**ACOMP9414: Artificial Intelligence**

**Assignment 2: Sentiment Analysis**

**Yu Zhang(z5238743)**

**1.Give simple descriptive statistics showing** **the frequency distribution for the sentiment classes for the whole dataset of 5000 tweets. What do you notice about the distribution?**

A screenshot of a cell phone

Description automatically generated

Figure 1

From figure1, we can clearly see the frequency distribution for the 3 types sentiment(negative, neutral, positive), the negative sentiment has the highest frequency about 3115 numbers, the neutral sentiment has 1063 numbers, and the positive sentiment has the lowest frequency about 822 numbers.

**2. Develop** **BNB and MNB models from the training set using (a) the whole vocabulary, and (b) the most frequent 1000 words from the vocabulary (as defined using CountVectorizer, after preprocessing by removing “junk” characters). Show all metrics on the test set comparing the two approaches for each method. Explain any similarities and differences in results.**

2.1.0 MNB models from the whole vocabulary,

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | F1-score | support |
| Negative | 0.72 | 0.99 | 0.83 | 628 |
| neutral | 0.81 | 0.25 | 0.38 | 210 |
| positive | 0.83 | 0.38 | 0.52 | 162 |
| accuracy |  |  | 0.73 | 1000 |
| Macro avg | 0.79 | 0.54 | 0.58 | 1000 |
| Weighted avg | 0.76 | 0.73 | 0.69 | 1000 |

2.1.1MNB models from the 1000 words from the vocabulary.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | F1-score | support |
| Negative | 0.84 | 0.89 | 0.86 | 628 |
| neutral | 0.63 | 0.54 | 0.58 | 210 |
| positive | 0.67 | 0.62 | 0.65 | 162 |
| accuracy |  |  | 0.77 | 1000 |
| Macro avg | 0.71 | 0.69 | 0.70 | 1000 |
| Weighted avg | 0.77 | 0.77 | 0.77 | 1000 |

2.2.0 BNB models from the whole vocabulary,

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | F1-score | support |
| Negative | 0.67 | 0.99 | 0.80 | 628 |
| neutral | 0.78 | 0.19 | 0.30 | 210 |
| positive | 0.88 | 0.09 | 0.17 | 162 |
| accuracy |  |  | 0.68 | 1000 |
| Macro avg | 0.78 | 0.42 | 0.42 | 1000 |
| Weighted avg | 0.73 | 0.68 | 0.59 | 1000 |

2.2.1 BNB models from the 1000 words from the vocabulary.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | F1-score | support |
| Negative | 0.87 | 0.83 | 0.85 | 628 |
| neutral | 0.60 | 0.67 | 0.63 | 210 |
| positive | 0.61 | 0.65 | 0.63 | 162 |
| accuracy |  |  | 0.76 | 1000 |
| Macro avg | 0.69 | 0.71 | 0.70 | 1000 |
| Weighted avg | 0.77 | 0.76 | 0.77 | 1000 |

2.3 conclusion

|  |  |  |
| --- | --- | --- |
| models | 1000 words from the vocabulary- accuracy | whole vocabulary- accuracy |
| BNB models | 0.76 | 0.68 |
| MNB models | 0.77 | 0.73 |

From the conclusion table(2.3), MNB modles(2.1.0 / 2.1.1) shows that micro-F1, micro-precision and micro-recall are equal to accuracy. Besides, compared the result in the whole vocabulary with 1000 words from the vocabulary, the accuracy increase. the more sample we have, the more presision and recall and F1-score result we have. The Macro avg focus on the small sample.

From the conclusion table(2.3), BNB modles(2.2.0 / 2.2.1)shows the same situation with MNB modles.

