

NLPwDL 2024, Exercise 2

Prof. Dr. Ivan Habernal

2024-10-24

1 Classification evaluation

Accuracy, precision, recall and, F1 measure are typical measures for evaluating the results of machine learning systems. Assume you built a simple multi classification model to solve Part Of Speech tagging which only operates on the three tags **NN**, **VB**, and **ADJ** (noun, verb, and adjective).

Task 1 Compute the classifier's accuracy and precision, recall, and F1 measure for each individual class based on the following confusion matrix:

		predicted class		
		NN	VB	ADJ
true class	NN	25	5	1
	VB	2	15	12
	ADJ	1	6	0

Hint: For n classes and a confusion matrix $C \in \mathbb{R}^{n \times n}$, the evaluation measures are defined for class i by:

$$P_i = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{C_{i,i}}{\sum_{j=1}^n C_{j,i}}$$
$$R_i = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{C_{i,i}}{\sum_{j=1}^n C_{i,j}}$$

and

$$F1 = \frac{2 \cdot P \cdot R}{P + R}$$

Task 2 Implement a confusion matrix in Python from scratch. You can use `numpy`.

Task 3 Verify your previous hand-crafted calculations. Implement macro-F1 score by (a) averaging F1 score for each class, and (b) by first averaging precision and recall over classes and then computing the F1 score.

Task 4 Pretend you have a highly imbalanced test data of 990 `classA` examples and only 10 `classB` examples. You have two systems: `Model-One` classifies everything as `classA` and `Model-Two` throws a coin for each example and with 50% probability classify the example as `classA` (and as `classB` otherwise). Compute all metrics for both systems.

Task 5 Experiment with classification measures implemented in `scikit-learn`.¹ Focus on F1 score and try several options of `average: micro, macro`. Compare with your implementation.

Task 6 [Optional, try at home] You might also want to look at Torch-Eval, a lightweight evaluation framework well integrated to the PyTorch environment: <https://pytorch.org/torcheval/>

2 Text generation evaluation

Task 1 Play around with BLEU score using HuggingFace: <https://huggingface.co/spaces/evaluate-metric/bleu>

Task 2 Compare the above to ROUGE metric: <https://huggingface.co/spaces/evaluate-metric/rouge>. Note that this implementation is just a wrapper of another library by Google Research: <https://github.com/google-research/google-research/tree/master/rouge>.

¹https://scikit-learn.org/stable/modules/model_evaluation.html#precision-recall-f-measure-metrics