Comp 6781
Language Detection System

Description of the program

Programming language:

I have implemented Question 1: (1) and (2) in C#, MySQL while part (3) in C#

- Training Set and Testing Set were both Tab limited, so it was easy to import it in MySQL and run queries on it to get count of the tweets of particular language and the tweets themselves. Finally MySQL was also used to export the Training/Testing tweets corresponding to each language to a text file
- All the clean data was processed using C#

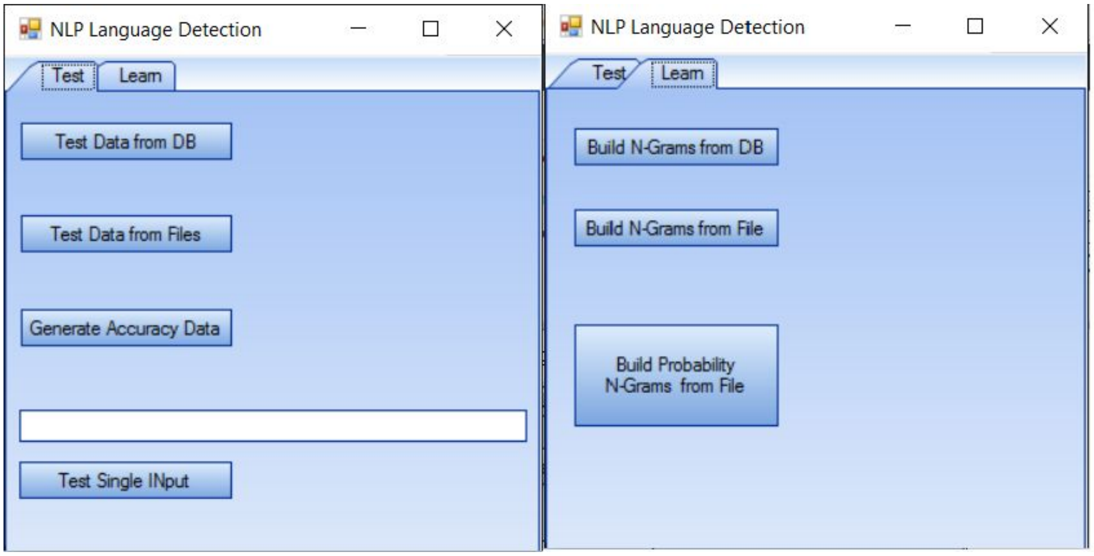
Main data structures: Data structures used in implementing the questions are:

- HashTable
- DataSet
- DataTable
- Array
- List
- Dictionary(sortable Hash Table)

Main Functions Used:

- getTrainingDataFor(Language) : Returns Tweets from a particular file one line at a time.
- getCleanTable(Table) : Cleans the Tweets from Diacritics,Emojis, Spaces,Uppercase.
- GetGram(Table, integer) : Converts a Hash Table into N gram depending on integer(1,2)
- ConvertTableToProbabilityTable(Table,N):Converts a HashTable from frequency to prob.
- applySmoothing(Table, smoothingFactor) :Smoothes a table based on smoothing factor
- ApplyBayesOnUnigram(String) : Takes a string and applies Bayes Reasoning, and returns confidence of each language.

I have developed a Windows Form User Interface named as MainForm.cs in which each button has a separate functionality(some of the buttons will not work because they require MySQL connection or they are intended to be future work)



“**Build Probability N-Gram from File**” (in Learn Tab) button will call btn_probabilityNgramfromFile_Click() Which will read the files from Trainingnlp folder and it will generate **unigramLM.txt** and **bigramLM.txt** which include the first 50 “grams” (i.e. unigram or bigram) along with their unsmoothed probability and their smoothed probability.

“**Test Data from Files**” (in Test Tab) button will call btn_testDatafromFiles_Click() Which will read the files from Testingnlp folder and it will generate **results-unigram.txt** and **results-bigram.txt** which include the tweet number and it’s most likely language.

“**Generate Accuracy Data**” (in Test Tab) button will call btn_generateAccuracyData_Click() Which will generate **analysis-unigram.txt** and **analysis-bigram.txt** which will include
The overall accuracy of the LMs, The accuracy for each language, A confusion matrix indicating the correct classification versus the classification of my system.

Note: In order the System to work, the buttons must be pressed in the above order.

