Popularity Prediction of Online News



**Team Members**

Mayuresh Naik

Feroz Shaik

Sreekanth Reddy

1. [Introduction](#Introduction): 3

1.1 [Problem description](#_Problem_description:): 3

1.2 [Motivation](#_Motivation:): 3

1.3 [Report Organization](#_1.3_Report_Organization:): 3

2. [Data Exploration:](#DataExploration) 3

3. [Methodology](#Methodology): 4

3.1 [Data Preprocessing](#Preprocessing): 4

3.1.1 [Discretization](#Descretization): 4

3.1.2 [Aggregation](#Aggregation): 4

3.1.3 [Removal of irrelevant features](#irrelevantfeatures): 4

3.1.3 [Feature Subset Selection](#FeatureSubsetSelection): 4

3.2 [Mining the data](#DataMining): 5

3.3 [Performance Measures:](#PerformanceMeasures) 5

4. [Logic of the problem](#LogicofProblem): 6

4.1 [Pros and Cons:](#ProsCons) 7

4.2 [Recommendations](#recommendations): 7

5. [Conclusion](#Conclusion): 7

6. [Appendix](#Appendix): 8

[Model Output](#modeloutputs): 8

[Discretization Accuracy](#discretizationAccuracy): 12

[Feature Selection](#Featureappendix): 12

[Attributes Table](#attributedescription): 12

[Outlier](#outliers): 14

7. [Reference](#references) 15

1. **Introduction:**

## Problem description:

## When the World Wide Web was first invented in 1989, it was a very rudimentary version of what it is now. Since then, the Web has expanded to unconceivable heights, one vertical of which is online news. Due to this, prediction of the popularity of online news is becoming a trendy topic. But, the popularity of the news castings depends on “Entertainment/Lifestyle” and “Word ratings”, two factors which are not easy to predict. Thus, it is difficult to predict user behavior to know if the published news will become popular or not, both in general and to a specific audience. In this ocean that is online news, only a few websites will become popular. This can be seen in the disproportionality between the numbers of online news to the popular ones.

## Motivation:

## Large number of online news are getting published daily. Success of its popularity mainly depends on article contents as well as user interests. It is possible to predict the percentage popularity of news before publications by analyzing its features. This popularity prediction can be used by author/publishers for the news enhancement by optimizing the required features of articles which is ready for publication.

## 1.3 Report Organization:

The main purpose of this report is to notify the viewer about the reason behind choosing and solving this problem. It also tells the readers from where the dataset is retrieved along with its source and the characteristics of data. Later on, the report moves on to discuss about the various techniques used on the dataset to detect and remove anomalies like missing values, Noise, Outliers and extreme values in order to increase the data quality. It also describes how different methods like Aggregation, Feature subset selection helps to increase the model performance by removing the irrelevant attributes. The report then talks about how discretization helps in improving accuracies when applied on output attribute and rest of the input attributes. The report also gives a brief description on different classification algorithms like ZeroR, OneR, IBK, J48 and JRip are used on the dataset by recording their individual performances when tested among multiple test options. Then, it tells us the about the best suitable algorithm for our model by comparing all the algorithms used on the dataset. After the mining approaches the logic of problems usefulness and recommendations are explained. Finally, the domain problems which are faced while developing this project are described and the future scope of this project are detailed.

**2. Data Exploration:**

We obtained the data set regarding online news published on the “Mashable” forum, via the ***UCI machine learning community*** in the form of a CSV file. There are a variety of attributes which represents the characteristics of online news data. Among these, there are some attributes which play an important role in popularity prediction with the use of a classification or regression model. Some of those attributes are as follows*:*

*Shares, rate\_positive\_words, rate\_negative\_words, token\_title, token\_content, num\_videos, and num\_images*.

**Data Source URL**: [***http://archive.ics.uci.edu/ml/machine-learning-databases/00332/***](http://archive.ics.uci.edu/ml/machine-learning-databases/00332/)

|  |  |
| --- | --- |
| **Number of Records:** 39644 (before data preprocessing) **Number of Records:** 23246 (after data preprocessing) | **Characteristics of Data:** Multivariate |
|  |
| |  |  | | --- | --- | | **Attribute Property** | **Number of Attributes** | | Nominal | 15 | | Ordinal | 3 | | Ratio | 5 | | Numeric | 38 |   **Number of Attributes:** 60 | **Characteristics of attribute:** Integer, Numeric, Nominal |
| **Approach performed:** Discretization and Classification | **Missing Values:** 0 |

**Dataset Summary:**

**Missing Values and Outliers:** We do not have any missing values in our dataset and with the help of WEKA software we have used IQR (Inter Quartile Range) filter in order to remove **16398** outliers from our dataset.

**3. Methodology:**

**3.1 Data Preprocessing:**

**3.1.1** **Discretization:** As our output label ***“Shares”*** is numeric attribute having 1167 distinct values , we decided to discretize the output label into 3 distinct values (High, Medium, Low) using discretization process. Distribution of instances among 3 distinct values is as follows : ***High – 6670 (28.96%) , Low – 8856(38.56%) , Medium - 7720 (33.21%)***   
Also, most of the predictive classification algorithms like J48, NB, IBK requires discrete input attributes for better performance than continuous value input. Classification algorithms mainly focus on learning nominal feature values. Thus, Discretization helps in improving accuracy of model.  Using WEKA we performed Discretization on *23 attributes* having large amount of distinct values and then found out that the performance of the model built using NB Classifier, J48 and IBK improved after discretization. Refer Appendix A for more details.

**3.1.2** **Aggregation**: As our dataset has two fields named “num\_hrefs” (which has all the links of the published articles from different sources except “Mashable Forum”) and “num\_self\_href” (which contains only the links published by “Mashable Forum”). So, we have manually added all the links from “num\_self\_href” into “num\_hrefs” and removed “num\_self\_href” using Remove filter in weka leaving “num\_hrefs” with all the links from different sources.

