TOPICS IN DATA MANAGEMENT
(ASSIGNMENT 4)

SANDHYA MURALI
sm2290@g.rit.edu

ABSTRACT:

This assignment is about extracting description and category from the api.txt file, parsing the data and storing the data in a csv file. This csv file is loaded in Python wherein tokenization is performed on the description and summary attribute of the csv file after which a TFIDF matrix is constructed from where important features are extracted. These features are split randomly into training and testing (where 70% is training data and 30% is testing data). The training data is passed into SVM, Random Forest and Multinomial Naive Bayes classifier and the categories are predicted for the test data. The accuracy and verification of the model is provided by displaying the accuracy and the number of correct and incorrect classifications made from the confusion matrix. This accuracy will also explain the best classifier from the three and the verification for the same.

INTRODUCTION:

The assignment involved classifying the description and summary attribute into specific categories by using text mining strategies wherein important words are extracted from the input attribute in order to predict the target variable. The important words are extracted using tokenization. This tokenization technique involves removing stop words, punctuations, non-ASCII characters, numbers and lemming. I tried using stemming and lemming. However lemming provided a better classification result as compared to stemming because lemming deals with morphological analysis of words whereas stemming operates on words without understanding the context of words. Therefore for example if we have words like going,gone,goes,went, for stemming the out was going,gone,go and went whereas in lemming it was go,go,go and go. Therefore using lemming provided a more generalized representation of words for multiple similar repeated words. After tokenization, a TF-IDF(Term frequency-Inverse Document frequency) was performed to determine the important features present in the description and summary attribute of the dataset. These features were then split randomly into 70% training data and 30% test data. This training data as passed into the SVM, Random Forest and Naive Bayes Classifier to build the model for predicting the category attribute. Now we use the test data to predict categories from the model. The accuracy is computed by comparing the predicted category and ground truth category. Verification and analysis of the models is made using the accuracy and the confusion matrix by determining the number of correctly and incorrectly classified values for each category.

The motivation behind performing classification is that it helps me discover some new categories based on the description and summary of the dataset. It is possible that the data can be put into several new categories apart from the category to which it is assigned. This approach gives me a new insight as to how a new conclusion can be drawn from the dataset by predicting new categories using the description and summary attribute as an input to the model built. By performing tokenization; one can infer that many new words can be discovered as important which can discover a new conclusion for the text data. This can further draw new conclusions about the data based on new classifications. In class, we spoke about different classification algorithms like Decision tree, Naive Bayes, SVM and so on. Using these classifiers, I have mined the text data to infer new predictions about this data which could be used for new analysis about this data. Therefore, the benefits of using my model is that based on semantic similarity between the words after performing tokenization , one can infer new categories which gives more insights about the model. There are some categories (like Other, Science) in the dataset which are very ambiguous and can have wider meaning. This may not give some clarity or insight about the data. Through this technique, some new categories or topics can be derived which gives better insight about the words in the document.

The remaining section is organized as follows. The design provides a detail description of the model built; Screenshots show some screenshots of the result obtained; Challenges talk about the challenges faced followed by conclusion.

DESIGN:

This section provides a detailed description about how the model is built right from the preprocessing stage to the model building and classifying stage. The model built is a classifier that predicts categories from the description and summary attribute of the dataset. The csv file is generated in java whereas the data mining problem is coded in python. The packages used for Data mining is Sklearn and NLTK.

The csv file is read which consists of description, summary and category. For the description and summary attribute, several preprocessing is done to extract only the important words instead of the entire data. First unwanted white spaces are removed and converted to lowercase. The non-ASCII and numbers are removed followed by punctuation marks using regular expression. The stop words are also remove using the nltk package where a list of stop words are provided. The stop words from the description and summary attribute are removed based on the list of stop words provided in the nltk package. The resultant are a list of words free of punctuation, white spaces and stop words; however they contain a lot of similar words (like word,words,wordings). In order to generalize these words, I have performed lemmatization. After lemmatization, I have got a list of nouns that are considered important from the description and summary attribute.

Next, I tried to perform Count Vectorization by counting the words occurring in each of the description and summary attribute. However, just knowing the count of the words does not reveal much of information. For example a word “provides” may occur many times in a sentence whereas a word “Obama” may not occur many times. Therefore, we cannot conclude the importance of the word based on its count. Therefore, to accurately determine the importance of the word, I have used TF-IDF (Term frequency-Inverse Document frequency) instead of Count Vectorization to determine importance of the words in the document. TF determines how many times a word occurs in a document where as IDF is the inverse of how many records the word occurs in. It's the inverse because, fewer the word occurs; more weightage we want to give that word. In the TFIDF vectorizer, I have used attributes such as min_df, max_df, ngram_range and max_features. The min_df stands for considering those words that occur at least in two of the description and summary attribute and max_df stands for considering those words that occur in less than 95% of the description and summary attribute. Ngram_range refers to a group of words that must be present together in the description and summary attribute and max_features is the maximum number of features we want to extract from the description and summary attribute.

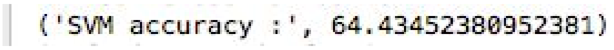
Once the TFIDF matrix is built, I have extracted the features that must be used to build the model. Before building the model; I have randomly split the features into training and testing (70% training and 30% testing). Once the data is split;I have passed the training data into the 3 classifiers (SVM, Random Forest and Naive Bayes). In Random Forest I have used the number of trees as 500. Once the model is build, I have passed the test data to predict the categories. This predicted result is compared with the ground truth category (obtained from our dataset) to compute the accuracy and build a confusion matrix.

In order to verify my models, I have determined the number of correctly classified and number of incorrectly classified ie True positive and False Negatives for each of categories. An analysis is made explaining the reason as to not classifying accurately for some categories whereas for some categories, there is a good prediction result. To support my answer, I have a confusion matrix and the accuracy score along with TP and FN for every category. The verification is explained with screenshots in the screenshot section.

SCREENSHOTS:

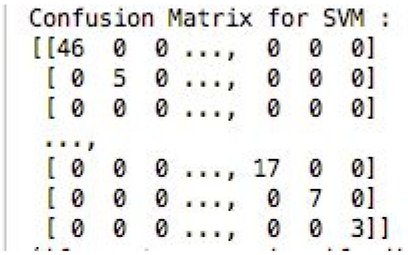
SVM classifier:

The SVM accuracy for the predicted data is close to 65%.



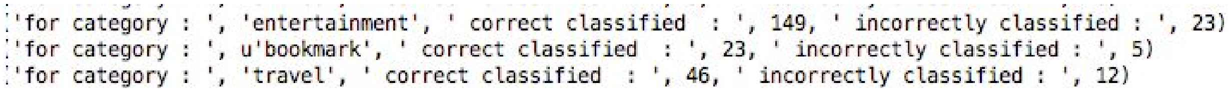
```
('SVM accuracy :', 64.43452380952381)
```

The confusion matrix is as follows.

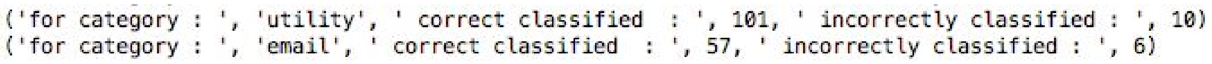


```
Confusion Matrix for SVM :
[[46  0  0 ...,  0  0  0]
 [ 0  5  0 ...,  0  0  0]
 [ 0  0  0 ...,  0  0  0]
 ...,
 [ 0  0  0 ..., 17  0  0]
 [ 0  0  0 ...,  0  7  0]
 [ 0  0  0 ...,  0  0  3]]
```

For better understanding of the confusion matrix, I have determined the number of correctly classified and incorrectly classified for each class (for each category)



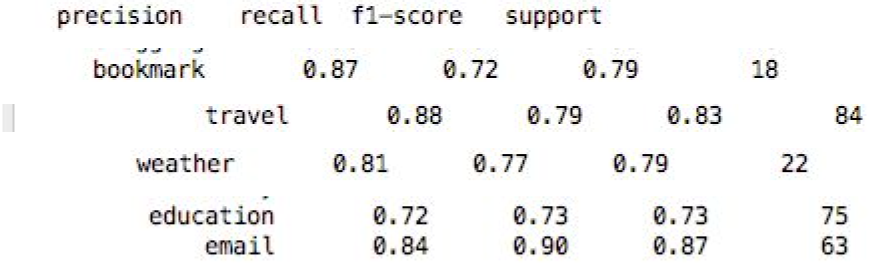
```
('for category : ', 'entertainment', ' correct classified  : ', 149, ' incorrectly classified : ', 23)
('for category : ', u'bookmark', ' correct classified  : ', 23, ' incorrectly classified : ', 5)
('for category : ', 'travel', ' correct classified  : ', 46, ' incorrectly classified : ', 12)
```



```
('for category : ', 'utility', ' correct classified  : ', 101, ' incorrectly classified : ', 10)
('for category : ', 'email', ' correct classified  : ', 57, ' incorrectly classified : ', 6)
```

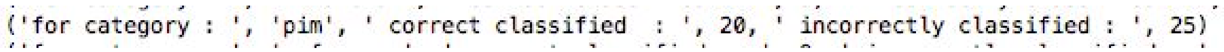
In the above screenshot we can see that, the categories entertainment, travel, utility give a fairly good prediction for the test data. This is because these keywords for category are self explanatory that give us insight about the data. For example, from travel, we can have an idea that the feature must be some topic pertaining to traveling as the meaning is not ambiguous and therefore classified 46 correctly and incorrectly classifies 12.

To further support my analysis, I have determined the precision,recall and f-1 score for the categories. Since the number of correctly and incorrectly classified does not talk about false positives and false negatives, I have used precision and recall measure to analyse the misses and false alarms for the categories.

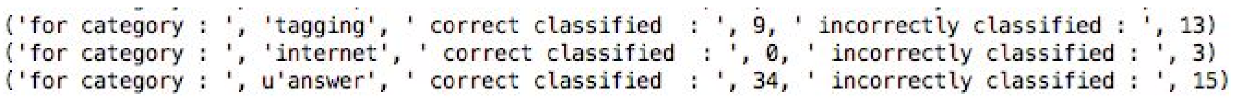


```
    precision    recall  f1-score   support

      bookmark       0.87      0.72      0.79        18

              travel       0.88      0.79      0.83        84

         weather       0.81      0.77      0.79        22

          education       0.72      0.73      0.73        75
              email       0.84      0.90      0.87        63
```

Higher the precision, the higher is the change of eliminating false positives. Higher the recall value, higher is the chance of eliminating false negatives. For example, travel has a correct classification 46 and incorrect classification of 12 has a precision value of 0.88 and recall value of 0.79. F-1 score is the harmonic mean of precision and recall and has a value of 0.83. From the precision and recall values (since they both are greater than 0.5), there are higher chances of not considering false alarms and neither missing any relevant attribute.



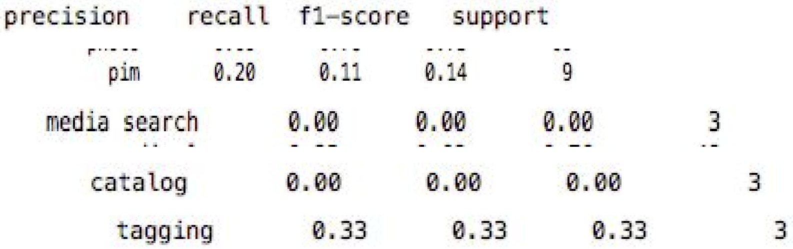
```
('for category : ', 'pim', ' correct classified  : ', 20, ' incorrectly classified : ', 25)
```



```
('for category : ', 'tagging', ' correct classified  : ', 9, ' incorrectly classified : ', 13)
('for category : ', 'internet', ' correct classified  : ', 0, ' incorrectly classified : ', 3)
('for category : ', u'answer', ' correct classified  : ', 34, ' incorrectly classified : ', 15)
```

On the other hand, the categories like tagging, answer, pim do not reveal much about the context of the description and summary attribute. For example, the category answer does not provide any meaningful category for the attribute. The category pim isn't also very clear with the meaning. It actually means Product Information Management. However, due to the abbreviation used, features who must actually belong to this category are not properly classified. Due to this ambiguity, pim category has only 20 correctly classified and 25 incorrectly classified as the features are randomly put based on the model created and how the model learns the features for that category.