Instructions to Run:

Files/Folders:

- Place the “Trainingnlp” folder on desktop, this folder is used to do part 1 of assignment , it will generate unigramLM.txt and bigramLM.txt on Desktop
- Place the “Testingnlp” folder on desktop, this folder is used to do part 2 of assignment , it will generate results-unigram.txt and results-bigram.txt on Desktop
- Go to A2_27635001\NLP Language Detection Final\NLP Language Detection Final\bin\Debug and run NLP Language Detection Final.exe

Once the User Interface shown above pops up, press:
“**Build Probability N-Gram from File**”wait until MessageDialog Shows task is done then press
“**Test Data from Files**” wait until MessageDialog Shows task is done then press
“**Generate Accuracy Data**” wait until MessageDialog Shows task is done
Finally see the files generated on Desktop

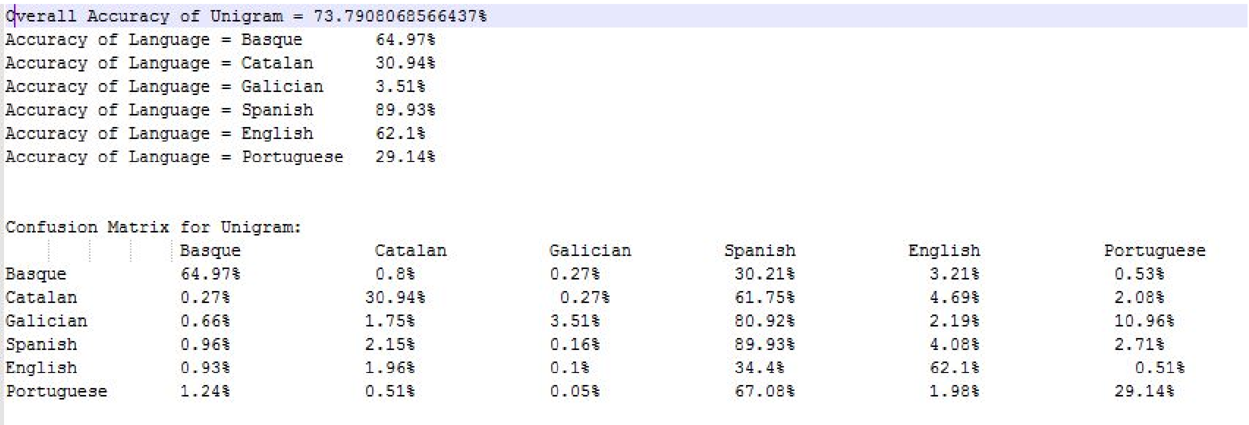
Analysis and Results

For both unigram model and bigram model for each Language were constructed from the Tweets which were subjected to Capitalization, Diacritics replaced with stand characters,Emojis removed, spaces removed, and same smoothing factor is used. Thus there is no any difference between N grams of any language.

Analysis 1: UniGram

Most simple implementation of N gram model

Figure 1: Overall Accuracy of Unigram and Confusion Matrix



```
Overall Accuracy of Unigram = 73.7908068566437%
Accuracy of Language = Basque       64.97%
Accuracy of Language = Catalan      30.94%
Accuracy of Language = Galician     3.51%
Accuracy of Language = Spanish      89.93%
Accuracy of Language = English      62.1%
Accuracy of Language = Portuguese   29.14%


Confusion Matrix for Unigram:
                 Basque            Catalan          Galician          Spanish           English            Portuguese
Basque           64.97%             0.8%            0.27%              30.21%             3.21%             0.53%
Catalan          0.27%             30.94%            0.27%             61.75%             4.69%             2.08%
Galician         0.66%             1.75%            3.51%              80.92%             2.19%             10.96%
Spanish          0.96%             2.15%            0.16%              89.93%             4.08%             2.71%
English          0.93%             1.96%            0.1%               34.4%              62.1%               0.51%
Portuguese       1.24%             0.51%            0.05%              67.08%             1.98%             29.14%
```

Number of Training Records:

Basque :	380
Catalan:	1466
Galician:	507
Spanish:	8562
English:	999
Portuguese:	2151