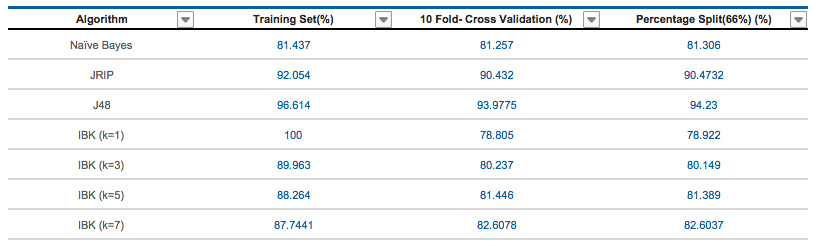
**3.1.3** **Removal of irrelevant features**: We have removed an attribute named “n\_non\_stop\_words” which describes the rate of nonstop words in an article. We have removed the field “n\_non\_stop\_words” because it has only one distinct value and the accuracy remained unchanged when we verified its relevance with class label both before and after applying “KNN/IBK/J48” algorithms.

**3.1.4** **Feature Subset Selection:** With the help of WEKA we have removed ***18 irrelevant features*** using ***Correlation and Learner based*** feature subset selection techniques. [“*CorrelationAttributeEval*” – “*Ranker*”] and [“*WrapperSubsetEval*” -- “*Best first*”] are the 2 Attribute evaluator and Ranker pairs used in Correlation based and Learner based subset selection respectively. Considering the output of Learning based feature selection and correlation based feature selection, we have decided to remove the ***18 attributes*** having correlation threshold factor less than **0.04.** Correlation threshold factor was decided by considering the best possible attributes subset provided by “wrapperEvaluater”. Hence, we now have ***43 attributes*** in total.

* 1. **Mining The Data:**

As we are building the predictive model which would predict the popularity of online news, we decided to apply supervised classification algorithms to build the model. ***OneR*** *and* ***ZeroR*** are the base algorithms we applied on our dataset owing to accuracy of ***60.178%*** *and* ***38.096%***respectively. We performed system performance analysis by applying different classification algorithms like J48,NB,JRIP,IBK and compared their accuracies, kappa statistics and RMS values obtained when tested with (Training set, Percentage split and Cross fold validation) as a testing data. Below table shows the % accuracy of a model built using different algorithms with different testing type.

*Accuracy Table A*



We could see from above table that accuracies of JRIP and J48 are better across all three testing types without any significant change in percentage values.

Working of these algorithms are listed below along with the parameters used:

* **J48**: This algorithm is used to generate the decision tree based on input attributes and splitting criteria. Output label of a testing instance is then predicted using generated decision tree by moving all the way from root node to leaf node. While running this algorithm confidence Factor was set to **0.25,** binarySplits was set to ***false*** and noFolds was set to ***3***.
* **NB Classifier**: Naïve Bayes classifier is based on Bayes theorem having strong independence assumptions while predicting the output label. It is the part of simple probabilistic classifier. Parameters set were *:* ***batchSize*** *= 100,* ***useKernelEstimator*** *= false.*
* **JRIP**: It is one of the rule based algorithm where classes are examined in increasing size with initial set of rules generated using incremental reduced error JRip (RIPPER). Parameters set were: folds = 3 , usePruning = true .
* **IBK**: Its simple and powerful K-Nearest Neighbor algorithm which decides the class of output label based on k-nearest neighbor calculated using Euclidian distance or cosine similarity. Majority vote of k-nearest neighbors is considered to decide the class label. We tried to build model with IBK having different K values like 1/3/5/7 and found k=3 to be best with better accuracy.
  1. **Performance Measures**:

We are using following performance measures to evaluate our model:

* **Accuracy**: It deals with fraction of instances classified correctly to the total number of instances. It’s one of the important performance measure to determine the system performance.
* **Kappa Statistics:** The kappa value is a metric that compares the resultant accuracy with that of expected accuracy. It is not only used to estimate single classifier but also used to estimate classifiers among themselves. In other terms kappa is a measure on classifiers performance, particularly on an unbalanced data set.
* **Root Mean Square Error:** It is a generally used measure of the differences between the sample values predicted by the model with that of observed values. In simple terms it’s a measure of accuracy, to relate anticipated errors of different models for a selected data but not between datasets. It increases with the variance of the frequency distribution of error magnitudes.

|  |  |  |  |
| --- | --- | --- | --- |
| **Algorithm** | **Accuracy** | **RMS** | **Kappa Stats** |
| Naïve Bayes | 81.26% | 0.316 | 0.7194 |
| JRIP | 90.43% | 0.2306 | 0.8553 |
| J48 | 93.97% | 0.1787 | 0.9089 |
| IBK (K=1) | 78.81% | 0.3759 | 0.6799 |
| IBK (K=3) | 80.24% | 0.3046 | 0.7022 |
| IBK (K=5) | 81.45% | 0.2898 | 0.7202 |
| IBK (K=7) | 82.60% | 0.2843 | 0.7381 |

**Motivation:** It’s necessary to evaluate the performance of the model built in order to check our efficiently and correctly expected class values are assigned to unknown instances. Kappa statistics and room mean square error helped in model evaluation by comparing the resultant accuracies with expected accuracies even by measuring the difference in sample values predicted by model with that of observed values. Thus, it allowed us to select model built with better classifier using above performance measure. Performance measures were estimated using k-fold cross validation (k=10) as it gives equal chance for all the subsets of instances to participate as a training set within (k-1) iterations. Thus, it helps to improve accuracy of the model. Below is the model performance evaluation measured using 10-fold cross validation testing technique.

*Performance measure table for different models. Table B*

Referring to above table, model built with J48 and JRIP shows better accuracy with low error rate and better Kappa stats. Also, performance of the system improves in case of IBK model as we go on increasing the value of k.

**4. Logic of Problem:**

Logic of problem played a prominent role for us while developing our web based decision supporting system. Initially it helped us to organize all our ideas, assumptions, implications, methods and approaches which we wanted to implement in developing our system. When we started building our model we used our logic of problem as a reference while performing each step and updated every time when the intended approach doesn’t work. In the first checkpoint the sections like purpose, assumptions, concepts, information and Interpretation gave us a basic idea on how to begin with our project. Later on while we were going ahead in developing our model through checkpoint 2 and 3 the sections like concepts, Assumptions, implications and Question as issue helped us to determine how we are looking at the issue, what concepts needs to be applied, what questions we need to elucidate, what assumptions we need to make, what results we might achieve whether the positive or negative and finally the solutions and conclusions we get after developing our model. Thus, unconditionally this tool helped us to think and collect the required information like the kind of data needs to be searched and what kind of methods needs to be chosen on our dataset in order to achieve the predicted results to make our model a reliable one.

