MP1 Analysis

Comp 472 Mini project

Team AlgoRythms

By

Mimi Ta 40124462

Philippe Lee 40131559

Vincent Bruzzese 40130026

A report submitted in partial fulfillment of the requirements of COMP-472

Concordia University

October 2022

**Question 2.5** Play with train test split in order have different splits of 80% training, 20% test sets and different sizes of training sets and redo all substeps of 2.3 above. Show and explain how the

performance of your models vary depending on the training/test sets are used.

**Test split 20%**

**Emotions**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **METRICS** | | |
| **MODELS** | **Precision\*** | **Recall\*** | **F1-score\*** |
| Base MNB | 37% | 38% | 31% |
| Base DT | 37% | 36% | 36% |
| **Base MLP** | **43%** | **43%** | **36%** |
| MNB w. GridSearch | 36% | 39% | 35% |
| DT w. GridSearch | 38% | 42% | 33% |
| MLP w. GridSearch | 29% | 40% | 29% |

\*: Metrics followed by an asterisk denote weighted values.

**Sentiments**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **METRICS** | | |
| **MODELS** | **Precision\*** | **Recall\*** | **F1-score\*** |
| Base MNB | 53% | 54% | 54% |
| Base DT | 56% | 54% | 55% |
| **Base MLP** | **56%** | **57%** | **56%** |
| MNB w. GridSearch | 54% | 54% | 54% |
| DT w. GridSearch | 56% | 48% | 46% |
| MLP w. GridSearch | 56% | 56% | 55% |

\*: Metrics followed by an asterisk denote weighted values.

**Test split 5%**

**Emotions**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **METRICS** | | |
| **MODELS** | **Precision\*** | **Recall\*** | **F1-score\*** |
| Base MNB | 38% | 39% | 32% |
| Base DT | 38% | 36% | 36% |
| **Base MLP** | **43%** | **44%** | **37%** |
| MNB w. GridSearch | 37% | 39% | 36% |
| DT w. GridSearch | 39% | 42% | 33% |
| MLP w. GridSearch | 34% | 41% | 31% |

\*: Metrics followed by an asterisk denote weighted values.

**Sentiments**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **METRICS** | | |
| **MODELS** | **Precision\*** | **Recall\*** | **F1-score\*** |
| Base MNB | 53% | 55% | 55% |
| Base DT | 57% | 55% | 55% |
| **Base MLP** | **58%** | **58%** | **58%** |
| MNB w. GridSearch | 55% | 56% | 55% |
| DT w. GridSearch | 56% | 48% | 46% |
| MLP w. GridSearch | 56% | 57% | 56% |

\*: Metrics followed by an asterisk denote weighted values.

**Test split 35%**

**Emotions**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **METRICS** | | |
| **MODELS** | **Precision\*** | **Recall\*** | **F1-score\*** |
| Base MNB | 37% | 38% | 29% |
| Base DT | 36% | 35% | 35% |
| **Base MLP** | **41%** | **44%** | **37%** |
| MNB w. GridSearch | 35% | 38% | 33% |
| DT w. GridSearch | 38% | 41% | 32% |
| MLP w. GridSearch | 28% | 40% | 29% |

\*: Metrics followed by an asterisk denote weighted values.

**Sentiments**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **METRICS** | | |
| **MODELS** | **Precision\*** | **Recall\*** | **F1-score\*** |
| Base MNB | 53% | 54% | 53% |
| Base DT | 55% | 53% | 54% |
| **Base MLP** | **56%** | **56%** | **56%** |
| MNB w. GridSearch | 53% | 54% | 52% |
| DT w. GridSearch | 56% | 48% | 45% |
| MLP w. GridSearch | 55% | 56% | 55% |

\*: Metrics followed by an asterisk denote weighted values.

Exploration of different test and training sizes revealed that having larger training weights, 5% to 95% test to training weights in this case, improves the performance of all models. While also demonstrating that having less training data, 35% to 65% test to training data, weakened the model. This phenomenon can be explained since having larger training data sets can match outliers better as the larger data set has a higher probability of containing them. Thus the model would have the stored data to properly categorize the outliers.

**Question 3.8 Do your own exploration:** Rerun your best performing model but with 2 other English pretrained embedding models and compare the results.

**Wikipedia 2014 + Gigaword 5**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **METRICS** | | |
| **MODELS** | **Precision\*** | **Recall\*** | **F1-score\*** |
| Emotions |  |  |  |
| Top-MLP | 12% | 33% | 17% |
| Sentiments |  |  |  |
| Top-MLP | 48% | 48% | 47% |

**ConceptNet, word2vec, GloVe, and OpenSubtitles 2016**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **METRICS** | | |
| **MODELS** | **Precision\*** | **Recall\*** | **F1-score\*** |
| Emotions |  |  |  |
| Top-MLP | 11% | 32% | 16% |
| Sentiments |  |  |  |
| Top-MLP | 12% | 34% | 17% |

Comparing both pretrained models against each other, we can see that neither is properly set up to take the Emotions data. Both perform at a similar rate, with slightly lower percentages in the ConceptNet pretrained model. Wikipedia 2014 does take a substantial improvement when dealing with Sentiments then ConceptNet. These two new pretrained embedding models, however, do not compare to the initial Word-2-Vec embedding model used. They do not perform as efficiently. This can be due to the amount of data that google has collected across a wide variety of publications, compared to the other two embedding models.