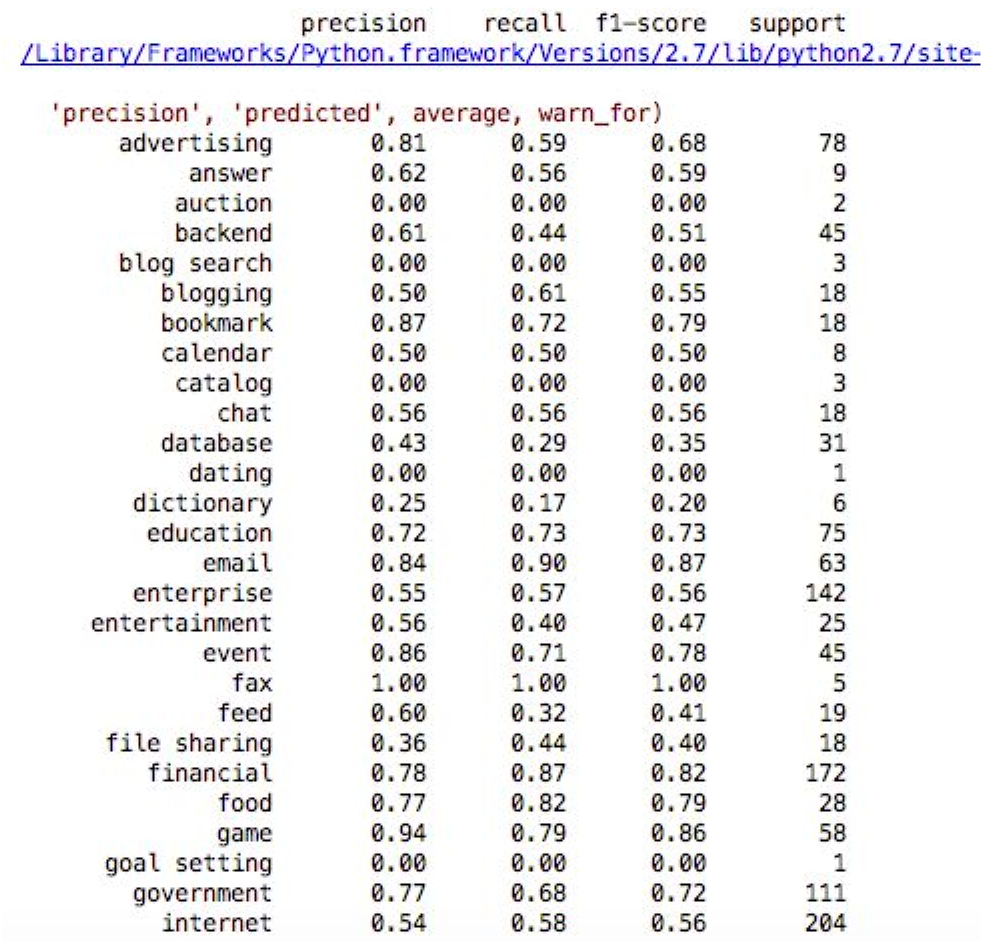
To further support my analysis, I have determined the precision,recall and f-1 score for the categories. Since the number of correctly and incorrectly classified does not talk about false positives and false negatives, I have used precision and recall measure to analyse the misses and false alarms for the categories.



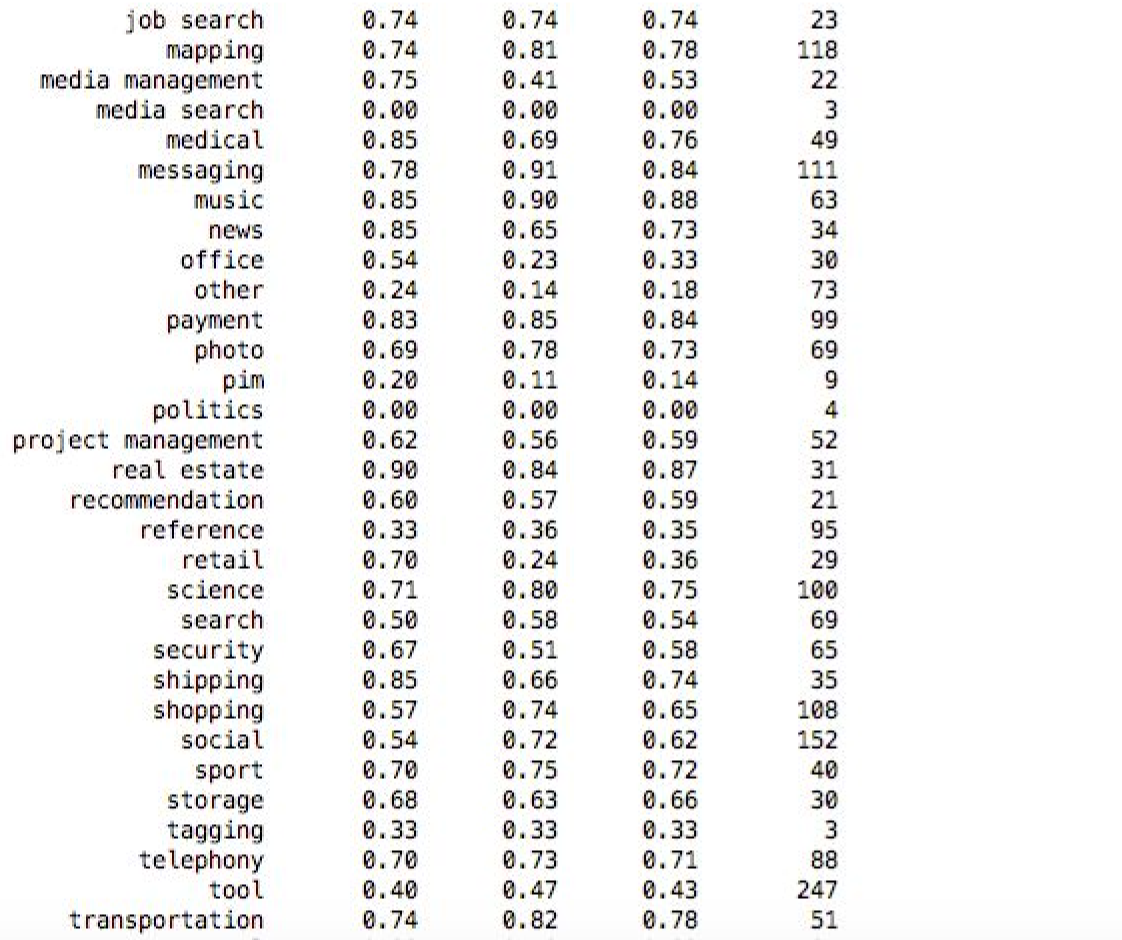
```
precision    recall  f1-score   support

       pim      0.20      0.11      0.14        9

  media search       0.00       0.00       0.00         3

     catalog       0.00       0.00       0.00         3

       tagging       0.33       0.33       0.33         3
```

Lower the precision, the lower is the change of eliminating false positives. Lower the recall value, lower is the chance of eliminating false negatives. For example, pim has a correct classification 20 and incorrect classification of 25 has a precision value of 0.20 and recall value of 0.11. F-1 score is the harmonic mean of precision and recall and has a value of 0.14. From the precision and recall values (since they both are lesser than 0.5), there are lower chances of not considering false alarms and neither missing any relevant attribute.

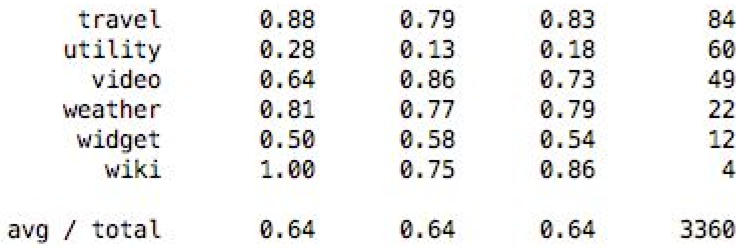
The precision, recall and f-1 score for all the categories is as follows:



```
                   precision    recall  f1-score   support
/Library/Frameworks/Python.framework/Versions/2.7/lib/python2.7/site-

  'precision', 'predicted', average, warn_for)
     advertising       0.81      0.59      0.68        78
          answer       0.62      0.56      0.59         9
         auction       0.00      0.00      0.00         2
         backend       0.61      0.44      0.51        45
     blog search       0.00      0.00      0.00         3
        blogging       0.50      0.61      0.55        18
        bookmark       0.87      0.72      0.79        18
        calendar       0.50      0.50      0.50         8
         catalog       0.00      0.00      0.00         3
            chat       0.56      0.56      0.56        18
        database       0.43      0.29      0.35        31
          dating       0.00      0.00      0.00         1
      dictionary       0.25      0.17      0.20         6
       education       0.72      0.73      0.73        75
           email       0.84      0.90      0.87        63
      enterprise       0.55      0.57      0.56       142
   entertainment       0.56      0.40      0.47        25
           event       0.86      0.71      0.78        45
             fax       1.00      1.00      1.00         5
            feed       0.60      0.32      0.41        19
    file sharing       0.36      0.44      0.40        18
       financial       0.78      0.87      0.82       172
            food       0.77      0.82      0.79        28
            game       0.94      0.79      0.86        58
    goal setting       0.00      0.00      0.00         1
      government       0.77      0.68      0.72       111
        internet       0.54      0.58      0.56       204
```

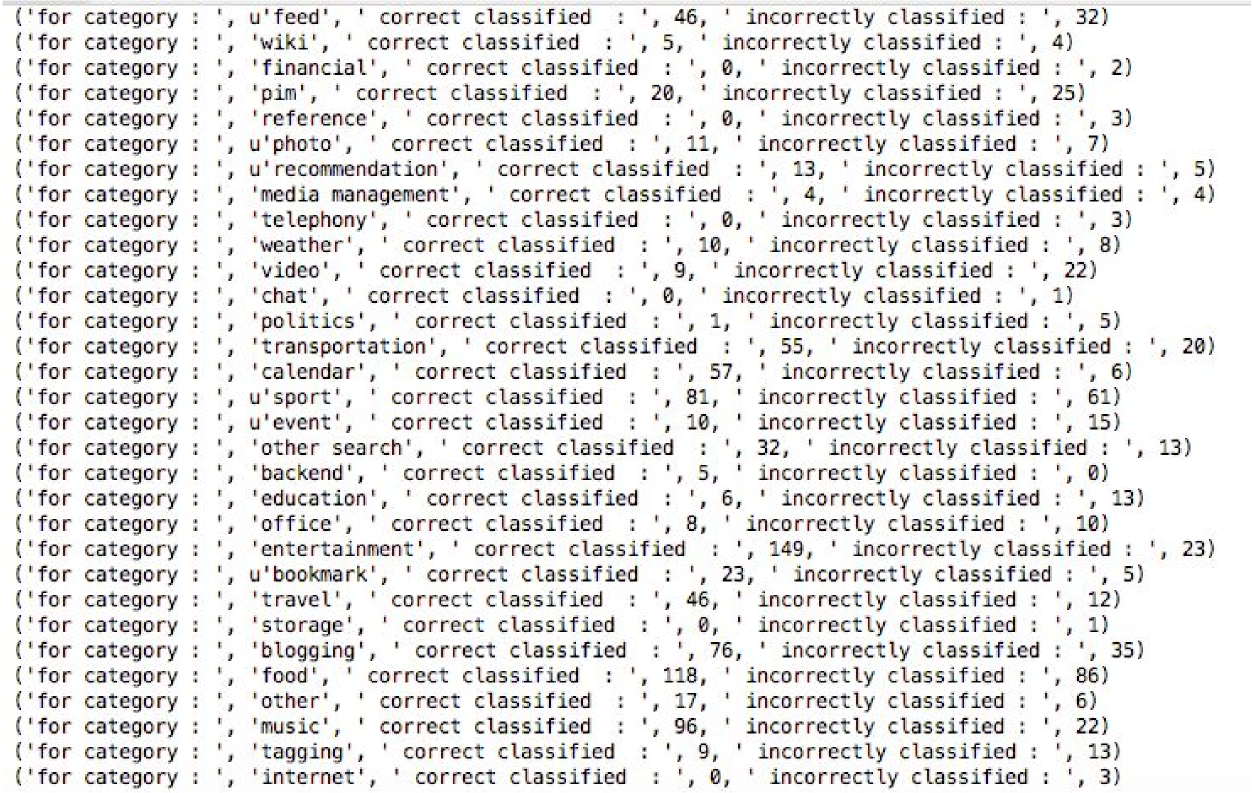


```
        job search       0.74      0.74      0.74        23
           mapping       0.74      0.81      0.78       118
  media management       0.75      0.41      0.53        22
      media search       0.00      0.00      0.00         3
           medical       0.85      0.69      0.76        49
         messaging       0.78      0.91      0.84       111
             music       0.85      0.90      0.88        63
              news       0.85      0.65      0.73        34
            office       0.54      0.23      0.33        30
             other       0.24      0.14      0.18        73
           payment       0.83      0.85      0.84        99
             photo       0.69      0.78      0.73        69
               pim       0.20      0.11      0.14         9
          politics       0.00      0.00      0.00         4
project management       0.62      0.56      0.59        52
       real estate       0.90      0.84      0.87        31
    recommendation       0.60      0.57      0.59        21
         reference       0.33      0.36      0.35        95
            retail       0.70      0.24      0.36        29
           science       0.71      0.80      0.75       100
            search       0.50      0.58      0.54        69
          security       0.67      0.51      0.58        65
          shipping       0.85      0.66      0.74        35
          shopping       0.57      0.74      0.65       108
            social       0.54      0.72      0.62       152
             sport       0.70      0.75      0.72        40
           storage       0.68      0.63      0.66        30
           tagging       0.33      0.33      0.33         3
         telephony       0.70      0.73      0.71        88
              tool       0.40      0.47      0.43       247
    transportation       0.74      0.82      0.78        51
```