**4.1** **Pros and Cons:** One major advantage of this tool is that it is developed in such a way that any person who wants to build anything can use it as an organizer for all his/her thoughts, ideas, assumptions and concepts which are needed to be applied in order to build his project in a systematic manner to achieve better results. One more positive thing about this tool is its elements of thought which has been divide into multiple sections in order to categorize all your thoughts and note the important things before actually building your project so that you can have a clear idea on what needs to be done and in what manner.The major drawback that everyone can notice is its UI which is very disturbing for the users who wants to use this tool apart from the tool has everything a user wants before building a model.

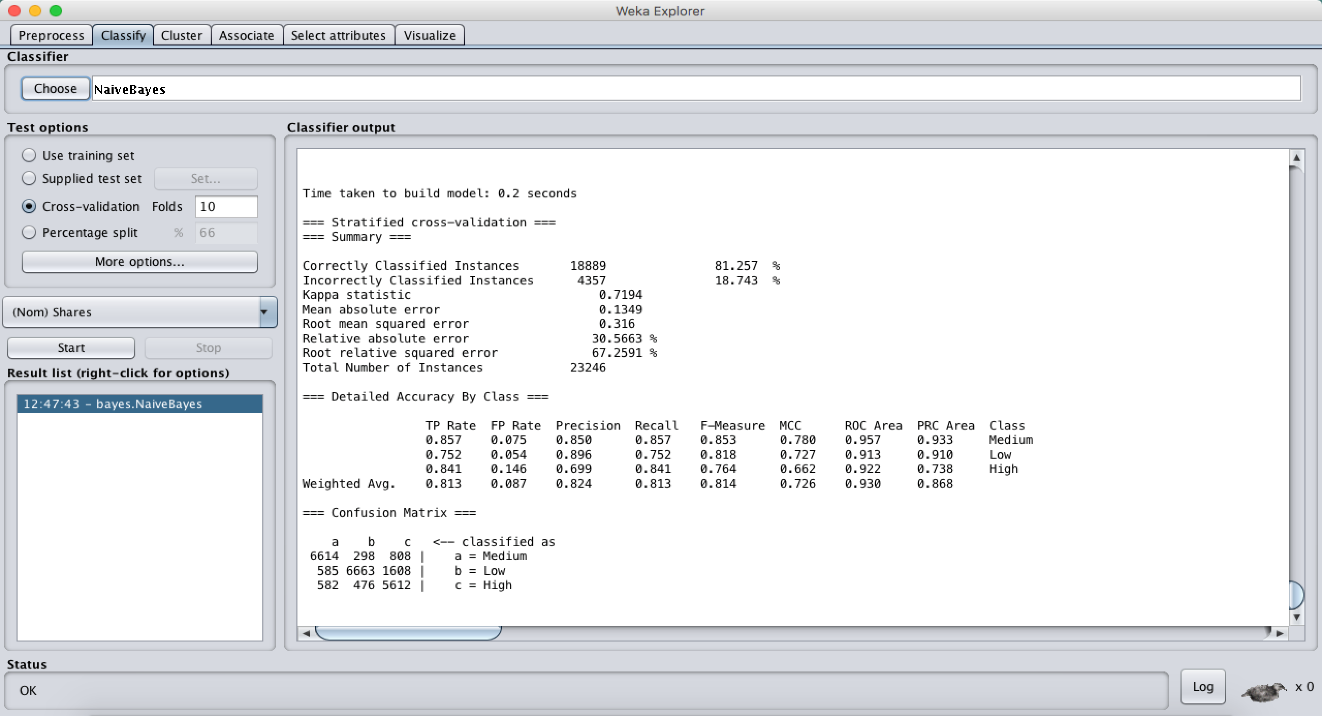
**4.2** **Recommendations:** After using the tool we thought it would be a good idea if the Statement of Problem section in the tool is updated with the points like domain area of the problem for which the tool is being used and it’s Time Frame or schedule in which it needs to be solved to make it more user friendly. We also thought it would be helpful if it is added with a Visualization section for converting the problem into high level visualization specifying the graphical texture of the problem. In this section user will add the important functionalities which he/she is intended to perform throughout the problem-solving process. This tool then generates the graphical flowchart showing sequential problem-solving steps based on their inputs.

**5.** **Conclusion:**  Considering the performance measures for the model built using different classifiers and the calculated accuracy matrix for different testing methods, we could say model built using ***J48 classifier*** is the better model giving *higher accuracy, low RMS error and highest Kappa stats greater than 0.9* which shows near perfect agreement [Refer Table A and Table B] . In case of IBK algorithm we could see that the accuracy of model increases with the increase of its k value. JRIP and Naïve Bayes shows consistent performance when applied along testing methods like “10-Fold cross validation” , “% Split” and “Training set” with no significant change in accuracy percentage.   
          ***J48 and JRIP*** showed better performance measures having accuracy more than 90% , RMS error less than 0.25 and Kappa stats more than 0.85.Our decision support system mainly deals with popularity prediction of news before publication which would help the publishers to enhance the features of the article based on the models’ popularity prediction. Thus, our model built with **J48** will provide better popularity prediction in terms of (Low,Medium,High).       
           One of the limitation of this model is that the news published at location 1 may not become popular at location 2. Hence, we would like to continue with this project in future by working on prediction of online news popularity based on location making Location based online news prediction our next target.