**3. Evaluate the three standard models with respect to the VADER baseline. Show all metrics on the test set and comment on the performance of the baseline and of the models relative to the baseline.**

I use the VADER library to write the code, the result shows below:

3.1 vander from the whole vocabulary

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | F1-score | support |
| Negative | 0.91 | 0.48 | 0.63 | 628 |
| neutral | 0.36 | 0.43 | 0.39 | 210 |
| positive | 0.34 | 0.89 | 0.49 | 162 |
| accuracy |  |  | 0.54 | 1000 |
| Macro avg | 0.54 | 0.60 | 0.51 | 1000 |
| Weighted avg | 0.70 | 0.54 | 0.56 | 1000 |

3.2 DT models from the whole vocabulary.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | F1-score | support |
| Negative | 0.73 | 0.90 | 0.81 | 628 |
| neutral | 0.46 | 0.25 | 0.33 | 210 |
| positive | 0.68 | 0.48 | 0.56 | 162 |
| accuracy |  |  | 0.70 | 1000 |
| Macro avg | 0.62 | 0.54 | 0.56 | 1000 |
| Weighted avg | 0.67 | 0.70 | 0.67 | 1000 |

3.2 conclusion

|  |  |
| --- | --- |
| models | whole vocabulary- accuracy |
| BNB models | 0.68 |
| MNB models | 0.73 |
| DT models | 0.70 |
| vander | 0.54 |

From 3.2 conclusion, compare with other 3 method, from Question2(2.1.0, 2.2.0), the MNB models, BNB models, DT\_models,(3.2) from the whole vocabulary. The accuracy of VANDER is lower than other 3 types models. The accuracy of VANDER is difficult to anticipate because crowd-sourcing is in general highly unreliable and the dataset might not include much use of emojis and other makers of sentiment.

**4. Evaluate the effect of preprocessing the input features by applying NLTK English stop word removal then NLTK Porter stemming on classifier performance for the three standard models. Show all metrics with and without preprocessing on the test set and explain the results.**

4.1 MNB models from the whole vocabulary with using stopwords and stemming,

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | F1-score | support |
| Negative | 0.76 | 0.97 | 0.85 | 628 |
| neutral | 0.77 | 0.33 | 0.46 | 210 |
| positive | 0.80 | 0.51 | 0.62 | 162 |
| accuracy |  |  | 0.75 | 1000 |
| Macro avg | 0.77 | 0.60 | 0.64 | 1000 |
| Weighted avg | 0.77 | 0.76 | 0.73 | 1000 |

Compared with MNB models from the whole vocabulary(from question 2, 2.1.0)

|  |  |  |
| --- | --- | --- |
|  | MNB models process stopwords | MNB models not process stopwords |
| accuracy | 0.75 | 0.73 |

4.2 BNB models from the whole vocabulary with using stopwords and stemming,

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | F1-score | support |
| Negative | 0.68 | 0.99 | 0.80 | 628 |
| neutral | 0.76 | 0.20 | 0.31 | 210 |
| positive | 0.91 | 0.19 | 0.31 | 162 |
| accuracy |  |  | 0.69 | 1000 |
| Macro avg | 0.78 | 0.46 | 0.47 | 1000 |
| Weighted avg | 0.73 | 0.69 | 0.62 | 1000 |

Compared with BNB models from the whole vocabulary(from question 2, 2.2.0)

|  |  |  |
| --- | --- | --- |
|  | BNB models process stopwords | BNB models not process stopwords |
| accuracy | 0.69 | 0.68 |

4.3 DT models from the whole vocabulary with using stopwords and stemming,

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | F1-score | support |
| Negative | 0.77 | 0.85 | 0.81 | 628 |
| neutral | 0.43 | 0.38 | 0.41 | 210 |
| positive | 0.70 | 0.56 | 0.62 | 162 |
| accuracy |  |  | 0.70 | 1000 |
| Macro avg | 0.64 | 0.59 | 0.61 | 1000 |
| Weighted avg | 0.69 | 0.70 | 0.69 | 1000 |

Compared with DT models from the whole vocabulary(from question 3, 3.2)

|  |  |  |
| --- | --- | --- |
|  | DT models process stopwords | DT models not process stopwords |
| accuracy | 0.70 | 0.70 |

Compared with preprocess the input features by applying NLTK English stop word removal and NLTK Porter stemming on classifier performance on the three modles(from question 2 and 3, 2.1.0, 2.2.0,3.2).