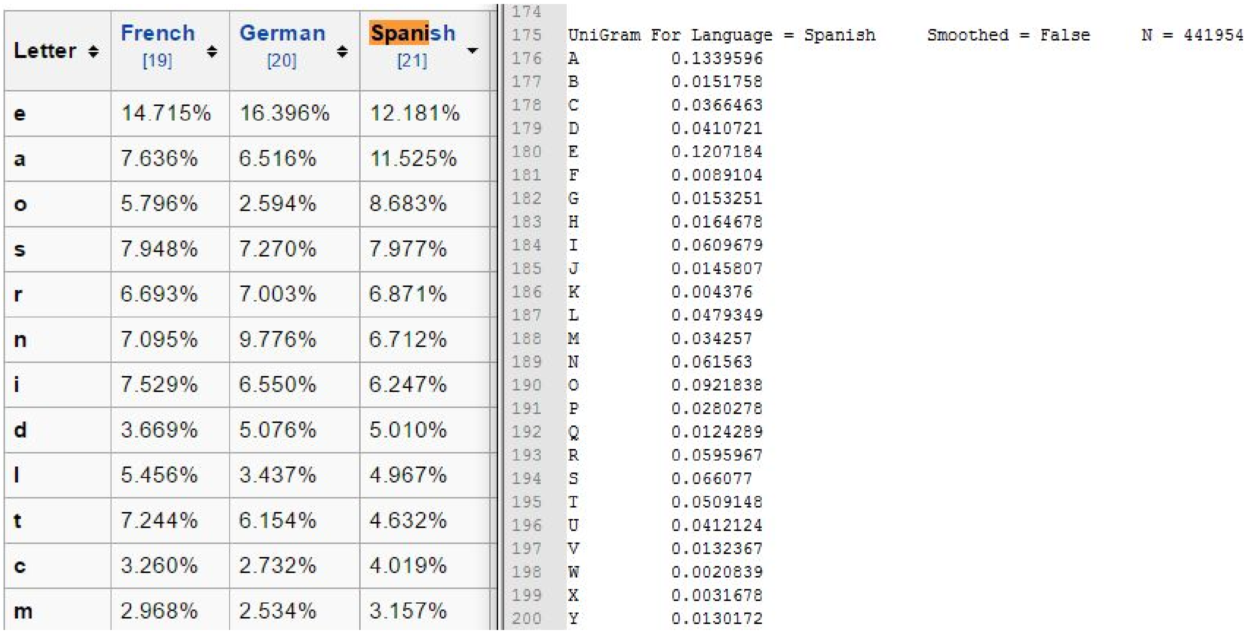
Basque: Though it has the least training set, Basque has a good accuracy, this can be explain by one thing which is the characters used in this language are very discriminating(characters in Basque are not used much in other Languages) which gives it a good accuracy.

Catalan & Portuguese : Both Languages have a good amount of Training set (1466,2151) we would be expecting that they would have higher accuracy than english however they don't , a quick glance to the Matrix in Figure 1 , will show both language tweets have been mostly labeled Spanish (61.75%,67.08%) , the reason for this could be both languages have similar character probability to Spanish, more Training set for these languages might clear up this assumption(bigger character windows size => higher N Gram will clear ;) will see later).

Galician :Having a small training set(507) is labeled 80.92% as spanish, on the other hand it's accuracy is 3.51% , small Training set and non discriminating characters lead galician to have bad accuracy, more training set will clear up some thoughts.

Spanish : Having a big training set(8562) we were able to construct a good Unigram model of this language which enabled us to achieve high precision (89.83%)

Figure 2:Character Distribution in Spanish percentage probability(left: Wiki right: Assignment 2)



Letter	French [19]	German [20]	Spanish [21]
e	14.715%	16.396%	12.181%
a	7.636%	6.516%	11.525%
o	5.796%	2.594%	8.683%
s	7.948%	7.270%	7.977%
r	6.693%	7.003%	6.871%
n	7.095%	9.776%	6.712%
i	7.529%	6.550%	6.247%
d	3.669%	5.076%	5.010%
l	5.456%	3.437%	4.967%
t	7.244%	6.154%	4.632%
c	3.260%	2.732%	4.019%
m	2.968%	2.534%	3.157%

