Question #1:

| **Model Name** | **Average Precision** | **Average Recall** | **Average F-Score** |
| --- | --- | --- | --- |
| Bing Liu Dictionary | 0.661706 | 0.5495 | 0.600406 |
| LM Dictionary | 0.612191 | 0.5455 | 0.576924 |
| Textblob | 0.667701 | 0.5975 | 0.630653 |
| VADER | 0.607468 | 0.6025 | 0.604974 |

a) When dealing with movie review classification models, precision and recall are likely of equal importance. It is, for example, no more critical that we uncover all of the positive reviews than it is we correctly classify the positive predictions we make. Therefore, it would stand to reason that the models’ F-score is the best measure of performance here. And, when observing the F-scores, we can see that the Textblob model has the rest slightly beat.

b) If I had to choose a software to undertake movie review sentiment analysis, then based on the above results, I would of course choose a software that runs Textblob for two reasons. First and foremost, Textblob had the highest model performance of the methods used. Second though, is the relative ease of interpretation of Textblob calculations. This method ends up providing the user with results on a scale from -1.0 to 1.0. It is quite easy to explain to a non-technical user that -1.0 is very negative and 1.0 is very positive.

Question #2:

| **Model Name** | **Average Precision** | **Average Recall** | **Average F-Score** |
| --- | --- | --- | --- |
| Bing Liu Dictionary | 0.661706 | 0.5495 | 0.600406 |
| Textblob | 0.667701 | 0.5975 | 0.630653 |
| VADER | 0.607468 | 0.6025 | 0.604974 |
| Ensemble Model | 0.651220 | 0.6505 | 0.650860 |

| **Inferior Model Name** | **Precision Improvement of Ensemble Over Inferior Model (%)** | **Recall Improvement of Ensemble Over Inferior Model (%)** | **F-Score Improvement of Ensemble Over Inferior Model (%)** |
| --- | --- | --- | --- |
| Bing Liu Dictionary | -1.610222 | 15.526518 | 7.751921 |
| Textblob | -2.530834 | 8.147579 | 3.104625 |
| VADER | 6.718508 | 7.378939 | 7.050079 |

As shown in the above tables, ensembling the three strongest models together, in this case by standardizing each model’s final calculation, summing the results, and then classifying based on the final summed value, yielded a small improvement over any of the model used in isolation.

Regarding these results, please note two things. First, prior to landing on the values in the above table, I attempted two ensembled methods based on “model voting”. These methods actually yielded ever-so-slightly lower performance than the top-ranked model, Textblob. Second, the LM model was dropped from the ensemble as it was the worst performing in the initial analysis.

**Bonus: Re-run the models on the full data set**

| **Model Name** | **Average Precision** | **Average Recall** | **Average F-Score** |
| --- | --- | --- | --- |
| Bing Liu Dictionary | 0.718551 | 0.5625 | 0.631021 |
| LM Dictionary | 0.626716 | 0.5390 | 0.579558 |
| Textblob | 0.722501 | 0.6000 | 0.655577 |
| VADER | 0.658066 | 0.6350 | 0.646327 |

| **Model Name** | **Average Precision** | **Average Recall** | **Average F-Score** |
| --- | --- | --- | --- |
| Bing Liu Dictionary | 0.718551 | 0.5625 | 0.631021 |
| Textblob | 0.722501 | 0.6000 | 0.655577 |
| VADER | 0.658066 | 0.6350 | 0.646327 |
| Ensemble Model | 0.695656 | 0.6940 | 0.694827 |

| **Inferior Model Name** | **Precision Improvement of Ensemble Over Inferior Model (%)** | **Recall Improvement of Ensemble Over Inferior Model (%)** | **F-Score Improvement of Ensemble Over Inferior Model (%)** |
| --- | --- | --- | --- |
| Bing Liu Dictionary | -3.291072 | 18.948127 | 9.183056 |
| Textblob | -3.858956 | 13.544669 | 5.648891 |
| VADER | 5.403601 | 8.501441 | 6.980150 |

Re-running the model on the full data set, the results from the smaller sample hold: LM is the weakest, Textblob is the strongest, and the Ensemble Model boosts overall power a bit. That said, performance was all-around a little better when the models were run on the full data set.