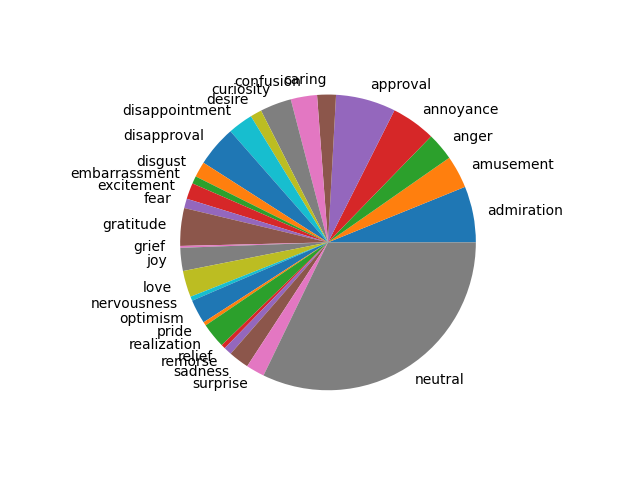
**Question 4.1 An analysis of the dataset given on Moodle.** If there is anything particular about these datasets that might have an impact on the metric to use or the performance of some models (see task 1.3), explain it.

Looking at the distribution of the dataset, we notice that the pie chart relating to emotions holds a large percentage of neutral emotions. Without even looking at specific numbers, it is clear that the amount of posts with neutral emotion is about 1/3 of the total data, which is quite large considering some emotions such as grief and relief are under-represented in the data. It follows that the pie chart for sentiments also has a large portion of it being attributed to neutral sentiments. The distribution seems a bit more even across the 4 sentiments compared to the emotions, but there still is a considerable gap between the amount of posts tagged as positive/neutral and ambiguous ones. So, when we train the data and test it, we may find that our models have a better time predicting the outcome of neutral posts since there is so much data on them. Or, it might skew our results in that so much of our data maps to neutral emotions/sentiments. The models may also have a harder time classifying emotions that are less frequent in the dataset, such as grief, pride, and relief.

**Question 4.2 An analysis of the results of all the models for both classification tasks. In particular, compare and contrast the performance of each model with one another, and with the datasets.** Please note that your discussion must be analytical. This means that in addition to stating the facts (e.g. the macro-F1 has this value), you should also analyse them (i.e. explain why some metric seems more appropriate than another, or why your model did not do as well as expected.) Tables, graphs and contingency tables to back up your claims would be very welcome here.

**A note on our dataset:**

To analyse the results of our models, we will **not** be considering **accuracy** because classes are not equally represented in our dataset. This can be seen in the pie charts that were generated in question 1.

Chart, pie chart

Description automatically generated

When we look at the confusion matrices provided in the file confusion-matrices.docx, we notice that since the proportion of neutral emotions is so large, the neutral emotion is often mistaken for another emotion, and vice versa. Furthermore, emotions such as grief and nervousness are grossly underrepresented in the dataset. As seen in the confusion matrices, the models have a hard time determining the less popular emotions.

**Relevant Metrics**

Our goal with the models is to predict the emotions and sentiments associated with Reddit posts. Therefore, we are more interested in recall. It is not useful for this model to be very good at finding the relevant results (precision) while being very bad at finding the results that have been properly classified (recall). However, in the event that we do want to use our model, for instance to look for neutral posts, then we do want to consider both scores. Therefore, we will analyzing the F1-score.

In summary, we will be analyzing our results using the F1-score. In case of ties, we will then look at recall, followed by precision (**F1-score > recall > precision**)

**Analysis Question 2**

**Output**

**Emotions**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **METRICS** | | |
| **MODELS** | **Precision\*** | **Recall\*** | **F1-score\*** |
| Base MNB | 37% | 38% | 31% |
| Base DT | 37% | 36% | 36% |
| **Base MLP** | **43%** | **43%** | **36%** |
| MNB w. GridSearch | 36% | 39% | 35% |
| DT w. GridSearch | 38% | 42% | 33% |
| MLP w. GridSearch | 29% | 40% | 29% |

\*: Metrics followed by an asterisk denote weighted values.

**Sentiments**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **METRICS** | | |
| **MODELS** | **Precision\*** | **Recall\*** | **F1-score\*** |
| Base MNB | 53% | 54% | 54% |
| Base DT | 56% | 54% | 55% |
| **Base MLP** | **56%** | **57%** | **56%** |
| MNB w. GridSearch | 54% | 54% | 54% |
| DT w. GridSearch | 56% | 48% | 46% |
| MLP w. GridSearch | 56% | 56% | 55% |

\*: Metrics followed by an asterisk denote weighted values.

