**Data Mining Course Project**

**What’s Cooking**

Made By- Mentored By-

1.)Akash Gangwar(170067) Prof. Faiz Hamid

2.)Aditya Rai(170053)

|  |  |  |
| --- | --- | --- |
| Index | | |
| S.No. | Topic | Page No. |
| 1 | Problem Statement | 1 |
| 2 | Data Understanding | 1 |
| 3 | Data Pre-Processing | 2 |
| 4 | Data Mining Algorithms | 4 |
| 5 | Performance Comparison | 6 |
| 6 | Interpretation of results | 7 |
| 7 | Recommendations | 7 |

**1.)Problem Statement**

The training dataset consists of different recipes distinguished by Recipe\_id. Each recipe has a list of ingredients and the cuisine which it belongs to. The objective is to develop a model from the given dataset which classifies a recipe’s cuisine given its ingredients.

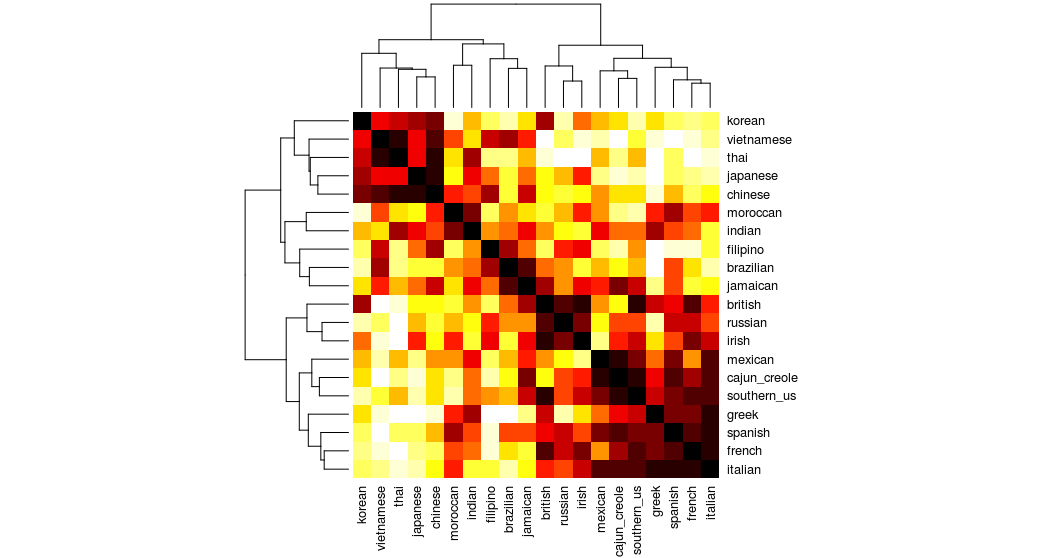
**2.)Data**  **Understanding**

The dataset is in JSON format. The dependent variable is cuisine and independent variable is ingredients.

* Train data consists of 39774 dishes as dish-ids, ingredients and the cuisine they belong to.
* Test data consists of 9944 dishes as dish-ids and ingredients.
* There are 20 different unique cuisines present in the data
* There are total 428275 number of ingredients among which 6703 are unique

The dataset does not have many attributes.In our dataset the main attribute is a list of set of ingredients.Each set represents a dish.The data is basically categorical data and data types is list of strings