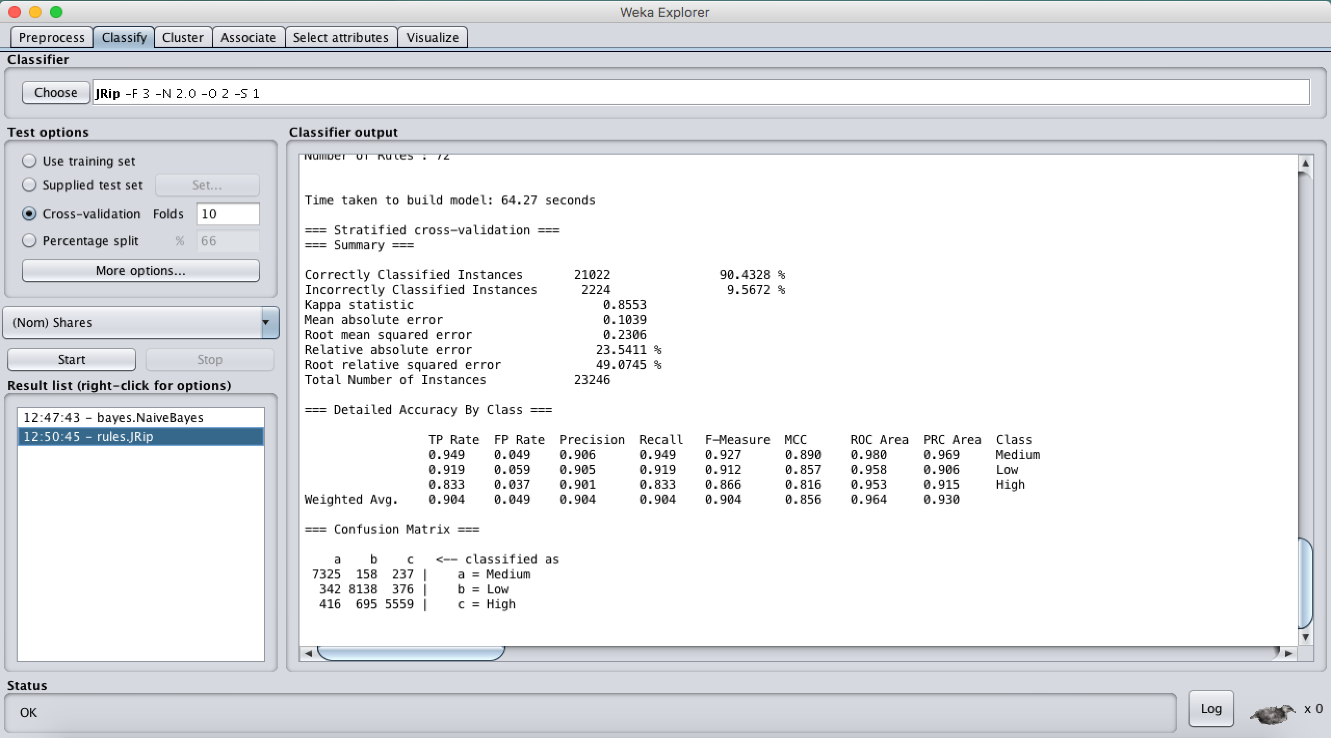
**6.** **Appendix A:**

**Model Outputs**

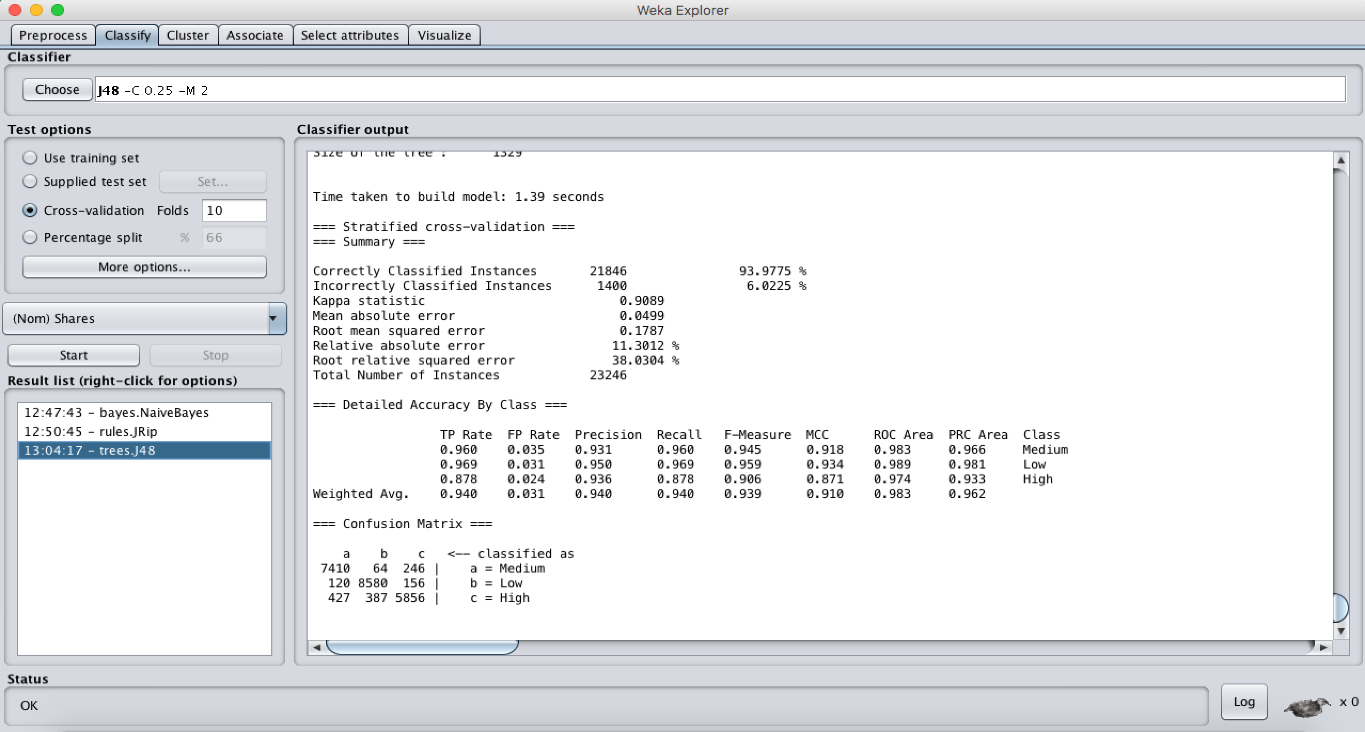
**Naive Bayes Classifier:**



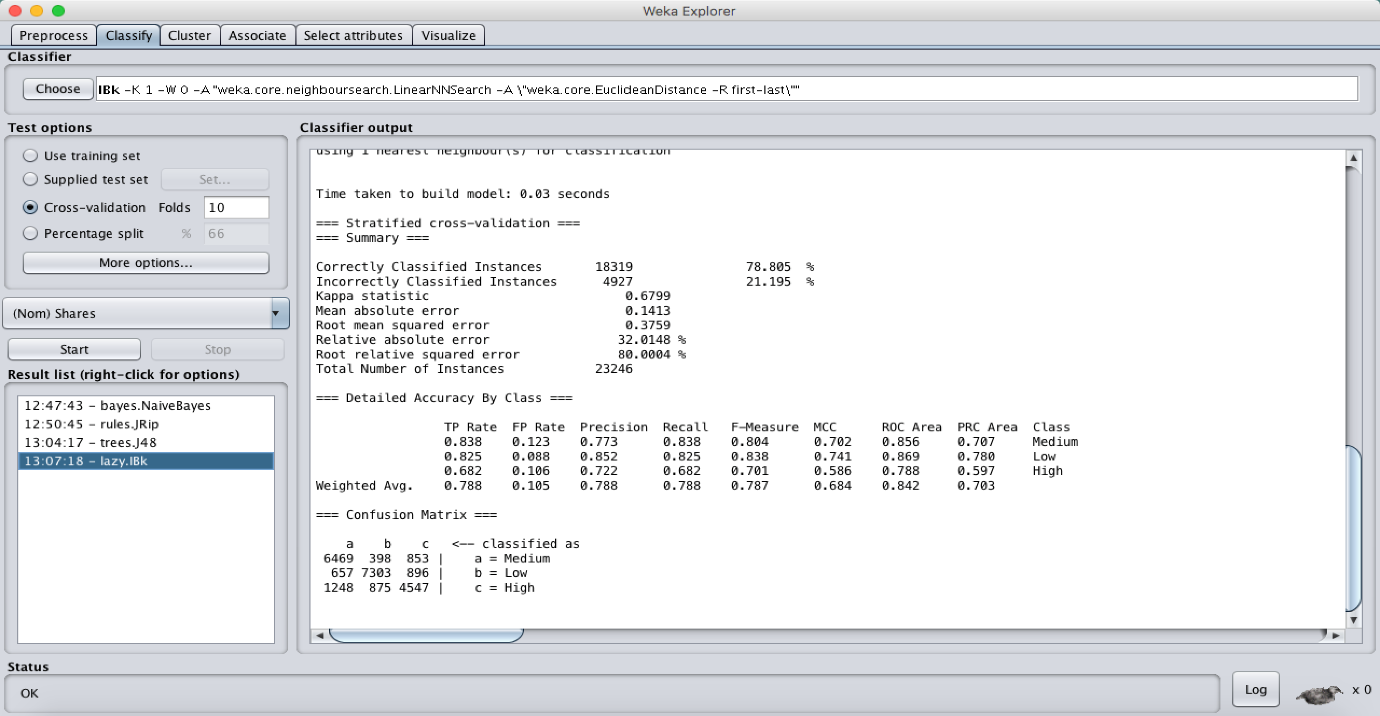
