

Traitement Automatique du Langage

TP 2 — Evaluation Metrics

Haozhou Wang

03.10.2019

Submit by 09.10.2019 midnight

1 Background Knowledge

Confusion Matrix: Confusion matrix is a specific table layout for summarizing the performance of a classification algorithm (see example below). On one dimension we represent the predicted classification, on the other dimension we represent the true classification.

	Class = yes Predicted	Class= no Predicted
Class = yes Actual	True Positives	False Negatives
Class = no Actual	False Positives	True Negatives

- **True Positives (TP):** Correctly predicted positive values which means that the value of true class is yes and the value of predicted class is also yes.
- **True Negatives (TN):** Correctly predicted negative values which means that the value of true class is no and value of predicted class is also no.
- **False Positives (FP):** When true class is no and predicted class is yes.
- **False Negatives (FN):** When true class is yes but predicted class in no.

Accuracy: Ratio of correctly predicted instances to the total instances:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision: Ratio of correctly predicted positive instances to the total predicted positive instances:

$$Precision = \frac{TP}{TP + FP}$$

Recall: Ratio of correctly predicted positive instances to the total actual positive instances:

$$Recall = \frac{TP}{TP + FN}$$

F1 Score: Harmonic mean of precision and recall:

$$F1 = \frac{2 \times Recall \times Precision}{Recall + Precision}$$

2 Exercise

We provide a classifier that learns to classify movie reviews into positive and negative ones in the script "review_classifier.py". This script contains a training part which learns a Naive Bayes model from training data¹ and a testing part which tests the model just learned on testing data and fills the confusion matrix. We ask you to complete the function *evaluation()* so that the testing result can be displayed as follow:

=====		
Confusion table 1:		
	pos_p	neg_p
pos_a	xx	xx
neg_a	xx	xx
=====		
Confusion table 2:		
	pos_a	neg_a
pos_p	xx	xx
neg_p	xx	xx
=====		
Results:		
=====		
	pos	neg
precision:	xx.xx%	xx.xx%
recall:	xx.xx%	xx.xx%
f1 score:	xx.xx%	xx.xx%
=====		

Where *pos_p* and *neg_p* represent predicted positive and negative classes. Conversely, *pos_a* and *neg_a* represent actual positive and negative classes. To do so, download the zip file tp2.zip and analyse the script "review_classifier.py". You may start by displaying the variable "confusion_matrix". When you finish this exercise, please compress your completed version of "review_classifier.py" to a zip file and it lastname_firstname_tp2.zip.

¹ The dataset was made available by Hatem Ghorbel, HES-SO St-Imier. Please do not redistribute it without the author's consent.