**2.1)Data Visualization:** To gain an insight of data we created various plots whose drive links are given below:-  
 1.)[Plot of Cuisines with the count of no Ingredients present in a given cuisine.](https://drive.google.com/file/d/1PQGqMuwVwQWVbUYESpRZ821k8NM0fG2D/view?usp=sharing)  
 2.)[Plot of distribution of recipes in each cuisine.](https://drive.google.com/file/d/1s9Y1WBnNrzZIzr7MEUD2rIsBpUeYiHgp/view?usp=sharing)  
 3.)[Histogram of Recipes length with respect to Ingredients.](https://drive.google.com/file/d/1CYZWh45g2f5m6iH6dq_s0TBl50FNOqbi/view?usp=sharing)  
 4.)[Plot of most common 20 Ingredients.](https://drive.google.com/file/d/1INdy6laZ4a5cf6btk9k7wy_QuJUiySKk/view?usp=sharing)   
 5.) [Plot of number of unique ingredients used in each cuisine](https://drive.google.com/file/d/1es-9tF1D6ceaYtwRDw6Yzfb2lFVUaEku/view?usp=sharing)   
 6.)[Features Importance plots of Ingredients based on TF-IDF measure](https://drive.google.com/file/d/18rDcOK6g9T_uyGrSdIZVFtt1ZyFukt7c/view?usp=sharing)  
 7.)[Plot of some important ingredient with all cuisines](https://drive.google.com/file/d/1Zej2NFwXWht3R3HeiasGqYbNhqnkCVeG/view?usp=sharing)   
  
To check the similarity and correlation between different Cuisines we performed word embedding using Glove vectors .We performed training on aggregated global word to word co-occurrence statistics from a corpus, and the resulting representations showcase the interesting linear substructures of the word vector space.Correlation heat map obtained looks like this-



Similarly we also used a t-SNE to reduce the data set obtained from Glove to two attributes and plotted the similarity of different cuisines which can be seen by [Clicking Here](https://drive.google.com/file/d/1QtvX9PZOqv6Z9xcosqt0KBZP_0iGtFBo/view?usp=sharing).

**3.)Data Preprocessing**

**3.1)Data Cleaning:**

The given dataset contain some quirks and needs to be cleaned.We took the following steps in Data Cleaning-

a).Checked the presence of missing values which came out to be NIL.

b).The incoming data can be loosely structured, multilingual, textual or might have poor spelling for example:

* Some ingredients may contain special symbols that are not relevant like @,# etc and are needed to be removed.
* Since text cases doesn’t change the meaning of the word so its better to convert them into single lower case.
* Punctuation elements like “ “ , - etc are not useful and shall be removed .For ex, ‘Chilli Flakes’ and chilli-flakes are same.
* Stopwords are those words which add no value .They don’t describe any sentiment. Examples are ‘i’,’me’,’myself’,’they’,’them’ and many more. Hence, these words should be removed if present.
* Stemming means bringing a word back to its roots. It is generally used for words which are similar but only differ by tenses.To treat word and word derivatives as same we stemmed our dataset.

c).Checked and removed duplicate data which resulted in numerosity reduction.  
d).Set the ingredient data to unique for each row.

Ideally, we have done a Data cleaning such that it returned a more relevant version of these ingredients.  
**3.2)Data Reduction and Transformation :**

**a.)**Combined the ingredients of a row to form a text document which gave a data set of about 39,774 documents with a class label attached to it.  
**b.)**Removed the Duplicate rows present in Data which resulted in a little of numerosity Reduction.

Data Reduced: 39774 to 39256

**c.)Document Term Matrix:**

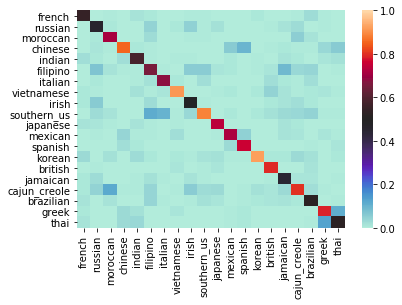
* From the text data (set of ingredients), we have created document term matrix (DTM) which assigns 1 to a ingredients present in a dish and 0 to all those not present in the dish.
* ‘Since DTM was a Sparse matrix so we tried to reduce it’s sparseness by removing the ingredients with frequency less than 3 in whole dataset.