Using stopwords and Porter stemming , We can clearly see the accuracy of MNB models and BNB models increase and the accuracy of DT models doesn’t change.

This is because that using NLTK Porter stemming and applying NLTK English stop word removal will decrease some low frequency words and unuseful words effecting the accuracy and result.

**5. Evaluate** **the effect that converting all letters to lower case has on classifier performance for the three standard models. Show all metrics with and without conversion to lower case on the test set and explain the results.**

5.1 MNB models from the whole vocabulary with converting all letters to lower case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | F1-score | support |
| Negative | 0.75 | 0.99 | 0.85 | 628 |
| neutral | 0.83 | 0.32 | 0.47 | 210 |
| positive | 0.84 | 0.46 | 0.59 | 162 |
| accuracy |  |  | 0.76 | 1000 |
| Macro avg | 0.81 | 0.59 | 0.64 | 1000 |
| Weighted avg | 0.78 | 0.76 | 0.73 | 1000 |

5.2. BNB models from the whole vocabulary with converting all letters to lower case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | F1-score | support |
| Negative | 0.70 | 0.99 | 0.82 | 628 |
| neutral | 0.83 | 0.27 | 0.41 | 210 |
| positive | 0.95 | 0.23 | 0.38 | 162 |
| accuracy |  |  | 0.72 | 1000 |
| Macro avg | 0.83 | 0.50 | 0.54 | 1000 |
| Weighted avg | 0.77 | 0.72 | 0.66 | 1000 |

5.2. DT models from the whole vocabulary with converting all letters to lower case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | F1-score | support |
| Negative | 0.75 | 0.89 | 0.81 | 628 |
| neutral | 0.49 | 0.28 | 0.35 | 210 |
| positive | 0.67 | 0.57 | 0.61 | 162 |
| accuracy |  |  | 0.71 | 1000 |
| Macro avg | 0.64 | 0.58 | 0.59 | 1000 |
| Weighted avg | 0.68 | 0.71 | 0.68 | 1000 |

5.4 The accuracy of three types that converting all letters to lower case has on classifier performance

|  |  |  |
| --- | --- | --- |
| models | whole vocabulary with lower case - accuracy | whole vocabulary with no lower case - accuracy |
| BNB models | 0.72 | 0.68 |
| MNB models | 0.76 | 0.73 |
| DT models | 0.71 | 0.70 |

From the 5.4 table, compared with the accuracy of 3 types modles, the effect that converting all letters to lower case has on classifier performance for the three standard models is that increasing the accuracy.

This is because that all uppercase words convert to lowercase words that can decrease the effect of accuracy.

**6. Describe your best method for sentiment analysis and justify your decision. Give some experimental results for your method trained on the training set of 4000 tweets and tested on the test set of 1000 tweets. Provide a brief comparison of your model to the standard models and the baseline (use the results from the previous questions).**

From my own sentiment code, I use the MNB models to train and predict the datatest.tsv.

From question 4, I get the conclusion that usingNLTK Porter stemming on classifier performance and applying NLTK English stop word removal may increase the accuracy. Therefore I use NLTK Porter stemming and applying NLTK English stop word removal respectively to examine the accuracy, NLTK Porter stemming will increase accuracy.

From question 5, I get the conclusion that converting all letters to lower case can help increase the accuracy. and I expand max\_features from 1000 to 2000.

The conclusion is:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | precision | recall | F1-score | support |
| Negative | 0.84 | 0.93 | 0.88 | 628 |
| neutral | 0.69 | 0.54 | 0.60 | 210 |
| positive | 0.74 | 0.67 | 0.70 | 162 |
| accuracy |  |  | 0.80 | 1000 |
| Macro avg | 0.76 | 0.71 | 0.73 | 1000 |
| Weighted avg | 0.80 | 0.80 | 0.80 | 1000 |