B0004: Comparison Exception Error

Bug ID: B0004

Title: Comparison Exception Error

Vender: ZHU Renjie	**Product:** News Clustering System	**Version**: 3.0
Severity: High	**Updated**: 2014-11-14	**OS**: Windows 7
Status: Closed	**Assigned to:** Fang Zhou	

Description:

When doing the IG sorting, the comparator will throw exceptions due to the occurrence of NaN of IG number. This bug is caused by Bug0004.

Steps to Reproduce:

Steps to Reproduce:

1.) Prepare the articles for each category under raw_data directory

2.) Make sure that there is not information gain file existed.

3.) Start the program

Expected result:

The IG number should be sorted successfully in ascending order.

Actual result:

An exception of illegalArgumentException is thrown out when sorting IG values.

```
Exception in thread "main" java.lang.IllegalArgumentException: Comparison method violates its general contract!
        at java.util.TimSort.mergeHi(TimSort.java:868)
        at java.util.TimSort.mergeAt(TimSort.java:485)
        at java.util.TimSort.mergeCollapse(TimSort.java:408)
        at java.util.TimSort.sort(TimSort.java:214)
        at java.util.TimSort.sort(TimSort.java:173)
        at java.util.Arrays.sort(Arrays.java:659)
        at java.util.Collections.sort(Collections.java:217)
        at source.FeatureSelection.sortMap(FeatureSelection.java:34)
        at source.FeatureSelection.featureSelect(FeatureSelection.java:20)
        at source.mainProcess.main(mainProcess.java:99)
```

Fang Zhou 2014-11-15 15:21 EDT

To get the feature words, we need to sort all the words by their information gain in descending order. However some errors happen when doing the comparison, we trace back the error and find that it is due to the appearance of NaN. Also the previous sorting gives an ascending order, which is not correct.

Before

```
ArrayList<Entry<String, Double>> sortedList = new ArrayList<Entry<String,
Double>>(IGMap.entrySet());

//sort the list according to map value
Collections.sort(sortedList, new Comparator<Map.Entry<String, Double>>(){

public int compare(Map.Entry<String, Double> entry1, Map.Entry<String, Double> entry2){

//error for below code, reason: some values are NaN, which will cause exception
double difference = entry1.getValue() - entry2.getValue();

        if(difference > 0)
                return 1;
        else if(difference == 0)
                return 0;
        else
                return -1;

}
});
```

After

After Change:

```
ArrayList<Entry<String, Double>> sortedList = new ArrayList<Entry<String,
Double>>(IGMap.entrySet());

//sort the list according to map value
Collections.sort(sortedList, new Comparator<Map.Entry<String, Double>>(){

public int compare(Map.Entry<String, Double> entry1, Map.Entry<String, Double> entry2){

        //add "-" to sort in descending order
        return -Double.compare(entry1.getValue(), entry2.getValue());
}
});
```

Zhu Renjie 2014-11-16 17:50 EDT

Test succeeds. The words are sorted in descending order.