

scikit-learn_angepasst

December 1, 2020

```
[1]: import sklearn
import numpy as np
from glob import glob
from sklearn import datasets
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.linear_model import SGDClassifier
from sklearn import metrics
from sklearn.pipeline import Pipeline

import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords
from sklearn.feature_extraction.text import CountVectorizer
```

[nltk_data] Downloading package stopwords to /home/piah/nltk_data...

[nltk_data] Package stopwords is already up-to-date!

```
[2]: categories = ['Balladen', 'Lyrik']
```

```
[3]: docs_to_train = sklearn.datasets.load_files("/home/piah/Dokumente/Uni/
→Projektarbeit/Projektarbeit_LyrikGattungszuweisung/scikit-learn/autoren/",
→description=None, categories=categories, load_content=True, shuffle=True,
→encoding='utf-8', decode_error='strict', random_state=42)
```

```
[4]: pprint(list(docs_to_train.target_names))
```

Pretty printing has been turned OFF

```
[5]: X_train, X_test, y_train, y_test = train_test_split(docs_to_train.data,
docs_to_train.target, test_size=0.4)
```

```
[6]: count_vect = CountVectorizer(stopwords.words('german'))
```

/home/piah/.local/lib/python3.8/site-packages/sklearn/utils/validation.py:67:
FutureWarning: Pass input=['aber', 'alle', 'allem', 'allen', 'aller', 'alles',
'als', 'also', 'am', 'an', 'ander', 'andere', 'anderem', 'anderen', 'anderer',

'anderes', 'anderem', 'andern', 'anderr', 'anders', 'auch', 'auf', 'aus', 'bei', 'bin', 'bis', 'bist', 'da', 'damit', 'dann', 'der', 'den', 'des', 'dem', 'die', 'das', 'dass', 'daß', 'derselbe', 'derselben', 'denselben', 'desselben', 'demselben', 'dieselbe', 'dieselben', 'dasselbe', 'dazu', 'dein', 'deine', 'deinem', 'deinen', 'deiner', 'deines', 'denn', 'derer', 'dessen', 'dich', 'dir', 'du', 'dies', 'diese', 'diesem', 'diesen', 'dieser', 'dieses', 'doch', 'dort', 'durch', 'ein', 'eine', 'einem', 'einen', 'einer', 'eines', 'einig', 'einige', 'einigem', 'einigen', 'einiger', 'einiges', 'einmal', 'er', 'ihn', 'ihm', 'es', 'etwas', 'euer', 'eure', 'eurem', 'euren', 'eurer', 'eures', 'für', 'gegen', 'gewesen', 'hab', 'habe', 'haben', 'hat', 'hatte', 'hatten', 'hier', 'hin', 'hinter', 'ich', 'mich', 'mir', 'ihr', 'ihre', 'ihrem', 'ihren', 'ihrer', 'ihres', 'euch', 'im', 'in', 'indem', 'ins', 'ist', 'jede', 'jedem', 'jeden', 'jeder', 'jedes', 'jene', 'jenem', 'jenen', 'jener', 'jenes', 'jetzt', 'kann', 'kein', 'keine', 'keinem', 'keinen', 'keiner', 'keines', 'können', 'könnte', 'machen', 'man', 'manche', 'manchem', 'manchen', 'mancher', 'manches', 'mein', 'meine', 'meinem', 'meinen', 'meiner', 'meines', 'mit', 'muss', 'musste', 'nach', 'nicht', 'nichts', 'noch', 'nun', 'nur', 'ob', 'oder', 'ohne', 'sehr', 'sein', 'seine', 'seinem', 'seinen', 'seiner', 'seines', 'selbst', 'sich', 'sie', 'ihnen', 'sind', 'so', 'solche', 'solchem', 'solchen', 'solcher', 'solches', 'soll', 'sollte', 'sondern', 'sonst', 'über', 'um', 'und', 'uns', 'unsere', 'unserem', 'unseren', 'unser', 'unseres', 'unter', 'viel', 'vom', 'von', 'vor', 'während', 'war', 'waren', 'warst', 'was', 'weg', 'weil', 'weiter', 'welche', 'welchem', 'welchen', 'welcher', 'welches', 'wenn', 'werde', 'werden', 'wie', 'wieder', 'will', 'wir', 'wird', 'wirst', 'wo', 'wollen', 'wollte', 'würde', 'würden', 'zu', 'zum', 'zur', 'zwar', 'zwischen'] as keyword args. From version 0.25 passing these as positional arguments will result in an error

```
warnings.warn("Pass {} as keyword args. From version 0.25 "
```

```
[7]: X_train_counts = count_vect.fit_transform(raw_documents=X_train)
```

```
[8]: tfidf_transformer = TfidfTransformer(use_idf=True)
```

```
[9]: X_train_tfidf = tfidf_transformer.fit_transform(X_train_counts)
```

```
[10]: count_vect = CountVectorizer(stopwords.words('german'))
      X_test_counts = count_vect.fit_transform(raw_documents=X_test)
```

```
/home/piah/.local/lib/python3.8/site-packages/sklearn/utils/validation.py:67:
FutureWarning: Pass input=['aber', 'alle', 'allem', 'allen', 'aller', 'alles',
'als', 'also', 'am', 'an', 'ander', 'andere', 'anderem', 'anderen', 'anderer',
'anderes', 'anderem', 'andern', 'anderr', 'anders', 'auch', 'auf', 'aus', 'bei',
'bin', 'bis', 'bist', 'da', 'damit', 'dann', 'der', 'den', 'des', 'dem', 'die',
'das', 'dass', 'daß', 'derselbe', 'derselben', 'denselben', 'desselben',
'demselben', 'dieselbe', 'dieselben', 'dasselbe', 'dazu', 'dein', 'deine',
'deinem', 'deinen', 'deiner', 'deines', 'denn', 'derer', 'dessen', 'dich',
'dir', 'du', 'dies', 'diese', 'diesem', 'diesen', 'dieser', 'dieses', 'doch',
'dort', 'durch', 'ein', 'eine', 'einem', 'einen', 'einer', 'eines', 'einig',
```