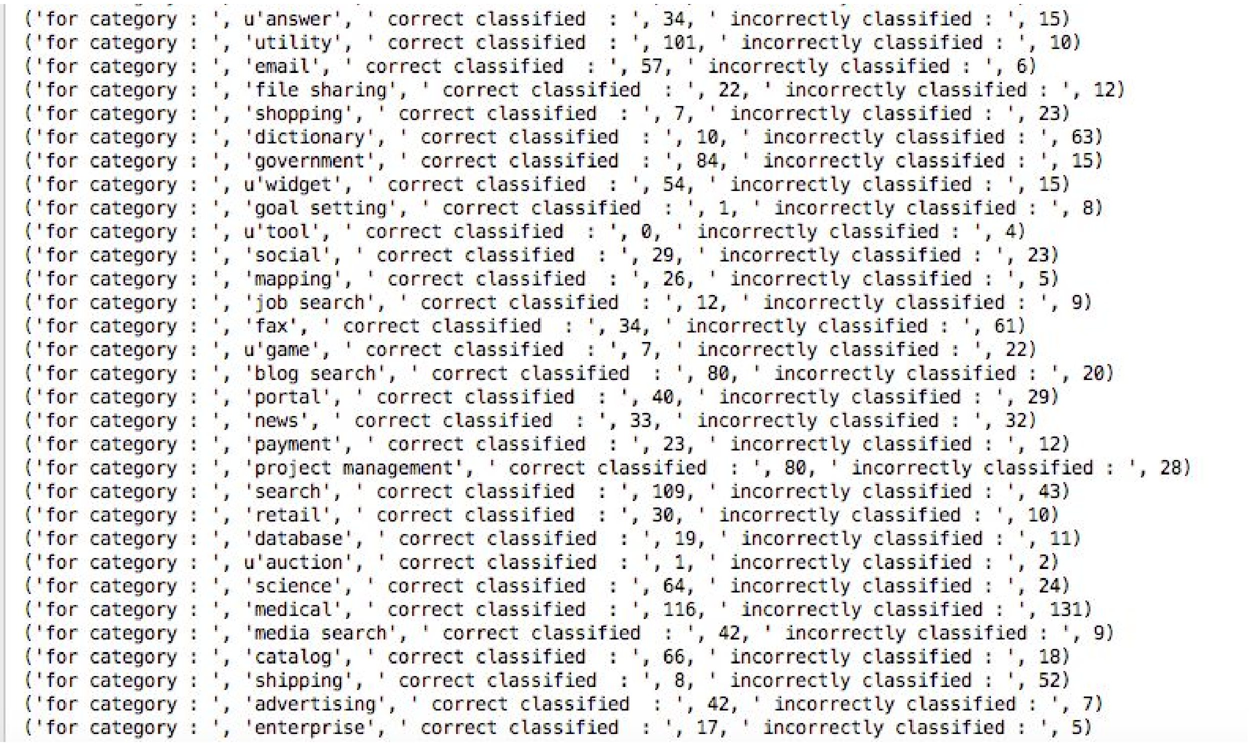


```
         travel       0.88      0.79      0.83        84
        utility       0.28      0.13      0.18        60
          video       0.64      0.86      0.73        49
        weather       0.81      0.77      0.79        22
         widget       0.50      0.58      0.54        12
           wiki       1.00      0.75      0.86         4

    avg / total       0.64      0.64      0.64      3360
```

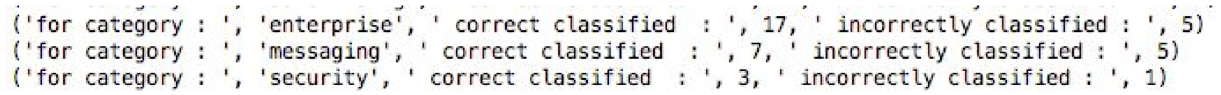
The count of correctly and incorrectly classified for all the categories is as follows:



```
('for category : ', u'feed', ' correct classified  : ', 46, ' incorrectly classified : ', 32)
('for category : ', 'wiki', ' correct classified  : ', 5, ' incorrectly classified : ', 4)
('for category : ', 'financial', ' correct classified  : ', 0, ' incorrectly classified : ', 2)
('for category : ', 'pim', ' correct classified  : ', 20, ' incorrectly classified : ', 25)
('for category : ', 'reference', ' correct classified  : ', 0, ' incorrectly classified : ', 3)
('for category : ', u'photo', ' correct classified  : ', 11, ' incorrectly classified : ', 7)
('for category : ', u'recommendation', ' correct classified  : ', 13, ' incorrectly classified : ', 5)
('for category : ', 'media management', ' correct classified  : ', 4, ' incorrectly classified : ', 4)
('for category : ', 'telephony', ' correct classified  : ', 0, ' incorrectly classified : ', 3)
('for category : ', 'weather', ' correct classified  : ', 10, ' incorrectly classified : ', 8)
('for category : ', 'video', ' correct classified  : ', 9, ' incorrectly classified : ', 22)
('for category : ', 'chat', ' correct classified  : ', 0, ' incorrectly classified : ', 1)
('for category : ', 'politics', ' correct classified  : ', 1, ' incorrectly classified : ', 5)
('for category : ', 'transportation', ' correct classified  : ', 55, ' incorrectly classified : ', 20)
('for category : ', 'calendar', ' correct classified  : ', 57, ' incorrectly classified : ', 6)
('for category : ', u'sport', ' correct classified  : ', 81, ' incorrectly classified : ', 61)
('for category : ', u'event', ' correct classified  : ', 10, ' incorrectly classified : ', 15)
('for category : ', 'other search', ' correct classified  : ', 32, ' incorrectly classified : ', 13)
('for category : ', 'backend', ' correct classified  : ', 5, ' incorrectly classified : ', 0)
('for category : ', 'education', ' correct classified  : ', 6, ' incorrectly classified : ', 13)
('for category : ', 'office', ' correct classified  : ', 8, ' incorrectly classified : ', 10)
('for category : ', 'entertainment', ' correct classified  : ', 149, ' incorrectly classified : ', 23)
('for category : ', u'bookmark', ' correct classified  : ', 23, ' incorrectly classified : ', 5)
('for category : ', 'travel', ' correct classified  : ', 46, ' incorrectly classified : ', 12)
('for category : ', 'storage', ' correct classified  : ', 0, ' incorrectly classified : ', 1)
('for category : ', 'blogging', ' correct classified  : ', 76, ' incorrectly classified : ', 35)
('for category : ', 'food', ' correct classified  : ', 118, ' incorrectly classified : ', 86)
('for category : ', 'other', ' correct classified  : ', 17, ' incorrectly classified : ', 6)
('for category : ', 'music', ' correct classified  : ', 96, ' incorrectly classified : ', 22)
('for category : ', 'tagging', ' correct classified  : ', 9, ' incorrectly classified : ', 13)
('for category : ', 'internet', ' correct classified  : ', 0, ' incorrectly classified : ', 3)
```



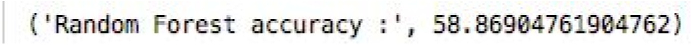
```
('for category : ', u'answer', ' correct classified  : ', 34, ' incorrectly classified : ', 15)
('for category : ', 'utility', ' correct classified  : ', 101, ' incorrectly classified : ', 10)
('for category : ', 'email', ' correct classified  : ', 57, ' incorrectly classified : ', 6)
('for category : ', 'file sharing', ' correct classified  : ', 22, ' incorrectly classified : ', 12)
('for category : ', 'shopping', ' correct classified  : ', 7, ' incorrectly classified : ', 23)
('for category : ', 'dictionary', ' correct classified  : ', 10, ' incorrectly classified : ', 63)
('for category : ', 'government', ' correct classified  : ', 84, ' incorrectly classified : ', 15)
('for category : ', u'widget', ' correct classified  : ', 54, ' incorrectly classified : ', 15)
('for category : ', 'goal setting', ' correct classified  : ', 1, ' incorrectly classified : ', 8)
('for category : ', u'tool', ' correct classified  : ', 0, ' incorrectly classified : ', 4)
('for category : ', 'social', ' correct classified  : ', 29, ' incorrectly classified : ', 23)
('for category : ', 'mapping', ' correct classified  : ', 26, ' incorrectly classified : ', 5)
('for category : ', 'job search', ' correct classified  : ', 12, ' incorrectly classified : ', 9)
('for category : ', 'fax', ' correct classified  : ', 34, ' incorrectly classified : ', 61)
('for category : ', u'game', ' correct classified  : ', 7, ' incorrectly classified : ', 22)
('for category : ', 'blog search', ' correct classified  : ', 80, ' incorrectly classified : ', 20)
('for category : ', 'portal', ' correct classified  : ', 40, ' incorrectly classified : ', 29)
('for category : ', 'news', ' correct classified  : ', 33, ' incorrectly classified : ', 32)
('for category : ', 'payment', ' correct classified  : ', 23, ' incorrectly classified : ', 12)
('for category : ', 'project management', ' correct classified  : ', 80, ' incorrectly classified : ', 28)
('for category : ', 'search', ' correct classified  : ', 109, ' incorrectly classified : ', 43)
('for category : ', 'retail', ' correct classified  : ', 30, ' incorrectly classified : ', 10)
('for category : ', 'database', ' correct classified  : ', 19, ' incorrectly classified : ', 11)
('for category : ', u'auction', ' correct classified  : ', 1, ' incorrectly classified : ', 2)
('for category : ', 'science', ' correct classified  : ', 64, ' incorrectly classified : ', 24)
('for category : ', 'medical', ' correct classified  : ', 116, ' incorrectly classified : ', 131)
('for category : ', 'media search', ' correct classified  : ', 42, ' incorrectly classified : ', 9)
('for category : ', 'catalog', ' correct classified  : ', 66, ' incorrectly classified : ', 18)
('for category : ', 'shipping', ' correct classified  : ', 8, ' incorrectly classified : ', 52)
('for category : ', 'advertising', ' correct classified  : ', 42, ' incorrectly classified : ', 7)
('for category : ', 'enterprise', ' correct classified  : ', 17, ' incorrectly classified : ', 5)
```



```
('for category : ', 'enterprise', ' correct classified  : ', 17, ' incorrectly classified : ', 5)
('for category : ', 'messaging', ' correct classified  : ', 7, ' incorrectly classified : ', 5)
('for category : ', 'security', ' correct classified  : ', 3, ' incorrectly classified : ', 1)
```

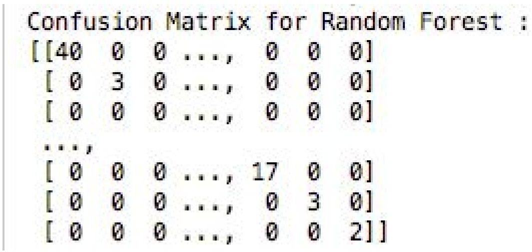
Random Forest Classifier:

The accuracy for Random Forest classifier which consists of an ensemble of 500 trees is as follows. The accuracy of Random Forest is not as good as SVM classifier but fairly manages to give a decent accuracy.