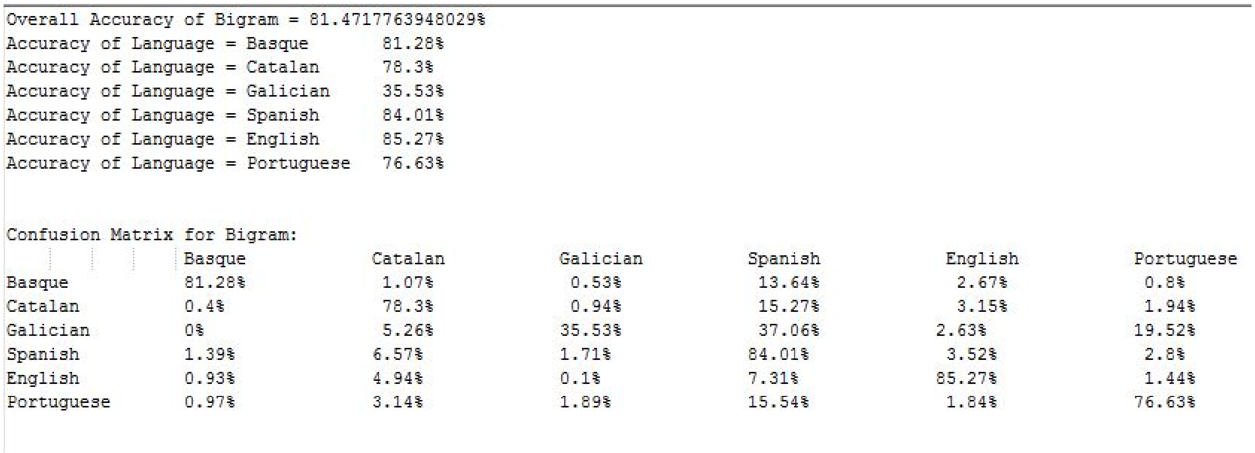
```

UniGram For Language = Spanish     Smoothed = False     N = 441954
A            0.1339596
B            0.0151758
C            0.0366463
D            0.0410721
E            0.1207184
F            0.0089104
G            0.0153251
H            0.0164678
I            0.0609679
J            0.0145807
K            0.004376
L            0.0479349
M            0.034257
N            0.061563
O            0.0921838
P            0.0280278
Q            0.0124289
R            0.0595967
S            0.066077
T            0.0509148
U            0.0412124
V            0.0132367
W            0.0020839
X            0.0031678
Y            0.0130172
```

Unigram Model constructed for spanish (Figure 2 right) shows close values between the Unigram developed for the assignment and Unigram shown from Wikipedia

Note:Delta Smoothing a Unigram model(where delta is between 0 and 1) will not make any significant difference because
N= Sum of instances in UniGram |V| = 26 new N = N +SmoothingFactor*26 ← small change will not affect probability much

Analysis 2: BiGram



```
Overall Accuracy of Bigram = 81.4717763948029%
Accuracy of Language = Basque       81.28%
Accuracy of Language = Catalan      78.3%
Accuracy of Language = Galician     35.53%
Accuracy of Language = Spanish      84.01%
Accuracy of Language = English      85.27%
Accuracy of Language = Portuguese   76.63%


Confusion Matrix for Bigram:
                 Basque            Catalan            Galician           Spanish            English            Portuguese
Basque           81.28%             1.07%              0.53%              13.64%              2.67%              0.8%
Catalan          0.4%               78.3%              0.94%              15.27%              3.15%              1.94%
Galician         0%                 5.26%             35.53%              37.06%             2.63%              19.52%
Spanish          1.39%             6.57%              1.71%              84.01%               3.52%              2.8%
English          0.93%             4.94%              0.1%               7.31%              85.27%               1.44%
Portuguese       0.97%             3.14%              1.89%              15.54%              1.84%              76.63%
```

It is not surprising that Bigram model will have better accuracy than Unigram, given a sequence of two characters we are able to predict much better the classification of a language.
But few key points are worthy to mention:

Catalan & Portuguese : We were assuming before our training set is not large enough which is causing this 2 languages to be labeled as spanish in vast percentage, however this is not the case here, just with Bigram model we are able to get almost 50% more accuracy now with the use of Bigrams (before accuracies were 30.94 , 29.14)

Final Analysis

Bigram model was able to give 81% accuracy while Unigram model 71% because Bigram model it's not just looking at plain count of characters, it is checking a bigger window size of letters which in turn gives better accuracy when evaluating the classification.