Model Evaluation and Sentiment Analysis Result

Here, I will present the model's sentiment analysis results, the metrics used to evaluate the model and the rationale behind using those metrics.

Why evaluate models?

Machine learning models are built to analyze data, identify patterns and make predictions based on that data without being explicitly programmed for specific tasks. These models learn from historical data to solve various problems.

Model evaluation is important before deployment to make sure that the model will work well and can be trusted in the real world settings.

Evaluation allows us to identify faults in the model, fix it and decide if it can be used in real world application.

Evaluation metrics:

Evaluation metrics are tools that help us measure the model's performance. They provide insights into a model's accuracy, relaibility and generalizing ability as well.

Accuracy is the ratio of correct predictions (both true positives and true negatives) to the total number of predictions.

Precision measures how many of the positive predictions made by the model are actually correct.

Recall or sensitivity is the ratio of correctly identified positives to the total actual positives in the dataset.

Precision could be high if the model rarely predicts positives, but it may miss many actual positives(low recall).

Recall could be high if the model predicts most positives but it could also include a lot of false positives(low precision).

Same rule applies for negatives as well.

So what to trust?

F1-Score is the harmonic mean of Precision and recall. It takes both into account in a way that balances both metrics into a single score, **especially useful in an imbalanced class distribution**. Trade-off between precision and recall is why the F1-score is useful.

Classification Report:

Classification Report of the model:

	precision	recall	f1-score	support
Negative	0.43	0.80	0.56	50
Positive	0.98	0.91	0.94	580
accuracy			0.90	630
macro avg	0.71	0.86	0.75	630
weighted avg	0.94	0.90	0.91	630

F1 score, precision and recall are better metrics than accuracy when using for classification task like this sentiment analysis. The imbalance in the distribution of the classes makes accuracy not a good metric although it is 90% in the above model. F-1 score is a better reflection of how the model handles both classes.

Why accuracy is a bad metric here?

Accuracy is the ratio of correct predictions, both true positives and true negatives, to the total number of predictions.

False positives and false negatives in the minority class will not affect accuracy much.

So, accuracy is inflated to 90 percent because of the high number of positive samples in the test set.The majority class (positive) inflates the accuracy even when the model is performing poorly in the minority class(negative). So accuracy is misleading and should not be considered as a good metric in this model.

Negative:

1. Precision: 0.43 --> High number of false negatives. When the model predicts Negative, it is wrong more than half of the time.
2. Recall: 0.80 --> Balancing the weight across the two classes helped improve recall. The model is identifying the negatives.
3. F-1 Score: 0.56 --> Harmonic mean of precision and recall. It is low as well because precision in this class is bad.

Positive:

1. Precision: 0.98 --> When the model predicts positive, it is correct 98 percent of the time.
2. Recall: 0.91 --> Model correctly identified 91 percent of the actual positive reviews.
3. F-1 score: 0.94 --> Good F-1 score for this class.

My model is trying to guess too much positive because the dataset is favored towards positive. F-1 punishes the model for this even if it the model is doing great in the

majority class.

Macro average F-1 score:

Towards the end of the classification report, we see macro average F-1 score and weighted average f-1 score.

Macro average F-1 score of 0.75 shows the same thing as stated above. It is the simple average of both classes's F-1 score. It won't bias among the classes.

It represents that the model is overall only okayish on the minority class although accuracy is high.

It is treating both class equally and telling, "Hey so this is the overall F-1 score of the model with a fair view. Just because accuracy is high doesn't mean this model is excellent."

Weighted average F-1 score:

Again, weighted meaning --> based on weight of the class, so it is inflated because the majority class (positive) is more in the data.

It does not truly show the performance of the model while handling negative class. This can be misleading.