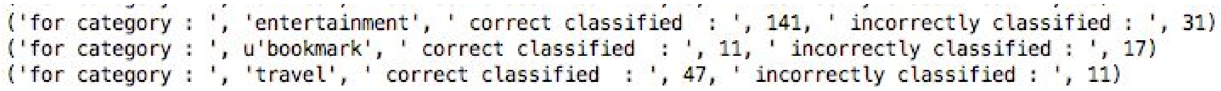
```
('Random Forest accuracy :', 58.86904761904762)
```

The confusion matrix for Random Forest is as follows:

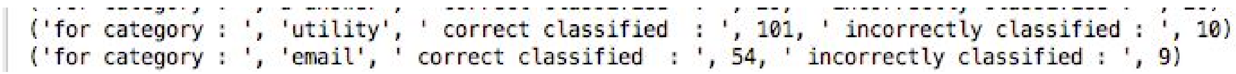


```
Confusion Matrix for Random Forest :
[[40  0  0 ...,  0  0  0]
 [ 0  3  0 ...,  0  0  0]
 [ 0  0  0 ...,  0  0  0]
 ...,
 [ 0  0  0 ...,  17  0  0]
 [ 0  0  0 ...,  0  3  0]
 [ 0  0  0 ...,  0  0  2]]
```

The confusion matrix seen for Random Forest classifier gives visible deviation in terms of accuracy and classification as compared to SVM classifier. For better understanding of the confusion matrix, I have determined the number of correctly classified and incorrectly classified for each class (for each category)



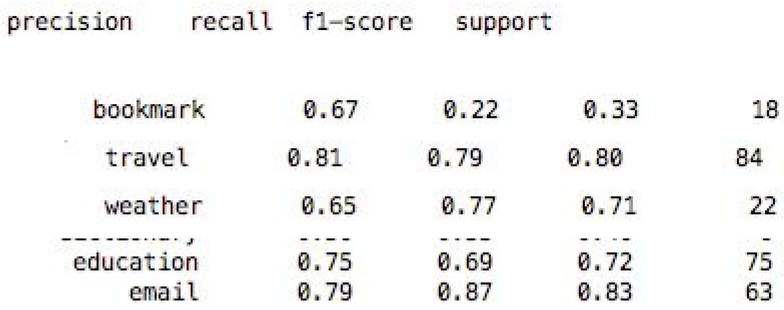
```
('for category : ', 'entertainment', ' correct classified  : ', 141, ' incorrectly classified : ', 31)
('for category : ', u'bookmark', ' correct classified  : ', 11, ' incorrectly classified : ', 17)
('for category : ', 'travel', ' correct classified  : ', 47, ' incorrectly classified : ', 11)
```



```
('for category : ', 'utility', ' correct classified  : ', 101, ' incorrectly classified : ', 10)
('for category : ', 'email', ' correct classified  : ', 54, ' incorrectly classified : ', 9)
```

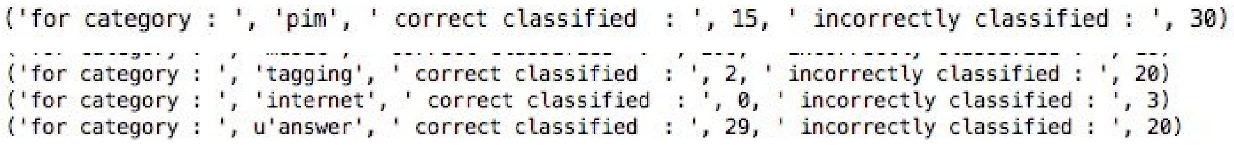
In the above screenshot we can see that, when compared to the SVM classifier, there is a slight deviation in the prediction. The categories entertainment, travel, utility which give a fairly good prediction for the test data using SVM classifier are slightly differing in the Random Forest classifier. For example in case of bookmark, 23 are correctly classified and only 5 are incorrectly classified in case of SVM classifier whereas when I use Random Forest, 11 are correctly classified and 17 is incorrectly classified. However travel and utility which are fairly obvious from their meaning is correctly classified and gives similar performance as compared to SVM classifier. This is because SVM works well for text classification and understands semantic similarity between words in a much efficient way as compared to Random Forest.

To further support my analysis, I have determined the precision,recall and f-1 score for the categories. Since the number of correctly and incorrectly classified does not talk about false positives and false negatives, I have used precision and recall measure to analyse the misses and false alarms for the categories.



```
precision    recall  f1-score   support

      bookmark       0.67      0.22      0.33        18
        travel       0.81      0.79      0.80        84
       weather       0.65      0.77      0.71        22
     education       0.75      0.69      0.72        75
         email       0.79      0.87      0.83        63
```

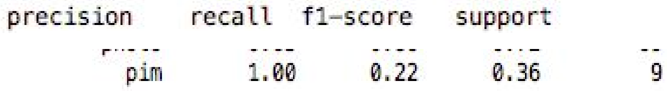
Higher the precision, the higher is the change of eliminating false positives. Higher the recall value, higher is the chance of eliminating false negatives. As compared to SVM classifier, random forest classifier has a slight deviation while eliminating false positives and false negatives. For example, bookmark category has a precision of 0.87, recall of 0.72 in case of SVM classifier and has a precision of 0.67 and 0.22 in case of Random Forest classifier. Hence even though the number of correctly classified and incorrectly classified is pretty close, there is a considerable deviation while computing false positives and false negatives. Therefore, from these values, we can conclude that in case of Random Forest classifier, there are still a large number of false positives and false negatives as compared to an SVM classifier.



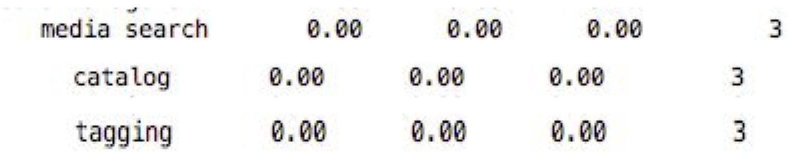
```
('for category : ', 'pim', ' correct classified  : ', 15, ' incorrectly classified : ', 30)
('for category : ', 'tagging', ' correct classified  : ', 2, ' incorrectly classified : ', 20)
('for category : ', 'internet', ' correct classified  : ', 0, ' incorrectly classified : ', 3)
('for category : ', u'answer', ' correct classified  : ', 29, ' incorrectly classified : ', 20)
```

On the other hand, the categories like pim, tagging, answer give worse results compared to SVM classifier. There are two reasons behind this. First is the category itself is ambiguous and not clear with the name. The second is text classification is not good with random forest classifier since it fails to understand semantic similarity between the words. Due to this, for example tagging which correctly classifies 9 and incorrectly classified 13 in SVM has only 2 correctly classified and 20 incorrectly classified using Random Forest. This is because SVM works well for text classification unlike random forest. Though random forest is an ensemble, random forest fails to give a better accuracy as compared to SVM classifier.

To further support my analysis, I have determined the precision,recall and f-1 score for the categories. Since the number of correctly and incorrectly classified does not talk about false positives and false negatives, I have used precision and recall measure to analyse the misses and false alarms for the categories.



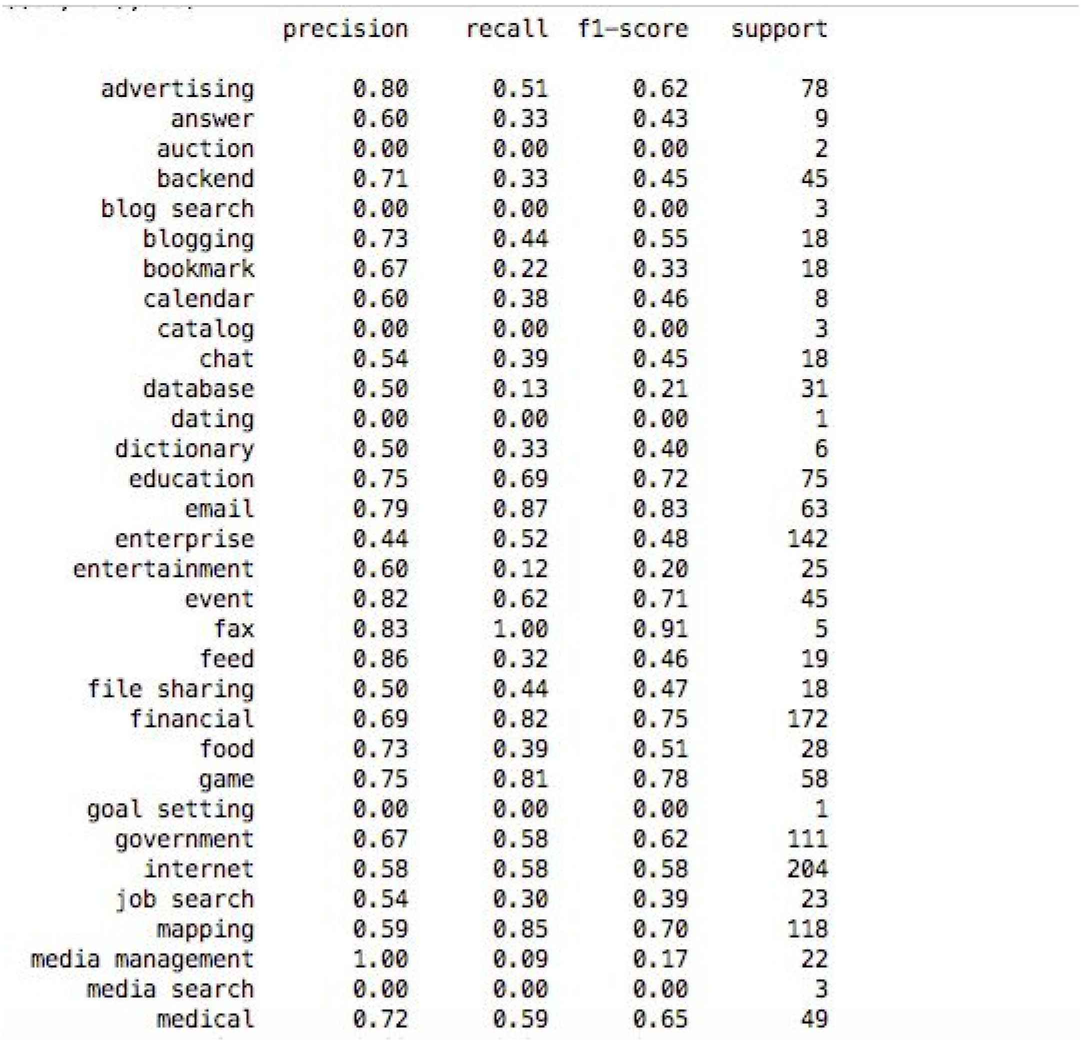
```
precision    recall  f1-score   support

          pim       1.00      0.22      0.36         9
```



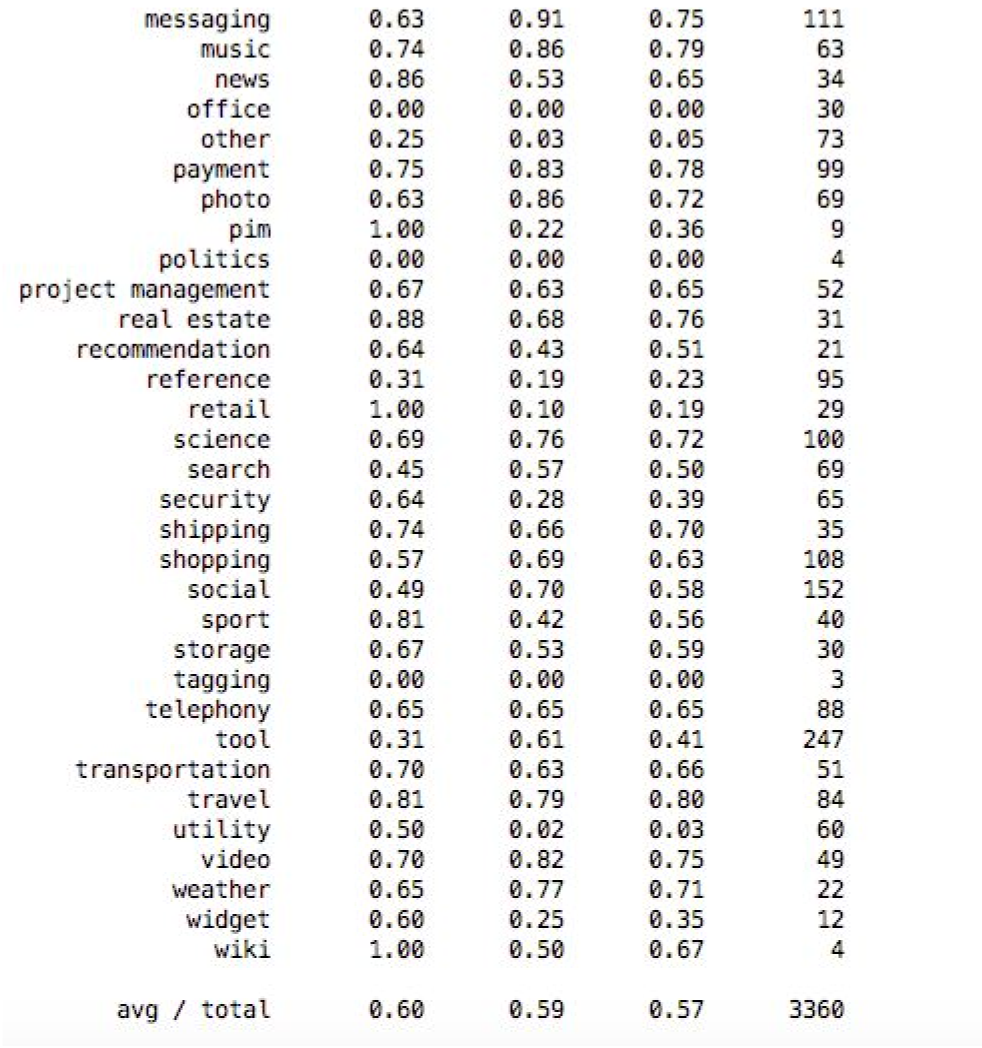
```
   media search       0.00      0.00      0.00         3
       catalog       0.00      0.00      0.00         3
       tagging       0.00      0.00      0.00         3
```

