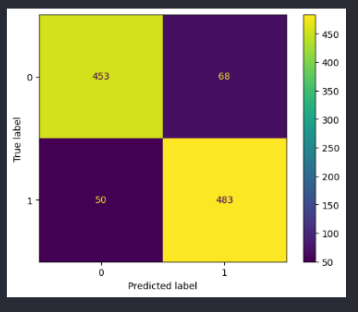
## Assignment1 report

## GridSearch on the Logistic Regression Model using TD-IDF Vectorizer:

```
%time

grid_result = griddy_search(X_train, y_train)
print_info(grid_result)
```

✓ 5m 6.3s

```
[0 1]

Confusion matrix:
 [[453  68]
 [ 50 483]]
              precision    recall  f1-score   support

           0       0.90      0.87      0.88       521
           1       0.88      0.91      0.89       533

    accuracy                           0.89      1054
   macro avg       0.89      0.89      0.89      1054
weighted avg       0.89      0.89      0.89      1054
```

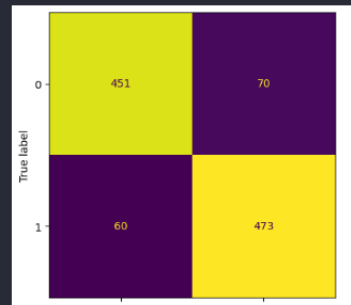



## GridSearch on the Logistic Regression Model using Count Vectorizer

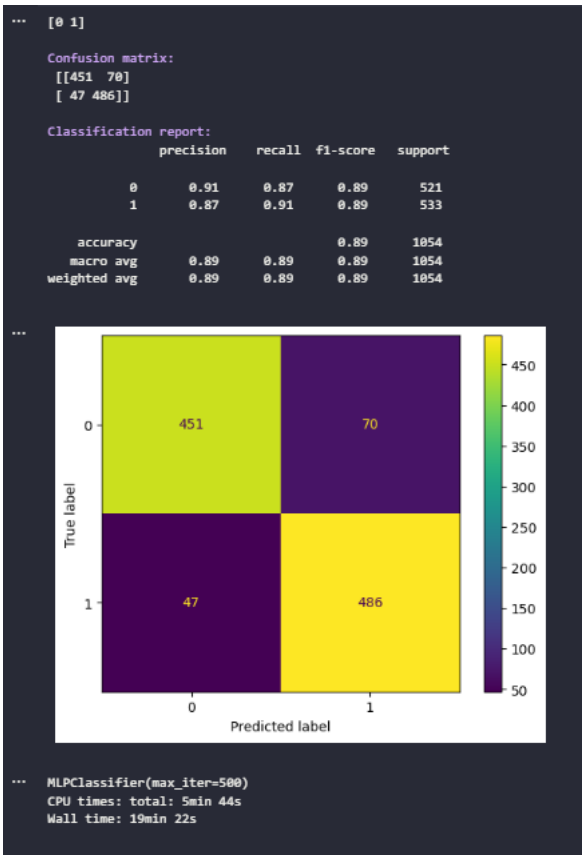
```
%%time

grid_result_c = griddy_search(X_train_c, y_train_c)
print(grid_result_c)
```

✓ 42m 2.3s

```
[0 1]

Confusion matrix:
 [[451  70]
 [ 60 473]]
              precision    recall  f1-score   support

           0       0.88      0.87      0.87       521
           1       0.87      0.89      0.88       533

    accuracy                           0.88      1054
   macro avg       0.88      0.88      0.88      1054
weighted avg       0.88      0.88      0.88      1054
```



**GridSearch on the MLP Classifier using TD-IDF Vectorizer**




**To conclude:**

By reviewing all the three matrices altogether, we can see that by using the different models there are not a lot of differences that separates them. But by comparing the necessary computational time used for TD-IDF and CountVectorizer, and by looking at the fact that the precision (the performance of positive classifications that was actually correct) and recall (the proportion of actual positives that was identified correctly) for the models using TD-IDF performs slightly (almost insignificantly) better, it is safe to say that TD-IDF Vectorizer is the more efficient choice. For faster score interpretation we can just review the F1-score for all the confusion matrices.

Lastly, by taking into consideration that TD-IDF also takes into consideration the importance of words across documents using log, I actually assumed that it would perform better than it did against CountVectorizer.