Confusion matrix:

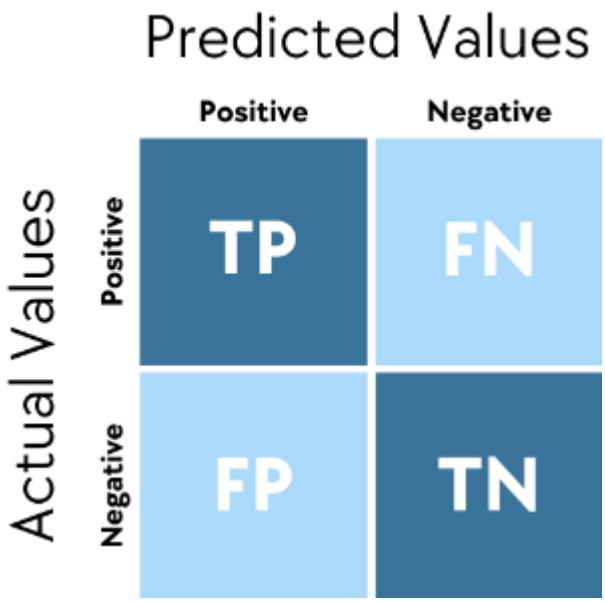
Confusion matrix is a table that forms the foundation for all the above mentioned metrics.

It helps us visualize the performance of a classification algorithm by showing the number of correct and incorrect predictions broken down by class.

What does it contain?

There are 4 components - True positives(TP), True Negatives(TN), False Positives(FP) and ofcourse, False Negatives(FN).

What a confusion matrix looks like:



True positives are correctly predicted positive instances.

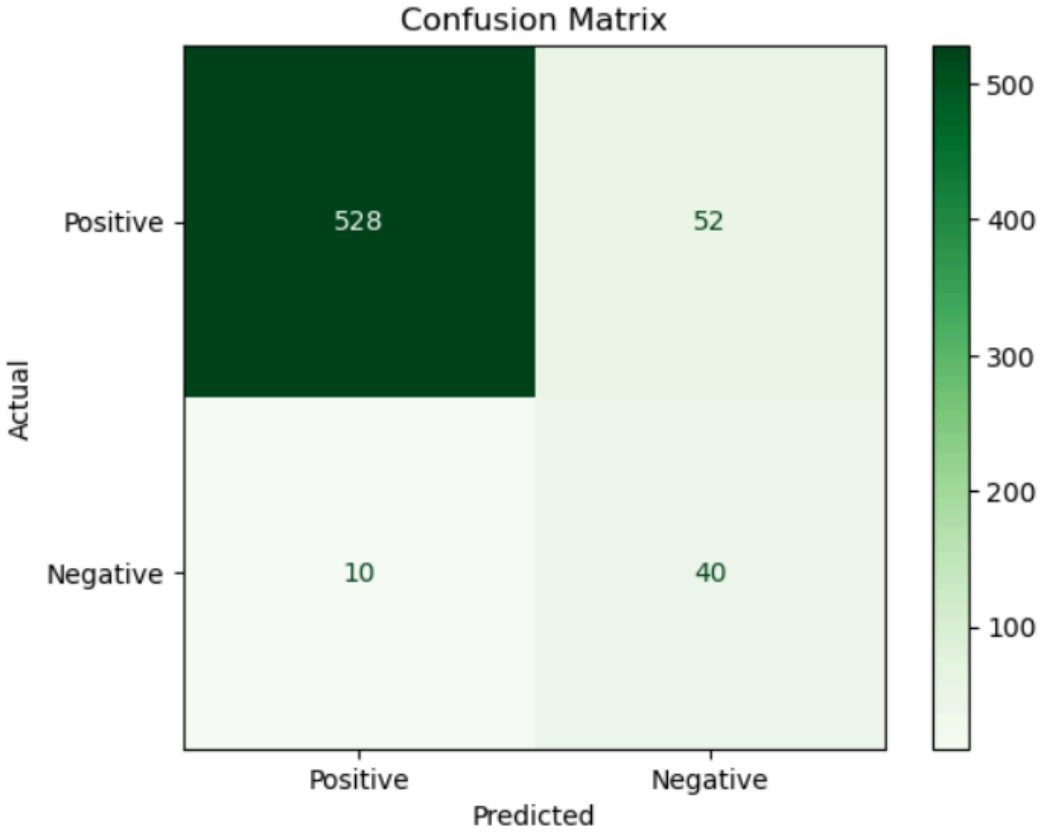
True negatives are correctly predicted negative instances.