Lower the precision, the lower is the change of eliminating false positives. Lower the recall value, lower is the chance of eliminating false negatives.As compared to SVM classifier, random forest classifier has a visible deviation. For example, tagging category has a precision of 0.33, recall of 0.33 in case of SVM classifier and has a precision of 0.00 and recall of 0.00 in case of Random Forest classifier. Therefore, from these values, we can conclude that in case of Random Forest classifier does not eliminate the false positives and false negatives for a ambiguous category whereas SVM does a fairly decent job of eliminating these even though the category is ambiguous.

The precision, recall and f-1 score for all the categories is as follows:

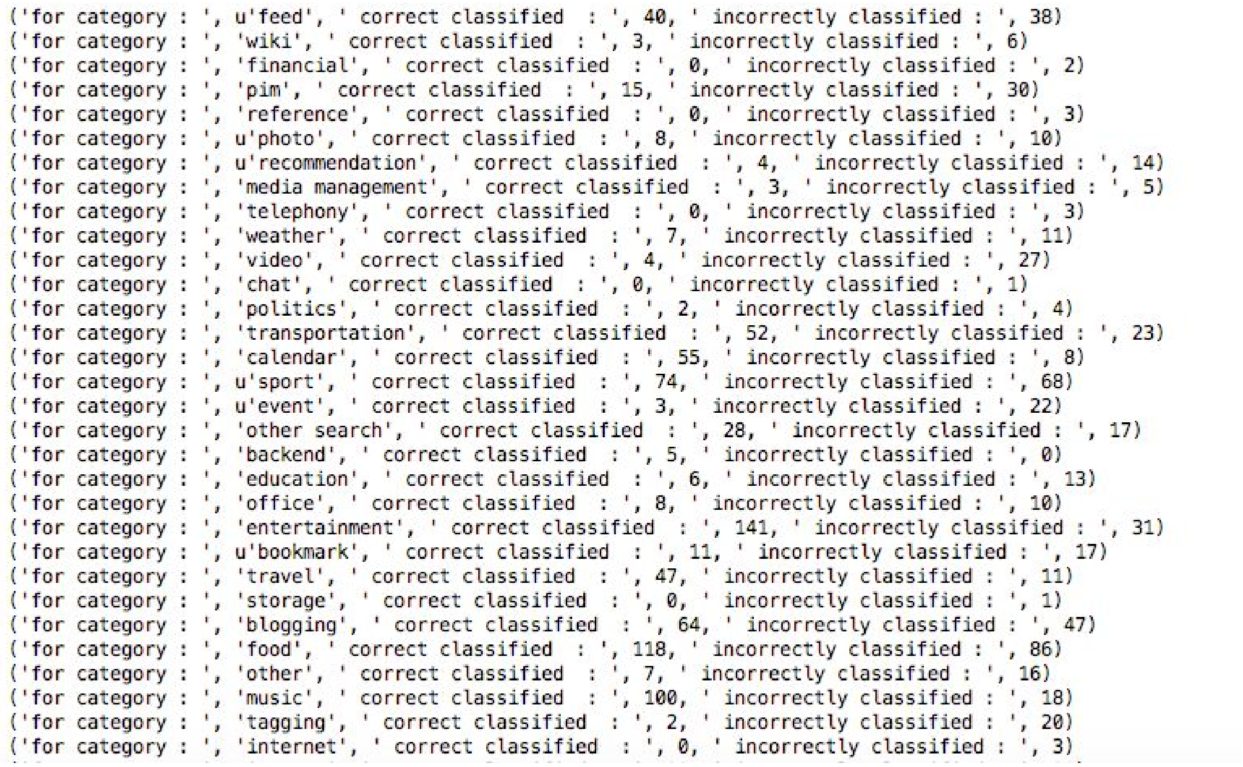


```
                  precision    recall  f1-score   support

      advertising       0.80      0.51      0.62        78
           answer       0.60      0.33      0.43         9
          auction       0.00      0.00      0.00         2
          backend       0.71      0.33      0.45        45
      blog search       0.00      0.00      0.00         3
         blogging       0.73      0.44      0.55        18
         bookmark       0.67      0.22      0.33        18
         calendar       0.60      0.38      0.46         8
          catalog       0.00      0.00      0.00         3
             chat       0.54      0.39      0.45        18
         database       0.50      0.13      0.21        31
           dating       0.00      0.00      0.00         1
       dictionary       0.50      0.33      0.40         6
        education       0.75      0.69      0.72        75
            email       0.79      0.87      0.83        63
       enterprise       0.44      0.52      0.48       142
    entertainment       0.60      0.12      0.20        25
            event       0.82      0.62      0.71        45
              fax       0.83      1.00      0.91         5
             feed       0.86      0.32      0.46        19
     file sharing       0.50      0.44      0.47        18
        financial       0.69      0.82      0.75       172
             food       0.73      0.39      0.51        28
             game       0.75      0.81      0.78        58
     goal setting       0.00      0.00      0.00         1
       government       0.67      0.58      0.62       111
         internet       0.58      0.58      0.58       204
       job search       0.54      0.30      0.39        23
          mapping       0.59      0.85      0.70       118
 media management       1.00      0.09      0.17        22
     media search       0.00      0.00      0.00         3
          medical       0.72      0.59      0.65        49
```

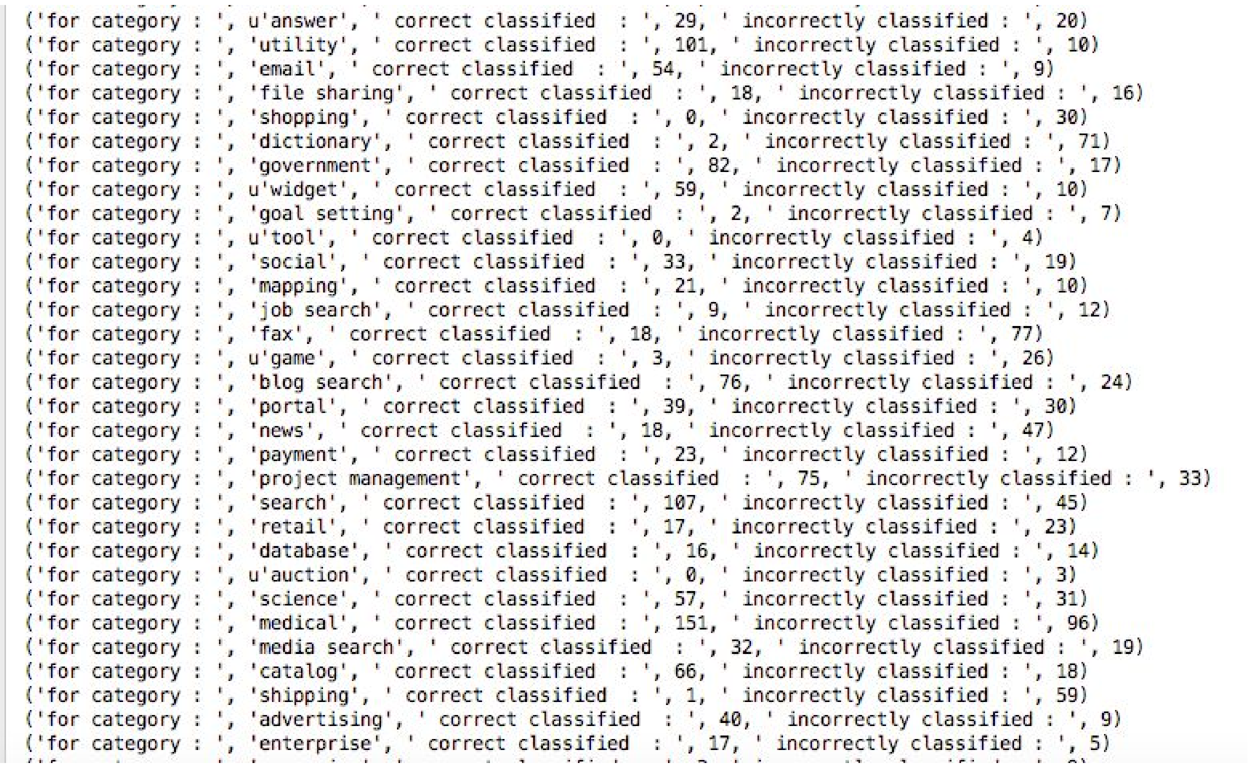


```
          messaging       0.63      0.91      0.75       111
              music       0.74      0.86      0.79        63
               news       0.86      0.53      0.65        34
             office       0.00      0.00      0.00        30
              other       0.25      0.03      0.05        73
            payment       0.75      0.83      0.78        99
              photo       0.63      0.86      0.72        69
                pim       1.00      0.22      0.36         9
           politics       0.00      0.00      0.00         4
 project management       0.67      0.63      0.65        52
        real estate       0.88      0.68      0.76        31
     recommendation       0.64      0.43      0.51        21
          reference       0.31      0.19      0.23        95
             retail       1.00      0.10      0.19        29
            science       0.69      0.76      0.72       100
             search       0.45      0.57      0.50        69
           security       0.64      0.28      0.39        65
           shipping       0.74      0.66      0.70        35
           shopping       0.57      0.69      0.63       108
             social       0.49      0.70      0.58       152
              sport       0.81      0.42      0.56        40
            storage       0.67      0.53      0.59        30
            tagging       0.00      0.00      0.00         3
          telephony       0.65      0.65      0.65        88
               tool       0.31      0.61      0.41       247
     transportation       0.70      0.63      0.66        51
             travel       0.81      0.79      0.80        84
            utility       0.50      0.02      0.03        60
              video       0.70      0.82      0.75        49
            weather       0.65      0.77      0.71        22
             widget       0.60      0.25      0.35        12
               wiki       1.00      0.50      0.67         4

        avg / total       0.60      0.59      0.57      3360
```

