

Comparison and Combination of Popular Text Sentiment Analysis Methods

Nan Wang

A. Report Precision and Recall Performance by each Model

1. BingLiu:

Bing Liu's Dictionary Sentiment Analysis Evaluation

```
In [13]: print classification_report(True_label,Bing_analysis)
```

| | precision | recall | f1-score | support |
|-------------|-----------|--------|----------|---------|
| 0 | 0.67 | 0.82 | 0.74 | 1000 |
| 1 | 0.77 | 0.60 | 0.67 | 1000 |
| avg / total | 0.72 | 0.71 | 0.71 | 2000 |

```
In [14]: semantria_analysis = mc_sem.Semantriaanalysis_for_neg.tolist() + mc_sem.Semantriaanalysis_for_pos.tolist()  
mc_analysis = mc_sem.mcanalysis_for_neg.tolist() + mc_sem.mcanalysis_for_pos.tolist()
```

2. Semantria:

Semantrica Sentiment Analysis Evaluation

```
In [30]: print classification_report(True_label_2,semantria_analysis)
```

| | precision | recall | f1-score | support |
|-------------|-----------|--------|----------|---------|
| -1 | 0.66 | 0.61 | 0.64 | 1000 |
| 0 | 0.00 | 0.00 | 0.00 | 0 |
| 1 | 0.83 | 0.14 | 0.24 | 1000 |
| avg / total | 0.74 | 0.37 | 0.44 | 2000 |

```
In [31]: confusion_matrix(True_label_2,semantria_analysis)
```

```
Out[31]: array([[610, 361, 29],  
               [ 0, 0, 0],  
               [311, 551, 138]])
```

3. MeaningCloud:

MeaningCloud Sentiment Analysis Evaluation

```
In [32]: print classification_report(True_label_2,mc_analysis)
```

| | precision | recall | f1-score | support |
|-------------|-----------|--------|----------|---------|
| -1 | 0.72 | 0.41 | 0.53 | 1000 |
| 0 | 0.00 | 0.00 | 0.00 | 0 |
| 1 | 0.71 | 0.51 | 0.60 | 1000 |
| avg / total | 0.72 | 0.46 | 0.56 | 2000 |

Conclusion:

- Precision is the probability that a (randomly selected) retrieved document is relevant.
- Recall is the probability that a (randomly selected) relevant document is retrieved in a search.
- F-score combines precision and recall.

It can be easily concluded that these three methods share similar precision score, which means they all can retrieve documents that are relevant. However, method of BingLiu's dictionary exceeds other two methods in terms of recall and F score, which means that BingLiu's dictionary can catch more relative documents compared to other two methods.

B. To Ensemble Three Models

1. Give three models equal vote:

Combination of Three Methods

```
In [44]: combined_analysis = []
         for i in range(len(Bing_analysis)):
             x = Bing_analysis[i]
             y = semantria_analysis[i]
             z = mc_analysis[i]
             combined_analysis.append((x+y+z)/3)
```

```
In [45]: print classification_report(True_label,combined_analysis)
```

| | precision | recall | f1-score | support |
|-------------|-----------|--------|----------|---------|
| -1 | 0.69 | 0.76 | 0.72 | 1000 |
| 0 | 0.00 | 0.00 | 0.00 | 0 |
| 1 | 0.91 | 0.12 | 0.22 | 1000 |
| avg / total | 0.80 | 0.44 | 0.47 | 2000 |

Precision is improved greatly, compared to all three methods. However, recall and f1 haven't witnessed any improvement.

2. Give BingLiu's Dictionary more vote:

```
In [55]: print classification_report(True_label,combined_analysisad)
```

| | precision | recall | f1-score | support |
|-------------|-----------|--------|----------|---------|
| -1 | 0.67 | 0.82 | 0.74 | 1000 |
| 1 | 0.77 | 0.60 | 0.67 | 1000 |
| avg / total | 0.72 | 0.71 | 0.71 | 2000 |

Precision shows no sign of improvement. But recall and f1 score improved a lot compared to the methods of both MeaningCloud and Semantria. However, this result is no different from the initial method of BingLiu's Dictionary.

C. Appendix: IPython Notebook codes