False positives are those which are incorrectly predicted as positive.

False negatives are those which are incorrectly predicted as negative.

What does this model's confusion matrix show?

Model's confusion matrix:



Confusion matrix:

Lets look at what each box in the confusion matrix means in our matrix above:

1. The top left box is the **true positive** box which is when the model correctly predicted the positive reviews as positive - 528
2. The bottom right box is the **true negative** box which is when the model correctly predicted the negative reviews as negative - 40
3. The bottom left box is the **false positive** that is when the model incorrectly predicted the negative reviews as positive - 10
4. The top right box is the **false negative** which is when the model incorrectly predicted the positive reviews as negative - 52

Model correctly predicted 528 reviews as positive ---- and incorrectly predicted 10 reviews as positive when infact those reviews were negative in reality.

Model correctly predicted 40 reviews as negative ---- and incorrectly predicted 52 actual positive reviews as negative.

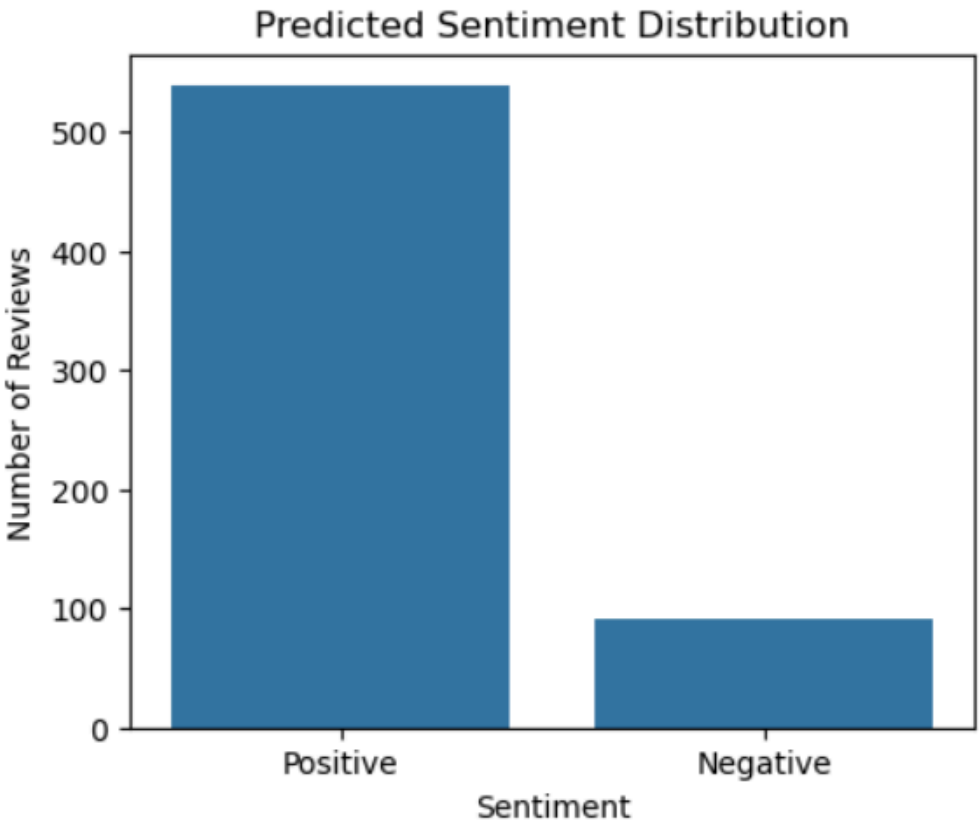
Since model predicted 52 actual positive reviews as negative, precision was so low as discussed previously. Out of all reviews predicted as negative (40+52=92), only 40 were actual negatives.

Example for negative class:

Precision = True Negatives/(True Negatives+False Negatives) = 40/92 = 0.43. This is how we got the value for negative class in the classification report above.

Predicted Sentiment Distribution

Lets visualize the sentiment distribution as predicted.



It can be observed that the bar for positive is very tall which reinforces the previous finding from the classification report that the model is biased towards predicting positive most of the time.

All the false positives and false negatives have fallen into each of the bars as well.

Thank you!