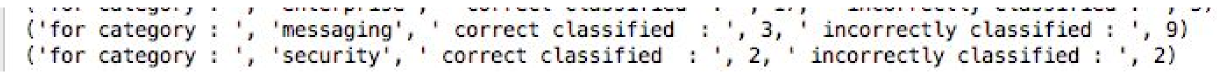
The count of correctly and incorrectly classified for all the categories is as follows:



```
('for category : ', u'feed', ' correct classified  : ', 40, ' incorrectly classified : ', 38)
('for category : ', 'wiki', ' correct classified  : ', 3, ' incorrectly classified : ', 6)
('for category : ', 'financial', ' correct classified  : ', 0, ' incorrectly classified : ', 2)
('for category : ', 'pim', ' correct classified  : ', 15, ' incorrectly classified : ', 30)
('for category : ', 'reference', ' correct classified  : ', 0, ' incorrectly classified : ', 3)
('for category : ', u'photo', ' correct classified  : ', 8, ' incorrectly classified : ', 10)
('for category : ', u'recommendation', ' correct classified  : ', 4, ' incorrectly classified : ', 14)
('for category : ', 'media management', ' correct classified  : ', 3, ' incorrectly classified : ', 5)
('for category : ', 'telephony', ' correct classified  : ', 0, ' incorrectly classified : ', 3)
('for category : ', 'weather', ' correct classified  : ', 7, ' incorrectly classified : ', 11)
('for category : ', 'video', ' correct classified  : ', 4, ' incorrectly classified : ', 27)
('for category : ', 'chat', ' correct classified  : ', 0, ' incorrectly classified : ', 1)
('for category : ', 'politics', ' correct classified  : ', 2, ' incorrectly classified : ', 4)
('for category : ', 'transportation', ' correct classified  : ', 52, ' incorrectly classified : ', 23)
('for category : ', 'calendar', ' correct classified  : ', 55, ' incorrectly classified : ', 8)
('for category : ', u'sport', ' correct classified  : ', 74, ' incorrectly classified : ', 68)
('for category : ', u'event', ' correct classified  : ', 3, ' incorrectly classified : ', 22)
('for category : ', 'other search', ' correct classified  : ', 28, ' incorrectly classified : ', 17)
('for category : ', 'backend', ' correct classified  : ', 5, ' incorrectly classified : ', 0)
('for category : ', 'education', ' correct classified  : ', 6, ' incorrectly classified : ', 13)
('for category : ', 'office', ' correct classified  : ', 8, ' incorrectly classified : ', 10)
('for category : ', 'entertainment', ' correct classified  : ', 141, ' incorrectly classified : ', 31)
('for category : ', u'bookmark', ' correct classified  : ', 11, ' incorrectly classified : ', 17)
('for category : ', 'travel', ' correct classified  : ', 47, ' incorrectly classified : ', 11)
('for category : ', 'storage', ' correct classified  : ', 0, ' incorrectly classified : ', 1)
('for category : ', 'blogging', ' correct classified  : ', 64, ' incorrectly classified : ', 47)
('for category : ', 'food', ' correct classified  : ', 118, ' incorrectly classified : ', 86)
('for category : ', 'other', ' correct classified  : ', 7, ' incorrectly classified : ', 16)
('for category : ', 'music', ' correct classified  : ', 100, ' incorrectly classified : ', 18)
('for category : ', 'tagging', ' correct classified  : ', 2, ' incorrectly classified : ', 20)
('for category : ', 'internet', ' correct classified  : ', 0, ' incorrectly classified : ', 3)
```



```
('for category : ', u'answer', ' correct classified  : ', 29, ' incorrectly classified : ', 20)
('for category : ', 'utility', ' correct classified  : ', 101, ' incorrectly classified : ', 10)
('for category : ', 'email', ' correct classified  : ', 54, ' incorrectly classified : ', 9)
('for category : ', 'file sharing', ' correct classified  : ', 18, ' incorrectly classified : ', 16)
('for category : ', 'shopping', ' correct classified  : ', 0, ' incorrectly classified : ', 30)
('for category : ', 'dictionary', ' correct classified  : ', 2, ' incorrectly classified : ', 71)
('for category : ', 'government', ' correct classified  : ', 82, ' incorrectly classified : ', 17)
('for category : ', u'widget', ' correct classified  : ', 59, ' incorrectly classified : ', 10)
('for category : ', 'goal setting', ' correct classified  : ', 2, ' incorrectly classified : ', 7)
('for category : ', u'tool', ' correct classified  : ', 0, ' incorrectly classified : ', 4)
('for category : ', 'social', ' correct classified  : ', 33, ' incorrectly classified : ', 19)
('for category : ', 'mapping', ' correct classified  : ', 21, ' incorrectly classified : ', 10)
('for category : ', 'job search', ' correct classified  : ', 9, ' incorrectly classified : ', 12)
('for category : ', 'fax', ' correct classified  : ', 18, ' incorrectly classified : ', 77)
('for category : ', u'game', ' correct classified  : ', 3, ' incorrectly classified : ', 26)
('for category : ', 'blog search', ' correct classified  : ', 76, ' incorrectly classified : ', 24)
('for category : ', 'portal', ' correct classified  : ', 39, ' incorrectly classified : ', 30)
('for category : ', 'news', ' correct classified  : ', 18, ' incorrectly classified : ', 47)
('for category : ', 'payment', ' correct classified  : ', 23, ' incorrectly classified : ', 12)
('for category : ', 'project management', ' correct classified  : ', 75, ' incorrectly classified : ', 33)
('for category : ', 'search', ' correct classified  : ', 107, ' incorrectly classified : ', 45)
('for category : ', 'retail', ' correct classified  : ', 17, ' incorrectly classified : ', 23)
('for category : ', 'database', ' correct classified  : ', 16, ' incorrectly classified : ', 14)
('for category : ', u'auction', ' correct classified  : ', 0, ' incorrectly classified : ', 3)
('for category : ', 'science', ' correct classified  : ', 57, ' incorrectly classified : ', 31)
('for category : ', 'medical', ' correct classified  : ', 151, ' incorrectly classified : ', 96)
('for category : ', 'media search', ' correct classified  : ', 32, ' incorrectly classified : ', 19)
('for category : ', 'catalog', ' correct classified  : ', 66, ' incorrectly classified : ', 18)
('for category : ', 'shipping', ' correct classified  : ', 1, ' incorrectly classified : ', 59)
('for category : ', 'advertising', ' correct classified  : ', 40, ' incorrectly classified : ', 9)
('for category : ', 'enterprise', ' correct classified  : ', 17, ' incorrectly classified : ', 5)
```

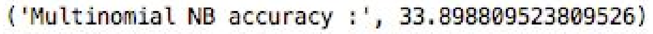


```
('for category : ', 'enterprise', ' correct classified  : ', 17, ' incorrectly classified : ', 5)
```

```
('for category : ', 'messaging', ' correct classified  : ', 3, ' incorrectly classified : ', 9)
('for category : ', 'security', ' correct classified  : ', 2, ' incorrectly classified : ', 2)
```

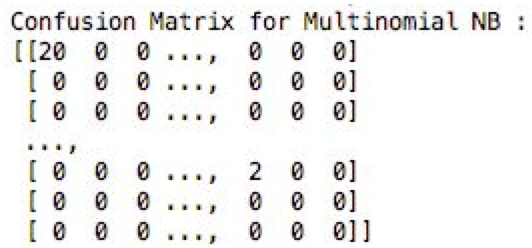
Multinomial Naive Bayes Classifier:

The accuracy for Multinomial Naive Bayes classifier is as follows. The accuracy of Multinomial Naive Bayes is worst as compared SVM classifier and Random Forest classifier.



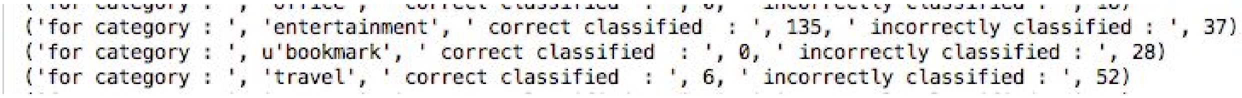
```
('Multinomial NB accuracy :', 33.898809523809526)
```

The confusion matrix for Multinomial Naive Bayes is as follows:

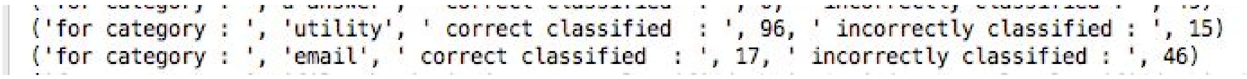


```
Confusion Matrix for Multinomial NB :
[[20  0  0 ...,  0  0  0]
 [ 0  0  0 ...,  0  0  0]
 [ 0  0  0 ...,  0  0  0]
 ...,
 [ 0  0  0 ...,  2  0  0]
 [ 0  0  0 ...,  0  0  0]
 [ 0  0  0 ...,  0  0  0]]
```

The confusion matrix seen for Multinomial Naive Bayes classifier gives a lot of visible deviation in terms of accuracy and classification as compared to SVM classifier and Random Forest classifier . For better understanding of the confusion matrix, I have determined the number of correctly classified and incorrectly classified for each class (for each category)



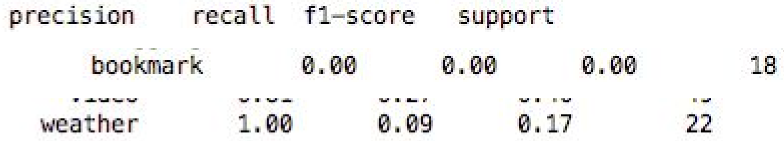
```
('for category : ', 'entertainment', ' correct classified  : ', 135, ' incorrectly classified : ', 37)
('for category : ', u'bookmark', ' correct classified  : ', 0, ' incorrectly classified : ', 28)
('for category : ', 'travel', ' correct classified  : ', 6, ' incorrectly classified : ', 52)
```



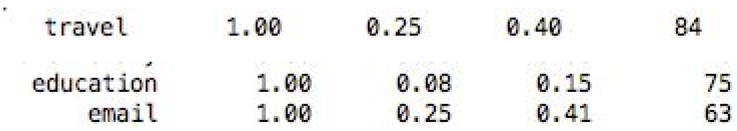
```
('for category : ', 'utility', ' correct classified  : ', 96, ' incorrectly classified : ', 15)
('for category : ', 'email', ' correct classified  : ', 17, ' incorrectly classified : ', 46)
```

In the above screenshot we can see that, when compared to the SVM classifier and Random Forest classifier, there is considerable deviation in the prediction. The categories bookmark, travel, utility which give a fairly good prediction for the test data using SVM classifier and average predictions in Random Forest classifier do not work well with the Multinomial Naive Bayes classifier. For example in case of bookmark, 23 are correctly classified and only 5 are incorrectly classified in case of SVM classifier whereas when I use Random Forest, 11 are correctly classified and 17 is incorrectly classified. However in case of Naive 0 are correctly classified and 28 are incorrectly classified. This is because SVM works well for text classification and understands semantic similarity between words in a much efficient way whereas Random Forest being an ensemble gives fairly good results. However Naive Bayes classifier uses Naive assumptions to make predictions due to which for text data ; it gives very bad results.

To further support my analysis, I have determined the precision,recall and f-1 score for the categories. Since the number of correctly and incorrectly classified does not talk about false positives and false negatives, I have used precision and recall measure to analyse the misses and false alarms for the categories.

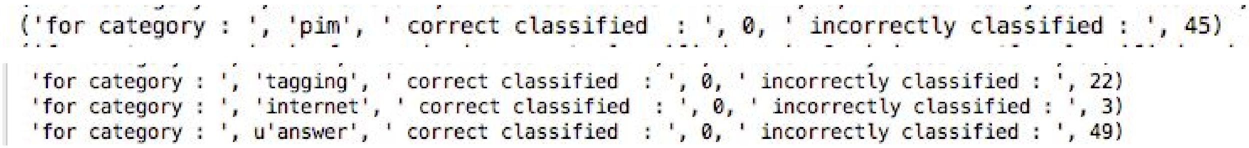


```
precision    recall  f1-score   support

     bookmark       0.00      0.00      0.00        18
  weather       1.00      0.09      0.17        22
```



```
  travel       1.00      0.25      0.40        84

 education       1.00      0.08      0.15        75
     email       1.00      0.25      0.41        63
```

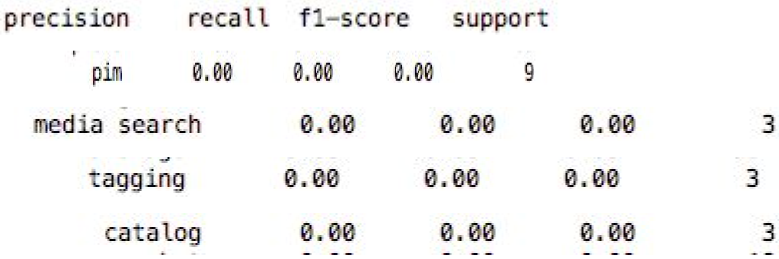
Higher the precision, the higher is the change of eliminating false positives. Higher the recall value, higher is the chance of eliminating false negatives. As compared to SVM classifier and random forest classifier, naive bayes classifier has a lot of difference while eliminating false positives and false negatives. For example, bookmark category has a precision of 0.87, recall of 0.72 in case of SVM classifier and has a precision of 0.67 and 0.22 in case of Random Forest classifier has a precision of 0.00 and 0.00 recall value for Naive Bayes classifier. Therefore, from these values, we can conclude that in case of Naive classifier, there are large number of false positives and false negatives as compared to an SVM and Random forest classifier.



```
('for category : ', 'pim', ' correct classified  : ', 0, ' incorrectly classified : ', 45)

 'for category : ', 'tagging', ' correct classified  : ', 0, ' incorrectly classified : ', 22)
 'for category : ', 'internet', ' correct classified  : ', 0, ' incorrectly classified : ', 3)
 'for category : ', u'answer', ' correct classified  : ', 0, ' incorrectly classified : ', 49)
```

From the above screenshot, the categories like tagging, answer, pim give worse results compared to SVM classifier and Random Forest classifier. There are two reasons behind this. First is the category itself is ambiguous and does not give a detailed understanding about the data in that category. The second is that Naive Bayes classifier makes very naive assumptions at the time of prediction. These Naive assumption gives very bad results when compared to the other two classifiers. Due to this, for example tagging which correctly classified 9 and incorrectly classified 13 in SVM and 2 correctly classified and 20 incorrectly classified using Random Forest has 0 correctly classified and 22 incorrectly classified using Naive Bayes classifier.