**JRIP Classifier:**



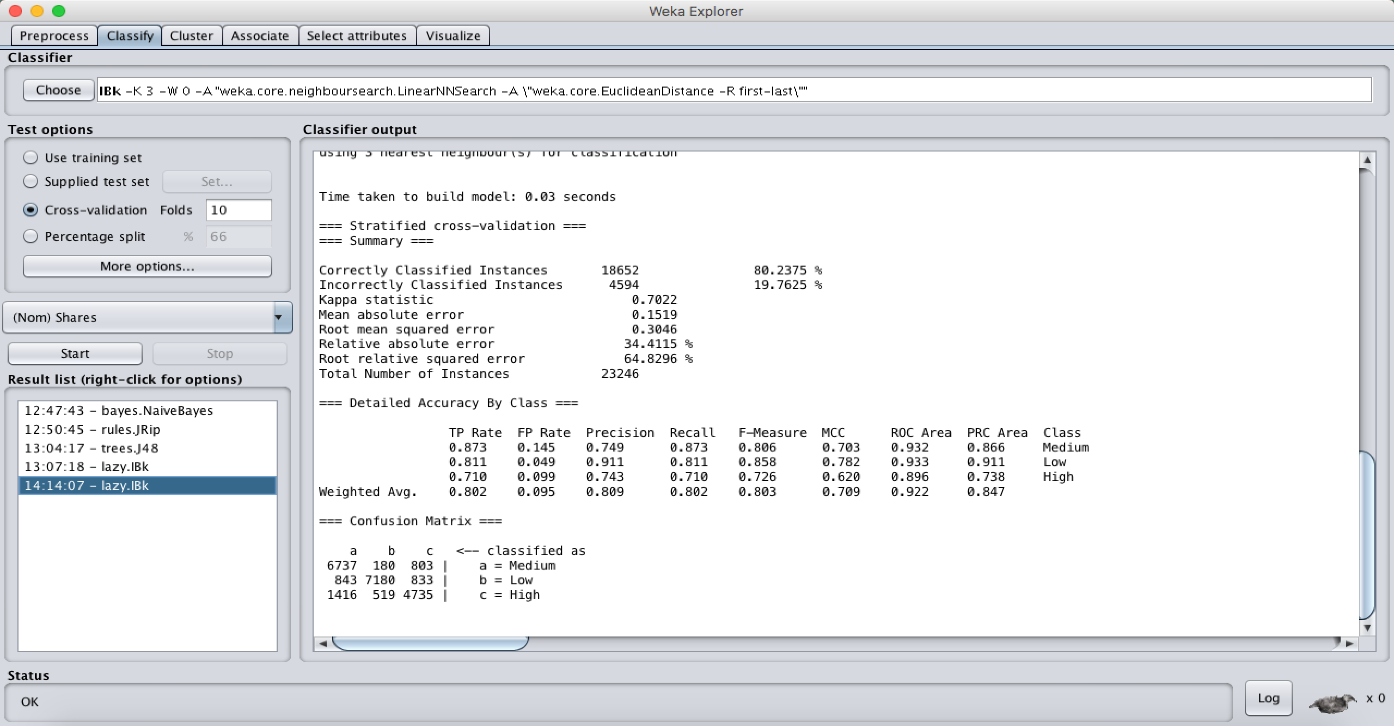
**J48:**



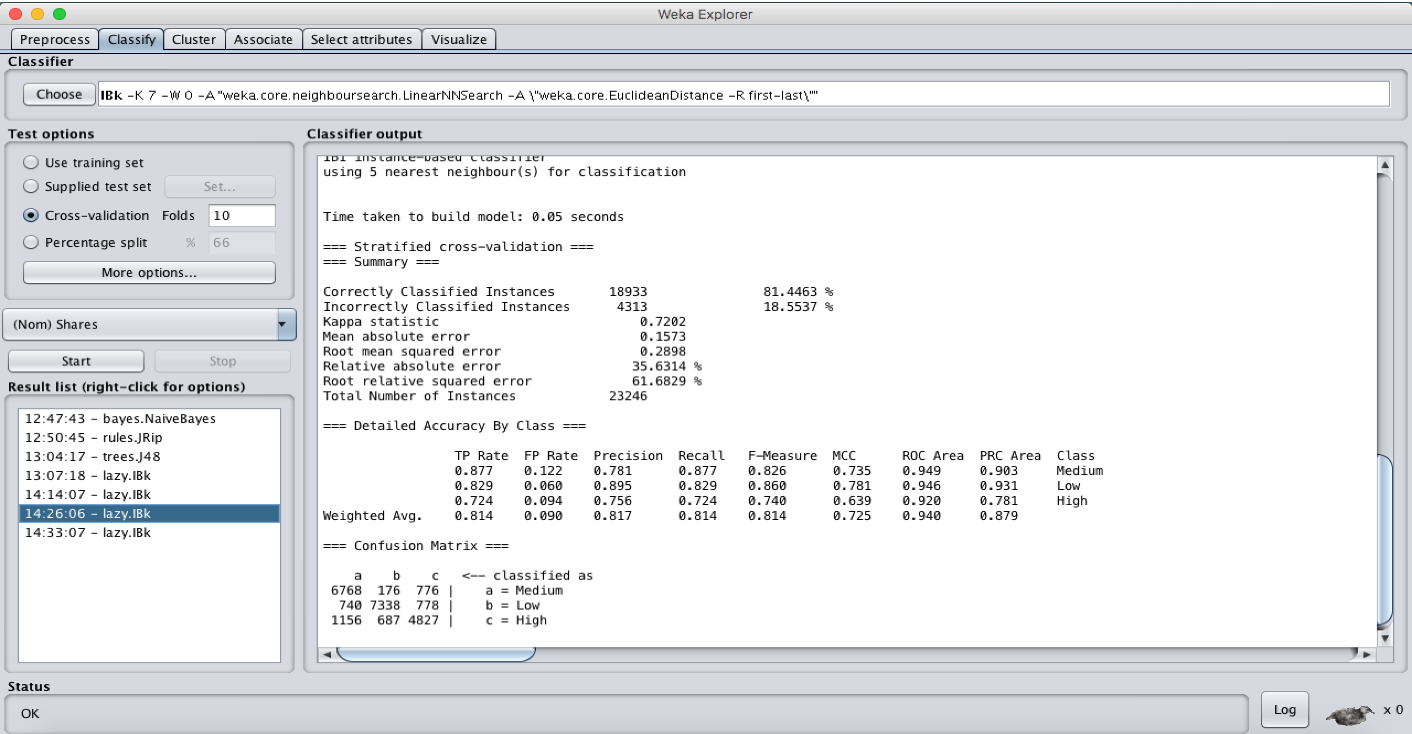
**Lazy IBK (K=1):**



