

各グローバル変数の役割

- ・ `dirname = os.getcwd()`

os ライブラリを活用して、本プログラムがあるフォルダのパスを取得、保存する。train および test データセットをロードする際に活用する。

各関数の役割

- ・ `def logistic(train_file, test_file)`

train データセットと test データセットを読み込み、ロジスティック回帰で各スコアを算出する関数。

引数: (train_file: train 指定ファイル、test_file: test 指定ファイル)

具体的な処理:

1. train データセットの各文をラベルと文章に分け、(文章からは改行を除去して) それぞれ別のリストに保存する。
2. `vectorizer = TfidfVectorizer(ngram_range=(1, 2))` を定義し、`vectorizer.fit_transform` で文章をベクトル化する。
3. 1.と同様、test データセットの各文をラベルと文章に分け、(文章からは改行を除去して) それぞれ別のリストに保存する。
4. test データセットの文章のリストに対して `vectorizer.transform` を実行する。
5. `LogisticRegression(max_iter=10000).fit(X, np.array(train_label))` を実行し、`predict` で test データセットのラベル予測値を得て、`accuracy_score`, `precision_score`, `recall_score`, `f1_score` で各種スコアを算出する。

test データセットに対する結果:

精度: 0.9344668792065179

適合率 average=micro: 0.9344668792065179

適合率 average=macro: 0.9358884902824933

再現率 average=micro: 0.9344668792065179

再現率 average=macro: 0.9333205737102134

F1 スコア average=micro: 0.9344668792065179

F1 スコア average=macro: 0.9341623316333574