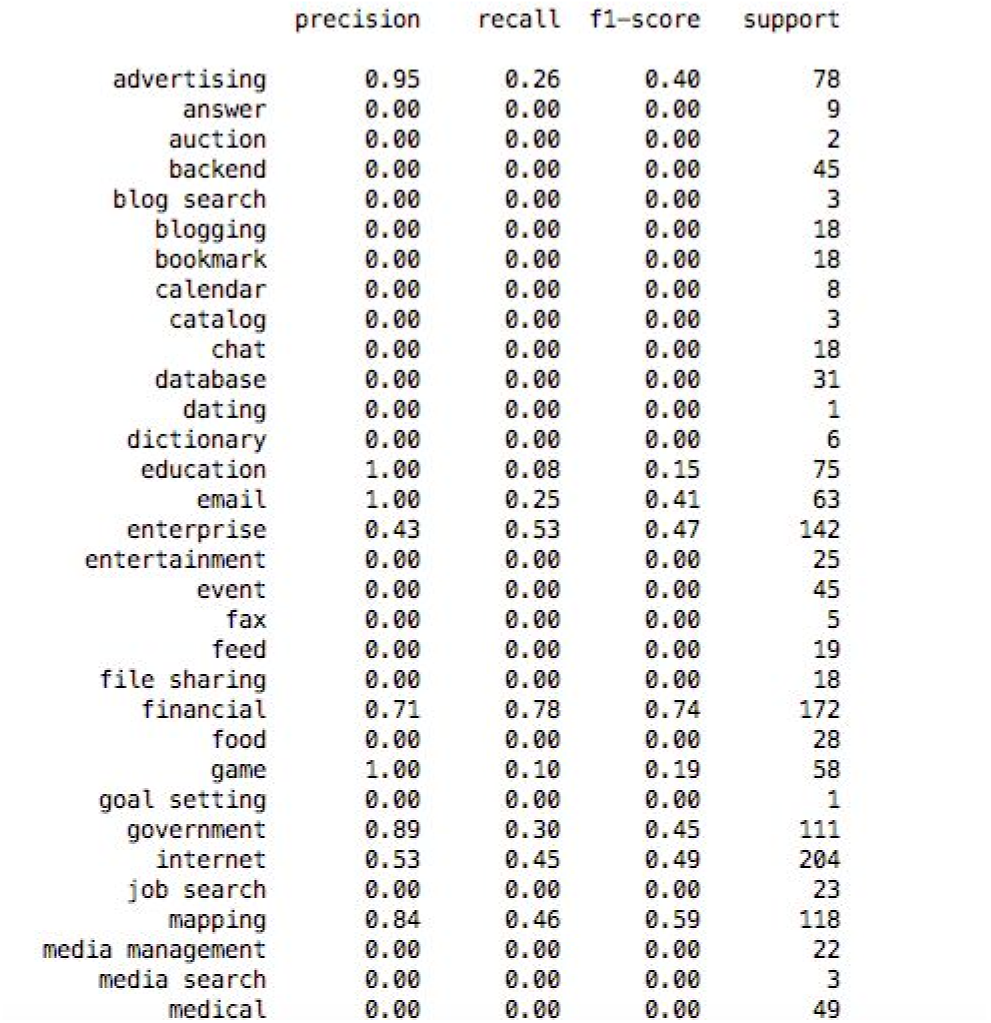
To further support my analysis, I have determined the precision,recall and f-1 score for the categories. Since the number of correctly and incorrectly classified does not talk about false positives and false negatives, I have used precision and recall measure to analyse the misses and false alarms for the categories.



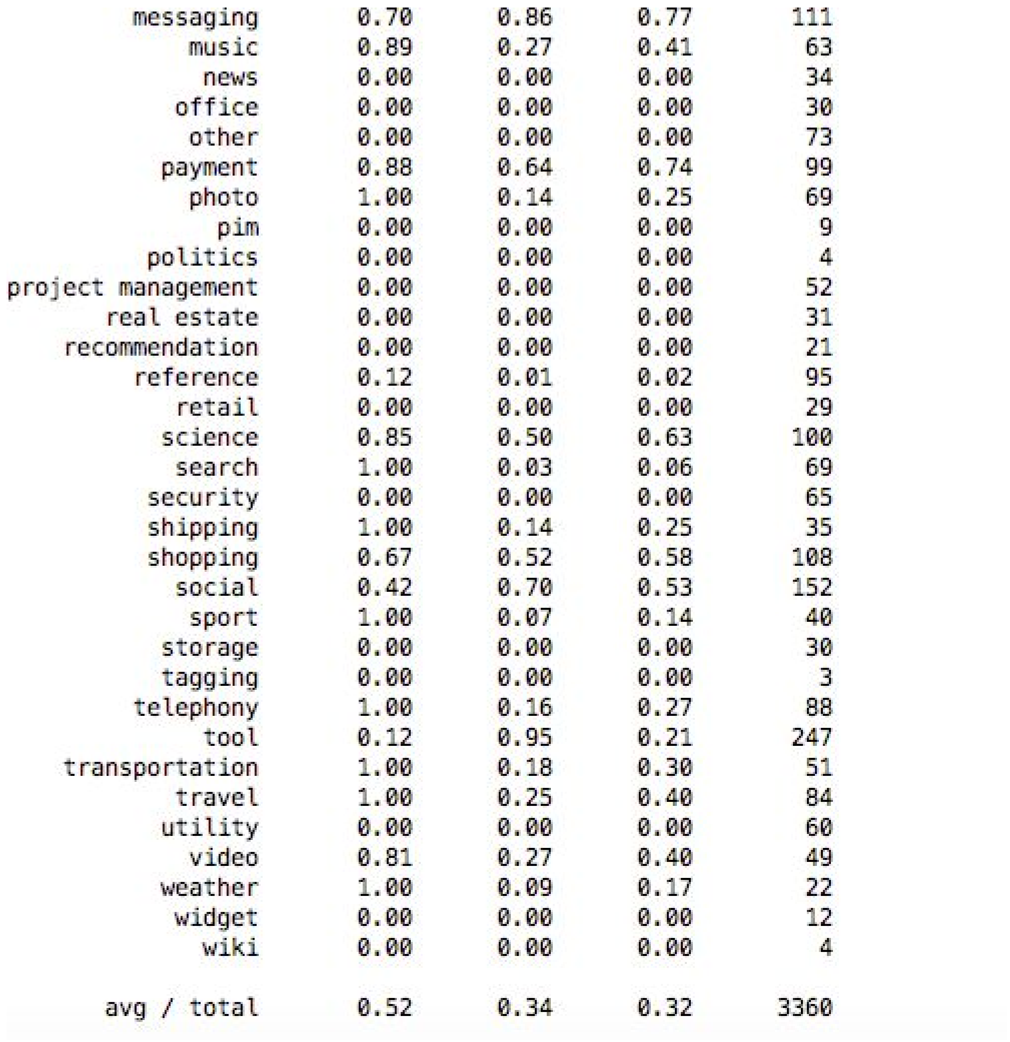
```
precision    recall  f1-score   support

       pim      0.00      0.00      0.00        9

  media search       0.00      0.00      0.00         3

      tagging       0.00      0.00      0.00         3

       catalog       0.00      0.00      0.00         3
```

Lower the precision, the lower is the change of eliminating false positives. Lower the recall value, lower is the chance of eliminating false negatives.As compared to SVM classifier and random forest classifier, naive bayes has a lot of deviation. For example, tagging category has a precision of 0.33, recall of 0.33 in case of SVM classifier and has a precision of 0.00 and recall of 0.00 in case of Random Forest classifier has 0.00 precision and 0.00 recall for random forest classifier. Therefore, from these values, we can conclude that in case of Naive Bayes classifier does not eliminate the false positives and false negatives for a ambiguous category whereas SVM does a fairly decent job of eliminating these even though the category is ambiguous.

The precision,recall and F-1 for all the categories is as follows:

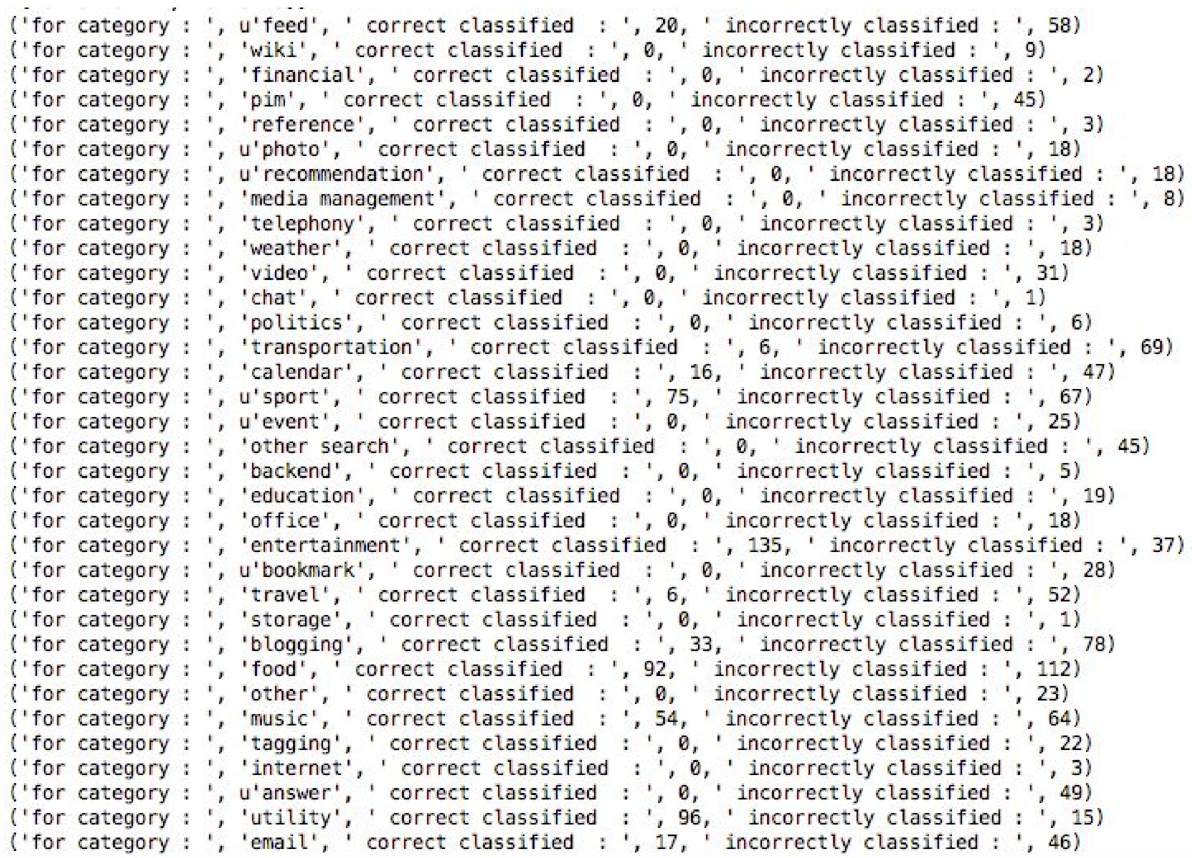


```
                  precision    recall  f1-score   support

     advertising       0.95      0.26      0.40        78
          answer       0.00      0.00      0.00         9
         auction       0.00      0.00      0.00         2
         backend       0.00      0.00      0.00        45
     blog search       0.00      0.00      0.00         3
        blogging       0.00      0.00      0.00        18
        bookmark       0.00      0.00      0.00        18
        calendar       0.00      0.00      0.00         8
         catalog       0.00      0.00      0.00         3
            chat       0.00      0.00      0.00        18
        database       0.00      0.00      0.00        31
          dating       0.00      0.00      0.00         1
      dictionary       0.00      0.00      0.00         6
       education       1.00      0.08      0.15        75
           email       1.00      0.25      0.41        63
      enterprise       0.43      0.53      0.47       142
   entertainment       0.00      0.00      0.00        25
           event       0.00      0.00      0.00        45
             fax       0.00      0.00      0.00         5
            feed       0.00      0.00      0.00        19
    file sharing       0.00      0.00      0.00        18
       financial       0.71      0.78      0.74       172
            food       0.00      0.00      0.00        28
            game       1.00      0.10      0.19        58
    goal setting       0.00      0.00      0.00         1
      government       0.89      0.30      0.45       111
        internet       0.53      0.45      0.49       204
      job search       0.00      0.00      0.00        23
         mapping       0.84      0.46      0.59       118
media management       0.00      0.00      0.00        22
    media search       0.00      0.00      0.00         3
         medical       0.00      0.00      0.00        49
```

