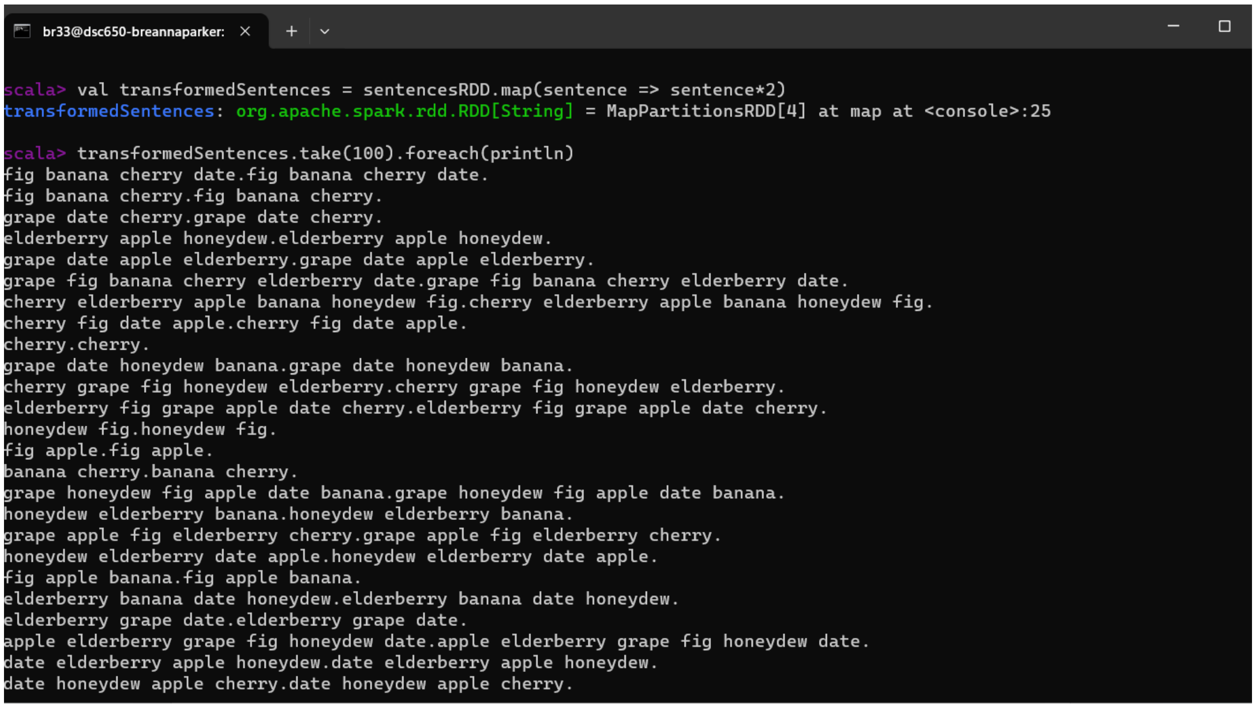
```
scala> numbersRDD.take(100).foreach(println)
402
468
669
262
412
410
337
397
560
11
323
1
889
418
714
113
260
3
14
```

Generated transformed sentence
– here I just printed the sentence, and added a one at the end of it. This was just that a command that I found and wanted to see if it would really just print whatever I wanted to at the end of each sentence. This could be useful if you needed to add something before or after a generated sentence.



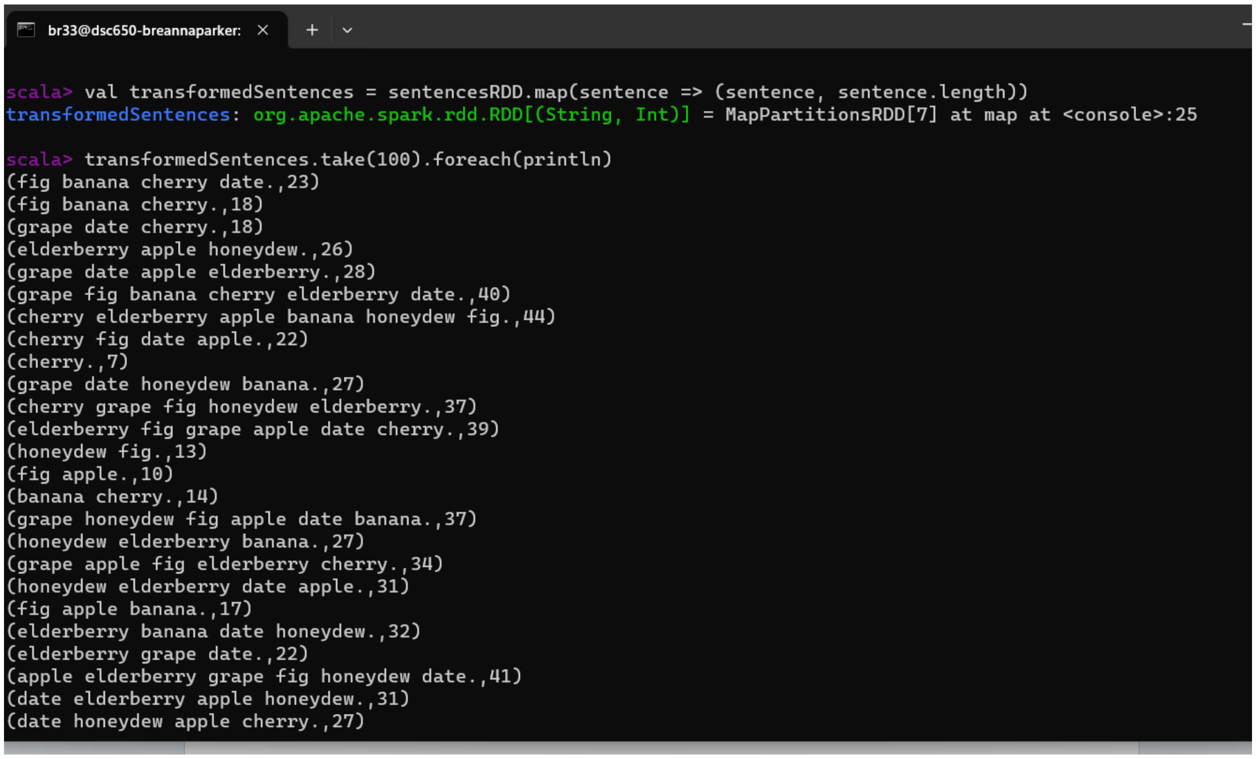
```
scala> val transformedSentences = sentencesRDD.map(sentence => (sentence,1))
transformedSentences: org.apache.spark.rdd.RDD[(String, Int)] = MapPartitionsRDD[1] at map at <console>:25

scala> transformedSentences.take(100).foreach(println)
(banana elderberry cherry date.,1)
(honeydew date cherry.,1)
(grape apple.,1)
(cherry fig date apple.,1)
(cherry apple honeydew fig.,1)
(apple honeydew cherry date banana.,1)
(banana grape date elderberry.,1)
(banana honeydew cherry elderberry date.,1)
(apple grape elderberry honeydew fig date.,1)
(honeydew cherry fig grape apple.,1)
(honeydew banana grape cherry apple elderberry.,1)
(apple.,1)
(fig grape banana cherry apple.,1)
(elderberry date banana.,1)
(grape honeydew.,1)
(apple banana date cherry elderberry honeydew.,1)
(banana date fig honeydew grape.,1)
(date fig elderberry grape cherry honeydew.,1)
(fig grape apple elderberry.,1)
(elderberry cherry.,1)
(fig date banana cherry apple.,1)
(elderberry fig grape banana.,1)
(grape fig apple.,1)
```