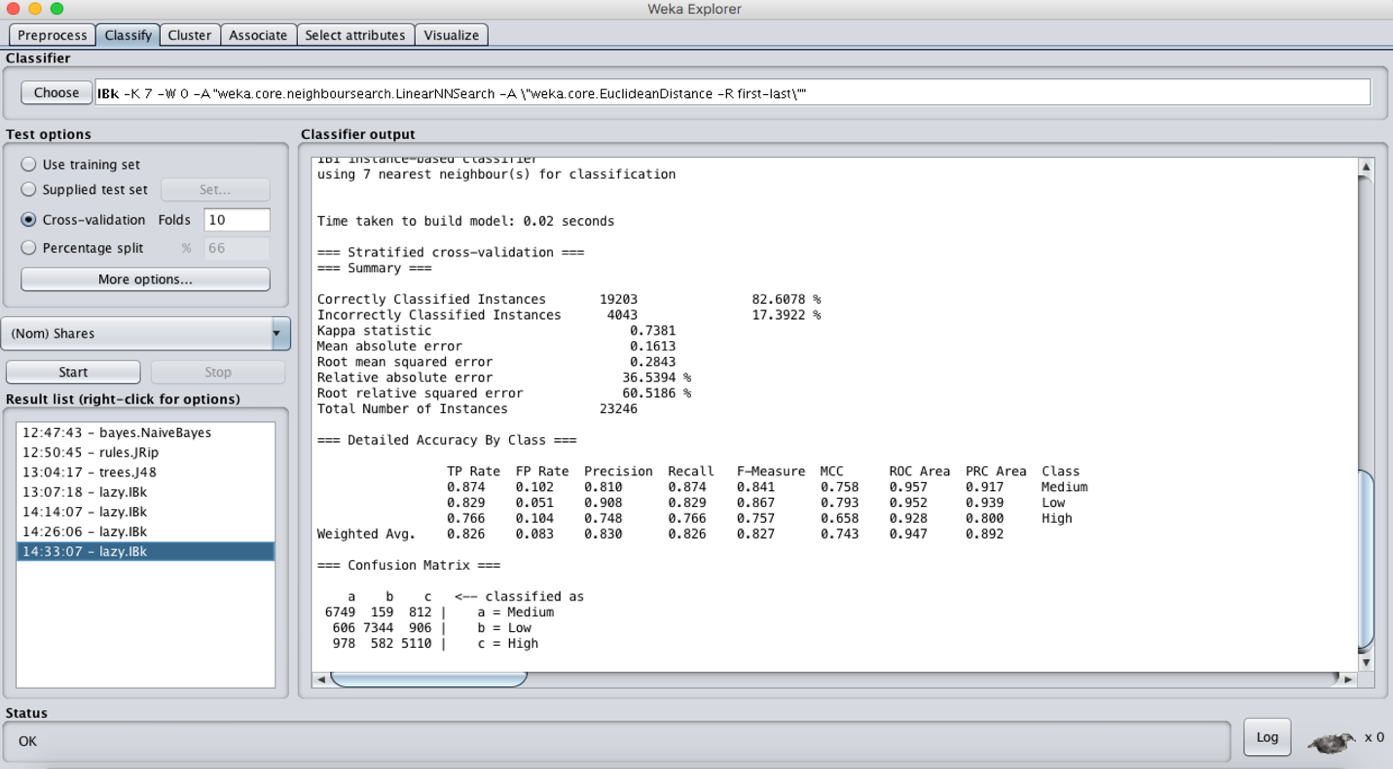
**Lazy IBK (K=3):**



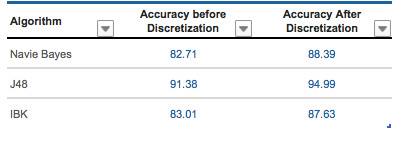
**Lazy IBK (K=5):**



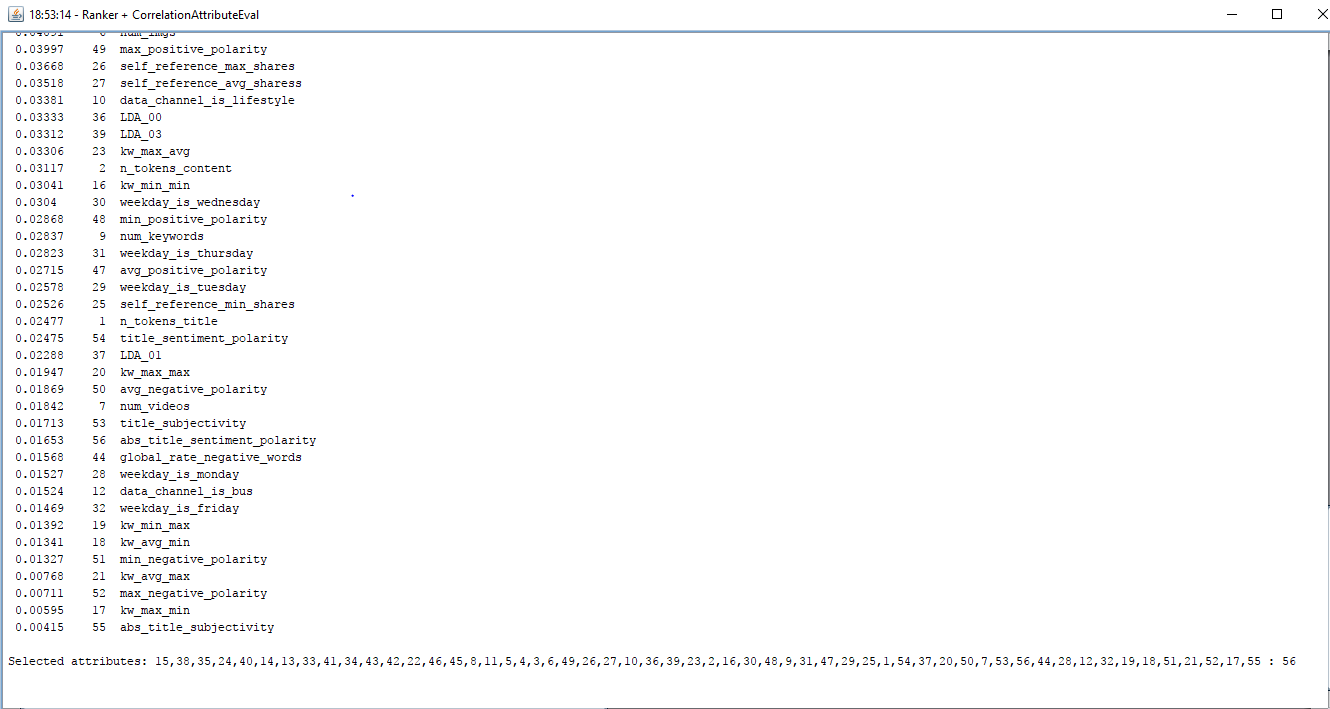
**Lazy IBK (K=7):**



**Before – After discretization accuracy**



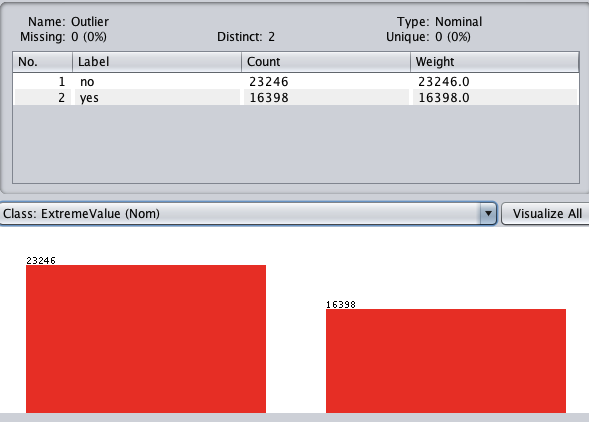
**Feature Subset Selection “CorrelationAttributeEval” – “Ranker”**