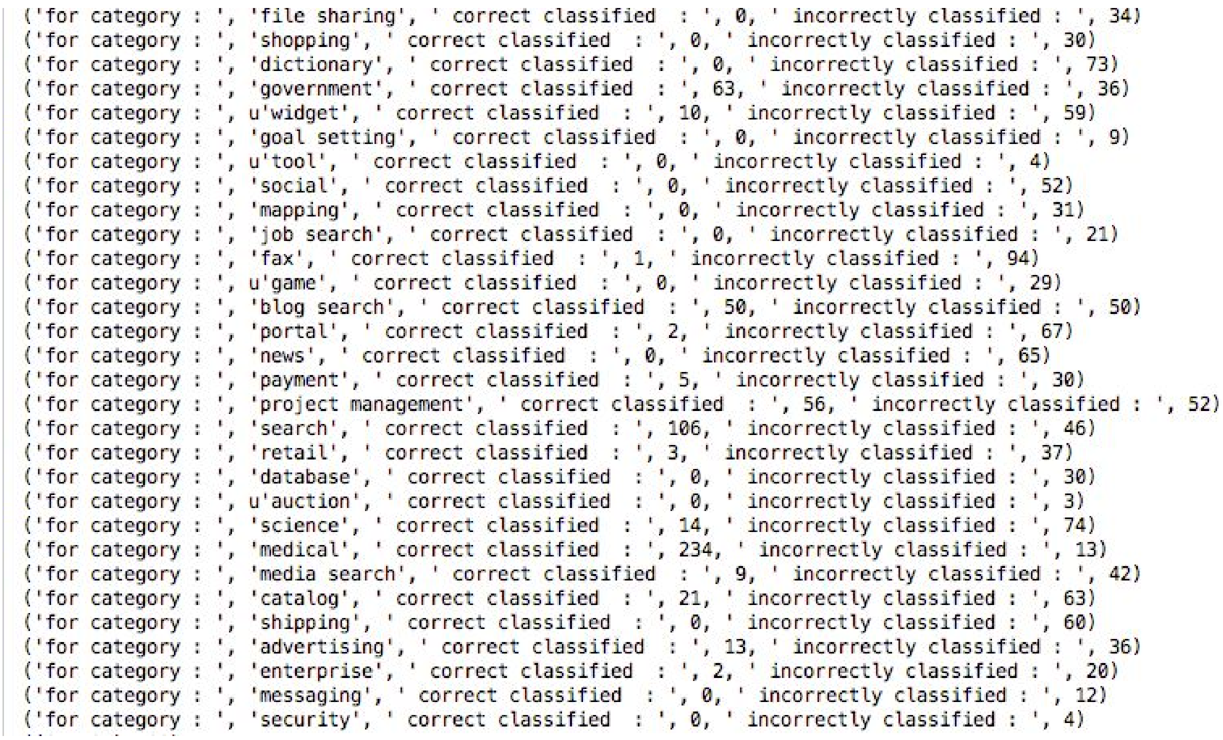


```
         messaging       0.70      0.86      0.77       111
             music       0.89      0.27      0.41        63
              news       0.00      0.00      0.00        34
            office       0.00      0.00      0.00        30
             other       0.00      0.00      0.00        73
           payment       0.88      0.64      0.74        99
             photo       1.00      0.14      0.25        69
               pim       0.00      0.00      0.00         9
          politics       0.00      0.00      0.00         4
project management       0.00      0.00      0.00        52
       real estate       0.00      0.00      0.00        31
    recommendation       0.00      0.00      0.00        21
         reference       0.12      0.01      0.02        95
            retail       0.00      0.00      0.00        29
           science       0.85      0.50      0.63       100
            search       1.00      0.03      0.06        69
          security       0.00      0.00      0.00        65
          shipping       1.00      0.14      0.25        35
          shopping       0.67      0.52      0.58       108
            social       0.42      0.70      0.53       152
             sport       1.00      0.07      0.14        40
           storage       0.00      0.00      0.00        30
           tagging       0.00      0.00      0.00         3
         telephony       1.00      0.16      0.27        88
              tool       0.12      0.95      0.21       247
    transportation       1.00      0.18      0.30        51
            travel       1.00      0.25      0.40        84
           utility       0.00      0.00      0.00        60
             video       0.81      0.27      0.40        49
           weather       1.00      0.09      0.17        22
            widget       0.00      0.00      0.00        12
              wiki       0.00      0.00      0.00         4

       avg / total       0.52      0.34      0.32      3360
```

The number of correctly and incorrectly classified for all the categories is as follows:



```
('for category : ', u'feed', ' correct classified  : ', 20, ' incorrectly classified : ', 58)
('for category : ', 'wiki', ' correct classified  : ', 0, ' incorrectly classified : ', 9)
('for category : ', 'financial', ' correct classified  : ', 0, ' incorrectly classified : ', 2)
('for category : ', 'pim', ' correct classified  : ', 0, ' incorrectly classified : ', 45)
('for category : ', 'reference', ' correct classified  : ', 0, ' incorrectly classified : ', 3)
('for category : ', u'photo', ' correct classified  : ', 0, ' incorrectly classified : ', 18)
('for category : ', u'recommendation', ' correct classified  : ', 0, ' incorrectly classified : ', 18)
('for category : ', 'media management', ' correct classified  : ', 0, ' incorrectly classified : ', 8)
('for category : ', 'telephony', ' correct classified  : ', 0, ' incorrectly classified : ', 3)
('for category : ', 'weather', ' correct classified  : ', 0, ' incorrectly classified : ', 18)
('for category : ', 'video', ' correct classified  : ', 0, ' incorrectly classified : ', 31)
('for category : ', 'chat', ' correct classified  : ', 0, ' incorrectly classified : ', 1)
('for category : ', 'politics', ' correct classified  : ', 0, ' incorrectly classified : ', 6)
('for category : ', 'transportation', ' correct classified  : ', 6, ' incorrectly classified : ', 69)
('for category : ', 'calendar', ' correct classified  : ', 16, ' incorrectly classified : ', 47)
('for category : ', u'sport', ' correct classified  : ', 75, ' incorrectly classified : ', 67)
('for category : ', u'event', ' correct classified  : ', 0, ' incorrectly classified : ', 25)
('for category : ', 'other search', ' correct classified  : ', 0, ' incorrectly classified : ', 45)
('for category : ', 'backend', ' correct classified  : ', 0, ' incorrectly classified : ', 5)
('for category : ', 'education', ' correct classified  : ', 0, ' incorrectly classified : ', 19)
('for category : ', 'office', ' correct classified  : ', 0, ' incorrectly classified : ', 18)
('for category : ', 'entertainment', ' correct classified  : ', 135, ' incorrectly classified : ', 37)
('for category : ', u'bookmark', ' correct classified  : ', 0, ' incorrectly classified : ', 28)
('for category : ', 'travel', ' correct classified  : ', 6, ' incorrectly classified : ', 52)
('for category : ', 'storage', ' correct classified  : ', 0, ' incorrectly classified : ', 1)
('for category : ', 'blogging', ' correct classified  : ', 33, ' incorrectly classified : ', 78)
('for category : ', 'food', ' correct classified  : ', 92, ' incorrectly classified : ', 112)
('for category : ', 'other', ' correct classified  : ', 0, ' incorrectly classified : ', 23)
('for category : ', 'music', ' correct classified  : ', 54, ' incorrectly classified : ', 64)
('for category : ', 'tagging', ' correct classified  : ', 0, ' incorrectly classified : ', 22)
('for category : ', 'internet', ' correct classified  : ', 0, ' incorrectly classified : ', 3)
('for category : ', u'answer', ' correct classified  : ', 0, ' incorrectly classified : ', 49)
('for category : ', 'utility', ' correct classified  : ', 96, ' incorrectly classified : ', 15)
('for category : ', 'email', ' correct classified  : ', 17, ' incorrectly classified : ', 46)
```



```
('for category : ', 'file sharing', ' correct classified  : ', 0, ' incorrectly classified : ', 34)
('for category : ', 'shopping', ' correct classified  : ', 0, ' incorrectly classified : ', 30)
('for category : ', 'dictionary', ' correct classified  : ', 0, ' incorrectly classified : ', 73)
('for category : ', 'government', ' correct classified  : ', 63, ' incorrectly classified : ', 36)
('for category : ', u'widget', ' correct classified  : ', 10, ' incorrectly classified : ', 59)
('for category : ', 'goal setting', ' correct classified  : ', 0, ' incorrectly classified : ', 9)
('for category : ', u'tool', ' correct classified  : ', 0, ' incorrectly classified : ', 4)
('for category : ', 'social', ' correct classified  : ', 0, ' incorrectly classified : ', 52)
('for category : ', 'mapping', ' correct classified  : ', 0, ' incorrectly classified : ', 31)
('for category : ', 'job search', ' correct classified  : ', 0, ' incorrectly classified : ', 21)
('for category : ', 'fax', ' correct classified  : ', 1, ' incorrectly classified : ', 94)
('for category : ', u'game', ' correct classified  : ', 0, ' incorrectly classified : ', 29)
('for category : ', 'blog search', ' correct classified  : ', 50, ' incorrectly classified : ', 50)
('for category : ', 'portal', ' correct classified  : ', 2, ' incorrectly classified : ', 67)
('for category : ', 'news', ' correct classified  : ', 0, ' incorrectly classified : ', 65)
('for category : ', 'payment', ' correct classified  : ', 5, ' incorrectly classified : ', 30)
('for category : ', 'project management', ' correct classified  : ', 56, ' incorrectly classified : ', 52)
('for category : ', 'search', ' correct classified  : ', 106, ' incorrectly classified : ', 46)
('for category : ', 'retail', ' correct classified  : ', 3, ' incorrectly classified : ', 37)
('for category : ', 'database', ' correct classified  : ', 0, ' incorrectly classified : ', 30)
('for category : ', u'auction', ' correct classified  : ', 0, ' incorrectly classified : ', 3)
('for category : ', 'science', ' correct classified  : ', 14, ' incorrectly classified : ', 74)
('for category : ', 'medical', ' correct classified  : ', 234, ' incorrectly classified : ', 13)
('for category : ', 'media search', ' correct classified  : ', 9, ' incorrectly classified : ', 42)
('for category : ', 'catalog', ' correct classified  : ', 21, ' incorrectly classified : ', 63)
('for category : ', 'shipping', ' correct classified  : ', 0, ' incorrectly classified : ', 60)
('for category : ', 'advertising', ' correct classified  : ', 13, ' incorrectly classified : ', 36)
('for category : ', 'enterprise', ' correct classified  : ', 2, ' incorrectly classified : ', 20)
('for category : ', 'messaging', ' correct classified  : ', 0, ' incorrectly classified : ', 12)
('for category : ', 'security', ' correct classified  : ', 0, ' incorrectly classified : ', 4)
```

To conclude, SVM classifier works best for text classification as compared to Random Forest and Naive Bayes. The term best is based on the number of correctly and incorrectly classified as well as how well it eliminates false positives and false negatives to a greater extent as compared to Random Forest and Naive Bayes.

CHALLENGES AND LEARNINGS:

From this assignment, I learnt how to use text mining techniques using nltk package to perform tokenization on text data which extracted important features to pass to the classification model. I also learn how to use Sklearn to build different classification models and build confusion matrix and compute accuracy using the package.

The challenges I faced in this assignment is to select the classifier and extract the best features

CONCLUSION:

In this way, tokenization, model fitting,prediction and verification is achieved.