**d.)TF-IDF Matrix:**

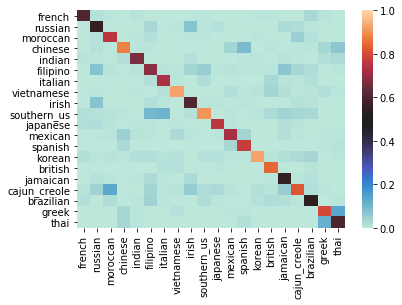
* This is a technique to quantify a word in documents, The weights to each word are computed which signifies the importance of the word in the document and corpus with respect to other documents.
* It gives more weight to those ingredients which occurs more often in one class of documents given it is less present in other classes of documents.
* From the DTM matrix we get the tf(Term frequency) and IDF(Inverse Document frequency) of each word in a document and multiplied them to obtain the TF-IDF value of a word.
* We assigned these tf-idf value to each term in document and formed a matrix.
* Reduced sparse number of columns from 6703 ingredients to 3010.

The TF-IDF matrix will be the input to most of our Models except in Naive Bayes where DTM is the training data.

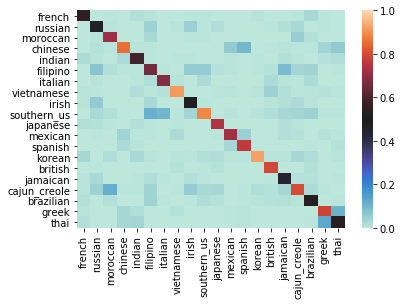
**4.)Data Mining Algorithms Applied**For training the Models and to check its performance we split the training data into two parts,80% of the train data was used for training and rest 20% was used for it’s validation.With this division of the dataset the following DM algorithms were applied:

**4.1.)Artificial Neural Networks:**We build a four layered ANN with an output layer having 20 nodes each representing a unique class.The input layer consist of 3010 input dimensions and weights were assigned using ‘he\_normal’ kernel initializer for further layers.There are 2 hidden layers each with 1000 node units and the activation function used for them is “relu”.The activation function used for the output layer containing 20 nodes is “softmax” .In training the input is feed in batch of 512 with maximum iterations set at 20 for the training data with validation split of 0.1. The Confusion Matrix(Plotted on Heat Map) and other evaluation Metrics which are given below

|  |  |
| --- | --- |
| **Metrics** | **Measure(%)** |
| **Accuracy** | 80.20 |
| **Precision** | 76 |
| **Recall** | 71 |
| **F1-Score** | 73 |
| **Support** | 9944 |

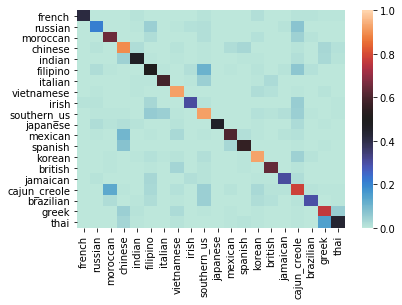
**4.2.)Support Vector Machine:**We built two SVM models using the LINEAR and RBF Kernels separately.Similar to ANN we trained our Model on 80% of the training data and evaluated the Model on the remaining data . Here also we evaluated our both classifiers on the basis of Confusion Matrix and other metrics which are given below.

|  |  |  |
| --- | --- | --- |
| **Metrics** | **RBF(%)** | **Linear(%)** |
| **Accuracy** | 82 | 79.05 |
| **Precision** | 78 | 78 |
| **Recall** | 73 | 69 |
| **F1-Score** | 76 | 72 |
| **Support** | 9944 | 9944 |

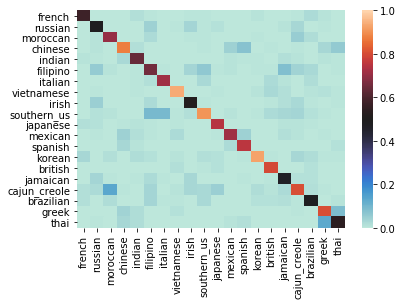
**4.3.)Logistic Regression:**Similar to ANN and SVM here also we trained the Logistic Regression with ‘C=20’ model on 80% of training and evaluated the performance of model on the remaining 20% . The confusion matrix heat-map and other measures are given below.

|  |  |
| --- | --- |
| **Metrics** | **Measure(%)** |
| **Accuracy** | 79.41 |
| **Precision** | 76 |
| **Recall** | 69 |
| **F1-Score** | 72 |
| **Support** | 9944 |