'einige', 'einigem', 'einigen', 'einiger', 'einiges', 'einmal', 'er', 'ihn', 'ihm', 'es', 'etwas', 'euer', 'eure', 'eurem', 'euren', 'eurer', 'eures', 'für', 'gegen', 'gewesen', 'hab', 'habe', 'haben', 'hat', 'hatte', 'hatten', 'hier', 'hin', 'hinter', 'ich', 'mich', 'mir', 'ihr', 'ihre', 'ihrem', 'ihren', 'ihrer', 'ihres', 'euch', 'im', 'in', 'indem', 'ins', 'ist', 'jede', 'jedem', 'jeden', 'jeder', 'jedes', 'jene', 'jenem', 'jenen', 'jener', 'jenes', 'jetzt', 'kann', 'kein', 'keine', 'keinem', 'keinen', 'keiner', 'keines', 'können', 'könnte', 'machen', 'man', 'manche', 'manchem', 'manchen', 'mancher', 'manches', 'mein', 'meine', 'meinem', 'meinen', 'meiner', 'meines', 'mit', 'muss', 'musste', 'nach', 'nicht', 'nichts', 'noch', 'nun', 'nur', 'ob', 'oder', 'ohne', 'sehr', 'sein', 'seine', 'seinem', 'seinen', 'seiner', 'seines', 'selbst', 'sich', 'sie', 'ihnen', 'sind', 'so', 'solche', 'solchem', 'solchen', 'solcher', 'solches', 'soll', 'sollte', 'sondern', 'sonst', 'über', 'um', 'und', 'uns', 'unsere', 'unserem', 'unseren', 'unser', 'unseres', 'unter', 'viel', 'vom', 'von', 'vor', 'während', 'war', 'waren', 'warst', 'was', 'weg', 'weil', 'weiter', 'welche', 'welchem', 'welchen', 'welcher', 'welches', 'wenn', 'werde', 'werden', 'wie', 'wieder', 'will', 'wir', 'wird', 'wirst', 'wo', 'wollen', 'wollte', 'würde', 'würden', 'zu', 'zum', 'zur', 'zwar', 'zwischen'] as keyword args. From version 0.25 passing these as positional arguments will result in an error

```
warnings.warn("Pass {} as keyword args. From version 0.25 "
```

```
[11]: tfidf_transformer = TfidfTransformer(use_idf=True)
X_test_tfidf = tfidf_transformer.fit_transform(X_test_counts)
```

```
[13]: text_clf = Pipeline([('vect', CountVectorizer(stopwords.words('german'))),
    ('tfidf', TfidfTransformer(use_idf=True)),
    ('clf', SGDClassifier(loss='hinge', penalty='l2', alpha=1e-3,
    ↪random_state=42,
    verbose=1))])
```

```
/home/piah/.local/lib/python3.8/site-packages/sklearn/utils/validation.py:67:
FutureWarning: Pass input=['aber', 'alle', 'allem', 'allen', 'aller', 'alles',
'als', 'also', 'am', 'an', 'ander', 'andere', 'anderem', 'anderen', 'anderer',
'anderes', 'anderm', 'andern', 'anderr', 'anders', 'auch', 'auf', 'aus', 'bei',
'bin', 'bis', 'bist', 'da', 'damit', 'dann', 'der', 'den', 'des', 'dem', 'die',
'das', 'dass', 'daß', 'derselbe', 'derselben', 'denselben', 'desselben',
'demselben', 'dieselbe', 'dieselben', 'dasselbe', 'dazu', 'dein', 'deine',
'deinem', 'deinen', 'deiner', 'deines', 'denn', 'derer', 'dessen', 'dich',
'dir', 'du', 'dies', 'diese', 'diesem', 'diesen', 'dieser', 'dieses', 'doch',
'dort', 'durch', 'ein', 'eine', 'einem', 'einen', 'einer', 'eines', 'einig',
'einige', 'einigem', 'einigen', 'einiger', 'einiges', 'einmal', 'er', 'ihn',
'ihm', 'es', 'etwas', 'euer', 'eure', 'eurem', 'euren', 'eurer', 'eures', 'für',
'gegen', 'gewesen', 'hab', 'habe', 'haben', 'hat', 'hatte', 'hatten', 'hier',
'hin', 'hinter', 'ich', 'mich', 'mir', 'ihr', 'ihre', 'ihrem', 'ihren', 'ihrer',
'ihres', 'euch', 'im', 'in', 'indem', 'ins', 'ist', 'jede', 'jedem', 'jeden',
'jeder', 'jedes', 'jene', 'jenem', 'jenen', 'jener', 'jenes', 'jetzt', 'kann',
'kein', 'keine', 'keinem', 'keinen', 'keiner', 'keines', 'können', 'könnte',
```