**Table of attribute description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Name of Attribute** | **Property** | **Description** | **Distinct Values** |
| 1 | data\_channel\_is\_entertainment | Nominal | Is data channel 'Entertainment'? | 2 |
| 2 | data\_channel\_is\_bus | Nominal | Is data channel 'Business'? | 2 |
| 3 | data\_channel\_is\_lifestyle | Nominal | Is data channel 'Lifestyle'? | 2 |
| 4 | data\_channel\_is\_socmed | Nominal | Is data channel 'Social Media'? | 2 |
| 5 | data\_channel\_is\_tech | Nominal | Is data channel 'Tech'? | 2 |
| 6 | data\_channel\_is\_world | Nominal | Is data channel 'World'? | 2 |
| 7 | weekday\_is\_monday | Nominal | Was the article published on a Monday? | 2 |
| 8 | weekday\_is\_tuesday | Nominal | Was the article published on a Tuesday? | 2 |
| 9 | weekday\_is\_wednesday | Nominal | Was the article published on a Wednesday? | 2 |
| 10 | weekday\_is\_thursday | Nominal | Was the article published on a Thursday? | 2 |
| 11 | weekday\_is\_friday | Nominal | Was the article published on a Friday? | 2 |
| 12 | weekday\_is\_saturday | Nominal | Was the article published on a Saturday? | 2 |
| 13 | weekday\_is\_Sunday | Nominal | Was the article published on a Sunday? | 2 |
| 14 | is\_weekend | Nominal | Was the article published on the weekend? | 2 |
| 15 | rate\_positive\_words | Ordinal | Rate of positive words among non-neutral tokens | 1555 |
| 16 | rate\_negative\_words | Ordinal | Rate of negative words among non-neutral tokens | 1554 |
| 17 | max\_negative\_polarity | Ordinal | Max. polarity of negative words | 39 |
| 18 | n\_tokens\_title | Ratio | Number of words in the title | 18 |
| 19 | n\_tokens\_conten | Ratio | Number of words in the content | 1821 |
| 20 | n\_unique\_tokens | Ratio | Rate of unique words in the content | 15120 |
| 21 | n\_non\_stop\_words | Ratio | Rate of non-stop words in the content | 1 |
| 22 | n\_non\_stop\_unique\_tokens | Ratio | Rate of unique non-stop words in the content | 11362 |
| 23 | num\_hrefs | Numeric | Number of links | 79 |
| 24 | num\_self\_hrefs | Numeric | Number of links to other articles published by Mashable | 31 |
| 25 | num\_imgs | Numeric | Number of images | 61 |
| 26 | num\_videos | Numeric | Number of videos | 39 |
| 27 | average\_token\_length | Numeric | Average length of the words in the content | 19536 |
| 28 | num\_keywords | Numeric | Number of keywords in the metadata | 10 |
| 29 | kw\_min\_min | Numeric | Worst keyword (min. shares) | 18 |
| 30 | kw\_max\_min | Numeric | Worst keyword (max. shares) | 916 |
| 31 | kw\_avg\_min | Numeric | Worst keyword (avg. shares) | 11518 |
| 32 | kw\_min\_max | Numeric | Best keyword (min. shares) | 776 |
| 33 | kw\_max\_max | Numeric | Best keyword (max. shares) | 25 |
| 34 | kw\_avg\_max | Numeric | Best keyword (avg. shares) | 19470 |
| 35 | kw\_min\_avg | Numeric | Avg. keyword (min. shares) | 9960 |
| 36 | kw\_max\_avg | Numeric | Avg. keyword (max. shares) | 12459 |
| 37 | kw\_avg\_avg | Numeric | Avg. keyword (avg. shares) | 23081 |
| 38 | self\_reference\_min\_shares | Numeric | Min. shares of referenced articles in Mashable | 938 |
| 39 | self\_reference\_max\_shares | Numeric | Max. shares of referenced articles in Mashable | 882 |
| 40 | self\_reference\_avg\_sharess | Numeric | Avg. shares of referenced articles in Mashable | 5418 |
| 41 | LDA\_00 | Numeric | Closeness to LDA topic 0 | 10875 |
| 42 | LDA\_01 | Numeric | Closeness to LDA topic 1 | 8186 |
| 43 | LDA\_02 | Numeric | Closeness to LDA topic 2 | 11316 |
| 44 | LDA\_03 | Numeric | Closeness to LDA topic 3 | 9838 |
| 45 | LDA\_04 | Numeric | Closeness to LDA topic 4 | 13703 |
| 46 | global\_subjectivity | Numeric | Text subjectivity | 21037 |
| 47 | global\_sentiment\_polarity | Numeric | Text sentiment polarity | 21093 |
| 48 | global\_rate\_positive\_words | Numeric | Rate of positive words in the content | 9386 |
| 49 | global\_rate\_negative\_words | Numeric | Rate of negative words in the content | 7117 |
| 50 | avg\_positive\_polarity | Numeric | Avg. polarity of positive words | 17273 |
| 51 | min\_positive\_polarity: | Numeric | Min. polarity of positive words | 21 |
| 52 | max\_positive\_polarity | Numeric | Max. polarity of positive words | 36 |
| 53 | avg\_negative\_polarity | Numeric | Avg. polarity of negative words | 9003 |
| 54 | min\_negative\_polarity | Numeric | Min. polarity of negative words | 52 |
| 55 | max\_negative\_polarity | Numeric | Max. polarity of negative words | 39 |
| 56 | title\_subjectivity | Numeric | Title subjectivity | 542 |
| 57 | title\_sentiment\_polarity | Numeric | Title polarity | 593 |
| 58 | abs\_title\_subjectivity | Numeric | Absolute subjectivity level | 438 |
| 59 | abs\_title\_sentiment\_polarity | Numeric | Absolute polarity level | 494 |
| 60 | shares | Numeric | Number of shares | 1196 |
| 61 | **Popularity (Label)** | Nominal | Target Class (Define popularity level [High, Medium, Low] | 3 |

**Outliers**

****

**7.** **References**

* *Data Mining practical machine learning tools and techniques*. (n.d.). Retrieved from <ftp://ftp.ingv.it/pub/manuela.sbarra/Data%20Mining%20Practical%20Machine%20Learning%20Tools%20and%20Techniques%20-%20WEKA.pdf>
* *Analytical Thinking By Dr. Linda Elder and Dr. Richard Paul. (n.d.). Retrieved from* <http://www.criticalthinking.org/files/SAM_Analytic_Think2007b.pdf>
* *Introduction to Data Mining*. (n.d.). Retrieved from

<https://www-users.cs.umn.edu/~kumar001/dmbook/sol.pdf>