**4.4.)Random Forest:**A single decision tree tends to over-fit the model.Random Forest basically improves the accuracy of decision tree since the ensemble of weak learner predict far better than a single decision tree.For our given set of input values from tf-idf matrix.we created Random forest with 10 number of features and n\_esimators set as 600.Here also the training will be done on 80% and validation on the remaining 20% of train data. The confusion matrix and other measures were plotted and calculated for our split train data.



|  |  |
| --- | --- |
| **Metrics** | **Measure** |
| **Accuracy** | 75.4 |
| **Precision** | 81 |
| **Recall** | 59 |
| **F1-Score** | 65 |
| **Support** | 9944 |

**4.5.)Naive Bayes:**Unlike the other models here the input data was a Document Term matrix of training data.From the DTM we formed a probability matrix for each cuisine.The naive bayes classifier predicts the cuisine on the basis of maximum value of probability for our train data set.This classifier gave the accuracy of **82%** on training data

**4.6.)Ensemble Classifier:**In this classifier we ensemble the three classification models into one classifier.The three models that we ensembled were Logistic Regression , SVM(Linear Kernel) and Random forest.The training was done in the same way as above models by a split of 80-20 percent.The confusion matrix Heat-Map and other model evaluation metrics are given below.

|  |  |
| --- | --- |
| **Metrics** | **Measure(%)** |
| **Accuracy** | 80.6 |
| **Precision** | 78 |
| **Recall** | 70 |
| **F1-Score** | 74 |
| **Support** | 9944 |

**5.)Performance Comparison**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Models\Metric** | **Accuracy** | **Precision** | **Recall** | **F1-score** | **Support** |
| **ANN** | 80.20 | 76 | 71 | 73 | 9944 |
| **SVM(RBF)** | 82 | 78 | 73 | 76 | 9944 |
| **Logistic Regression** | 79.41 | 75 | 69 | 72 | 9944 |
| **SVM(Linear)** | 79.45 | 78 | 69 | 72 | 9944 |
| **Random Forest** | 75.4 | 81 | 59 | 65 | 9944 |
| **Ensemble** | 80.6 | 78 | 70 | 74 | 9944 |

If we compare all the metrics of the model we can eliminate Random forest due to its low Recall.Similarly at the same time SVM(Linear) and Logistic Regression has comparable lower Accuracy,Precision,Recall and F1-score as compared to other models.So these three are weak Classifiers so we build an Ensemble of these three models and obtained has a second best Accuracy of all models tried.From all the metrics we can see that **SVM(RBF Kernel)** has the best of all the metrics so it would be our best choice to score our given test data.The second best according to our results was **Artificial Neural Network** as it also has comparable good Accuracy and precision as compared to other Models.So when final Submissions were made on Kaggle, scores were something like this-

**6)Interpretation of Results**

|  |  |
| --- | --- |
| **Models** | **Kaggle Score** |
| **ANN** | 80.752 |
| **SVM(RBF)** | 82.119 |
| **Ensemble** | 79.475 |
| **Naive Bayes** | 80.26 |
| **Logistic Regression** | 78.388 |
| **SVM(Linear Kernel)** | 78.207 |

So as predicted best accuracy was obtained by SVM(RBF) on the Kaggle.The maximum accuracy anyone could get from this data on Kaggle was 83%.So, as compared to the maximum score, SVM(RBF) is very good model.We could have increased the accuracy by including the SVM(RBF) in Ensemble model with ANN but then there are certain time complexity issues.Both Ensemble and SVM have high time complexity in training so including both in one would raise the time complexity exponentially .

**7.)Recommendations:**Since Accuracy could have been increased by ensembling a SVM(RBF) with ANN and other models but that would require a device of higher Computational power and model will also have a high time Complexity.We were short on both of these.But if any professional organisation is well equipped with both of these resources, it should try the Ensemble of SVM(RBF) with ANN and other models.The Model will surely work better than all of them.