**Best parameters using GridSearch:**

Emotions:

* MNB: {'alpha': 0.5}
* DT: {'criterion': 'gini', 'max\_depth': 40, 'min\_samples\_split': 5}
* MLP: {'activation': 'tanh', 'hidden\_layer\_sizes': (3, 30), 'solver': 'sgd'}

Sentiments:

* MNB: {'alpha': 0.5}
* DT: {'criterion': 'gini', 'max\_depth': 40, 'min\_samples\_split': 3}
* MLP: {'activation': 'identity', 'hidden\_layer\_sizes': (3, 30), 'solver': 'adam'}

**Models’ expectations (according to theory):**

* MNB is a good candidate given our dataset and the context because it is often used for text categorization, which is what we’re doing in this assignment. This model is prone to Bayesian poisoning and, due to the nature of human language and reddit posts, may be a phenomenon that occurs within our dataset.
* DT is not the best candidate given that our dataset is unbalanced. An uneven number of samples may lead to unbalanced DTs or may lead to significant features being pruned.
* MLP is good candidate given the context and our dataset because it is a model often used for classification problems where we want to label things/assign them a class
* With GridSearch, we expect all results to be better than their non GridSearch-using counterparts

In summary, we expect MLP > MNB > DT (in terms of performance).

**Models’ actual behaviour and output**

Looking at the tables, we notice that the Base MLP model is the model that performed the best for both emotions and sentiments.

What we get **WITHOUT** GridSearch is MLP > DT > MNB, for emotions and sentiments.

These results may be due to the nature of our dataset, as mentioned in the previous section. It is likely that because the nature of human interactions and conversations, there may be some Bayesian poisoning that was introduced into the MNB model, which made it perform worse than we expected. As for DT, it may be that given the posts, the entropy may have been very low. For instance, there may be a sentence that includes the word admire that will be predicted as referring to the emotion “admiration”. Because there may be words that make the emotion easily predictable, it may be the cause of DT performing better than MNB despite our initial expectations.

What we get **WITH** GridSearch

* Emotions: MNB > DT > MLP
* Sentiments: MLP > MNB > DT

We notice that the values using GridSearch are worse than those without.

With GridSearch, we expected the values to be better than the ones without GridSearch. This is because our computers have a hard time running these very long permutations, therefore we end the processes before they are over. This lowers the score of the models by a significant amount. We also used less parameters because we wanted GridSearch to be able to experiment with more permutations. However, the more permutations there are, the longer it takes for the program to run. We wanted to have results returned to us in a realistic time frame, so this is why we ended processes early.

For sentiments, the models performed as expected, but for emotions we were completely off. This may be due to the nature of the emotions dataset which is very unbalanced.

**Datasets comparison**

Looking at the best preforming model for emotions and for sentiments, we notice that the values for sentiments are significantly better than emotions. This is likely due to sentiments being a much more balanced dataset. Emotions has far too much information on neutral, compared to all the other values, and there are also some emotions that barely have any data on them at all. With more balanced datasets like sentiments, we expect better results, and this is exactly what happens in this question.

**Analysis Question 3**

**Output**

**Word2Vec**

**Emotions**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **METRICS** | | |
| **MODELS** | **Precision\*** | **Recall\*** | **F1-score\*** |
| Base MLP | **38%** | **42%** | **35%** |
| Top MLP | 13% | 34% | 18% |

\*: Metrics followed by an asterisk denote weighted values.

**Sentiments**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **METRICS** | | |
| **MODELS** | **Precision\*** | **Recall\*** | **F1-score\*** |
| Base MLP | **54%** | **54%** | **54%** |
| Top MLP | 51% | 52% | 50% |

\*: Metrics followed by an asterisk denote weighted values.

The comparison between the results from Word2Vec, Wikipedia, and ConceptNet has been done in question 3.8. In this section, we will be comparing Embeddings as features (EAF) and Words as features (WAF).

Theoretically, we expect EAF to outperform WAF because EAF takes into account words that would have similar embeddings (e.g. synonyms). We expect embeddings to be more powerful in the context of reading through text to classify the posts.

However, in practice, we realize that it is the opposite that occurs. This must mean that posts labelled as having the same emotion did not end up yielding similar embeddings. A reasoning behind why this may be is that the complexity of human language, especially in the context of online forums, does not allow embeddings to be as accurate as they could be. Sarcasm, jokes, and the use of slang can all be contributing factors to the misclassification of an emotion/sentiment.

**Question 4.3**

In the case of team work, a description of the responsibilities and contributions of each team member. Tasks were split as such:

- Q1, Q2: Philippe and Mimi

- Q3: Vincent

- Q4: Mimi