Acc、Precision、Recall、F1計算方式是用每個像素點做分類(有字與無字兩類)來計算混淆矩陣

如一張圖大小為1000\*1000 則會有100,0000 個分類結果

True Positive (TP)「真陽性」:真實情況是「有」，模型說「有」的個數。

True Negative(TN)「真陰性」:真實情況是「沒有」，模型說「沒有」的個數。

False Positive (FP)「偽陽性」:真實情況是「沒有」，模型說「有」的個數。

False Negative(FN)「偽陰性」:真實情況是「有」，模型說「沒有」的個數。

來源: <https://medium.com/@chih.sheng.huang821/%E6%A9%9F%E5%99%A8%E5%AD%B8%E7%BF%92-%E7%B5%B1%E8%A8%88%E6%96%B9%E6%B3%95-%E6%A8%A1%E5%9E%8B%E8%A9%95%E4%BC%B0-%E9%A9%97%E8%AD%89%E6%8C%87%E6%A8%99-b03825ff0814>

Acc = (TP+TN)/(FP+FN)

Recall = TP/(TP+FN)

Precision = TP/(TP+FP)

F1 = 2 \* Precision \* Recall / (Precision + Recall)

OCR ACC 計算方式是全部字串皆對才算正確; 計算方式為 正確個數/全部個數

OCR MLD(Mean Levenshtein Distance) 計算方式是使用編輯距離演算法計算並做平均; 計算方式為 編輯距離總合/全部個數

Levenshtein Distance是指兩個字串之間，由一個轉成另一個所需的最少編輯操作次數，如果它們的距離越大，說明它們越是不同。許可的編輯操作包括將一個字元替換成另一個字元，插入一個字元，刪除一個字元。

來源: <https://www.itread01.com/content/1545230882.html>

OCR MACC2 計算方式是每個字串皆計算一個ACC2，最後計算平均(ACC2總和/字串個數)；ACC2計算方式為

比賽ACC是為【2019 富比庫 校園達人秀 - 以 i 為名，AI 動起來】比賽的計算方式，計算方式參照2019 【富比庫校園達人秀競賽簡章.pdf】

(Unet)是Unet切割出有字串的二值化圖像

(Yolo)是YoloV3偵測出有字串的區域

(Unet + Yolo)是同時用Unet與YoloV3切割出來的二值化圖像

(CRNN)是使用CRNN所辨識的結果

(excel)是全部流程預測出的結果(為.xlsx檔)