– here I just multiplied the sentences times two. Again, just trying to see what it would let me do. Since these weren't numbers, I thought I would see how it does this with words. It just takes the full sentence and runs it right after each other. Something else that could be useful, depending on the situation.



```
scala> val transformedSentences = sentencesRDD.map(sentence => sentence*2)
transformedSentences: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[4] at map at <console>:25

scala> transformedSentences.take(100).foreach(println)
fig banana cherry date.fig banana cherry date.
fig banana cherry.fig banana cherry.
grape date cherry.grape date cherry.
elderberry apple honeydew.elderberry apple honeydew.
grape date apple elderberry.grape date apple elderberry.
grape fig banana cherry elderberry date.grape fig banana cherry elderberry date.
cherry elderberry apple banana honeydew fig.cherry elderberry apple banana honeydew fig.
cherry fig date apple.cherry fig date apple.
cherry.cherry.
grape date honeydew banana.grape date honeydew banana.
cherry grape fig honeydew elderberry.cherry grape fig honeydew elderberry.
elderberry fig grape apple date cherry.elderberry fig grape apple date cherry.
honeydew fig.honeydew fig.
fig apple.fig apple.
banana cherry.banana cherry.
grape honeydew fig apple date banana.grape honeydew fig apple date banana.
honeydew elderberry banana.honeydew elderberry banana.
grape apple fig elderberry cherry.grape apple fig elderberry cherry.
honeydew elderberry date apple.honeydew elderberry date apple.
fig apple banana.fig apple banana.
elderberry banana date honeydew.elderberry banana date honeydew.
elderberry grape date.elderberry grape date.
apple elderberry grape fig honeydew date.apple elderberry grape fig honeydew date.
date elderberry apple honeydew.date elderberry apple honeydew.
date honeydew apple cherry.date honeydew apple cherry.
```

– here I added in a count of the length of the sentences at the end. So, I believe it counted the total number of characters in each sentence and then printed it at the end of each sentence.


br33@dsc650-breannaparker:

```
scala> val transformedSentences = sentencesRDD.map(sentence => (sentence, sentence.length))
transformedSentences: org.apache.spark.rdd.RDD[(String, Int)] = MapPartitionsRDD[7] at map at <console>:25

scala> transformedSentences.take(100).foreach(println)
(fig banana cherry date.,23)
(fig banana cherry.,18)
(grape date cherry.,18)
(elderberry apple honeydew.,26)
(grape date apple elderberry.,28)
(grape fig banana cherry elderberry date.,40)
(cherry elderberry apple banana honeydew fig.,44)
(cherry fig date apple.,22)
(cherry.,7)
(grape date honeydew banana.,27)
(cherry grape fig honeydew elderberry.,37)
(elderberry fig grape apple date cherry.,39)
(honeydew fig.,13)
(fig apple.,10)
(banana cherry.,14)
(grape honeydew fig apple date banana.,37)
(honeydew elderberry banana.,27)
(grape apple fig elderberry cherry.,34)
(honeydew elderberry date apple.,31)
(fig apple banana.,17)
(elderberry banana date honeydew.,32)
(elderberry grape date.,22)
(apple elderberry grape fig honeydew date.,41)
(date elderberry apple honeydew.,31)
(date honeydew apple cherry.,27)
```

– this one shows how many times that sentence is duplicated. Most are one with a couple of sentences showing up 2 or 3 times . for some reason I guess my screenshot didn't paste properly so I don't have it. But, here is the code that I used.

val transformedSentences = sentencesRDD.map(sentence => sentence).sortByValue()