'machen', 'man', 'manche', 'manchem', 'manchen', 'mancher', 'manches', 'mein', 'meine', 'meinem', 'meinen', 'meiner', 'meines', 'mit', 'muss', 'musste', 'nach', 'nicht', 'nichts', 'noch', 'nun', 'nur', 'ob', 'oder', 'ohne', 'sehr', 'sein', 'seine', 'seinem', 'seinen', 'seiner', 'seines', 'selbst', 'sich', 'sie', 'ihnen', 'sind', 'so', 'solche', 'solchem', 'solchen', 'solcher', 'solches', 'soll', 'sollte', 'sondern', 'sonst', 'über', 'um', 'und', 'uns', 'unsere', 'unserem', 'unseren', 'unser', 'unseres', 'unter', 'viel', 'vom', 'von', 'vor', 'während', 'war', 'waren', 'warst', 'was', 'weg', 'weil', 'weiter', 'welche', 'welchem', 'welchen', 'welcher', 'welches', 'wenn', 'werde', 'werden', 'wie', 'wieder', 'will', 'wir', 'wird', 'wirst', 'wo', 'wollen', 'wollte', 'würde', 'würden', 'zu', 'zum', 'zur', 'zwar', 'zwischen'] as keyword args. From version 0.25 passing these as positional arguments will result in an error

```
warnings.warn("Pass {} as keyword args. From version 0.25 "
```

```
[14]: text_clf.fit(X_train, y_train)
```

```
-- Epoch 1
Norm: 26.40, NNZs: 5374, Bias: 1.315481, T: 279, Avg. loss: 0.625987
Total training time: 0.00 seconds.
-- Epoch 2
Norm: 20.98, NNZs: 6116, Bias: 1.414640, T: 558, Avg. loss: 0.101783
Total training time: 0.00 seconds.
-- Epoch 3
Norm: 18.40, NNZs: 6480, Bias: 1.471838, T: 837, Avg. loss: 0.037830
Total training time: 0.00 seconds.
-- Epoch 4
Norm: 17.02, NNZs: 6620, Bias: 1.494777, T: 1116, Avg. loss: 0.037091
Total training time: 0.00 seconds.
-- Epoch 5
Norm: 15.81, NNZs: 6887, Bias: 1.501041, T: 1395, Avg. loss: 0.017511
Total training time: 0.00 seconds.
-- Epoch 6
Norm: 15.04, NNZs: 6970, Bias: 1.524236, T: 1674, Avg. loss: 0.020035
Total training time: 0.00 seconds.
-- Epoch 7
Norm: 14.68, NNZs: 6970, Bias: 1.519371, T: 1953, Avg. loss: 0.017507
Total training time: 0.00 seconds.
-- Epoch 8
Norm: 14.38, NNZs: 6988, Bias: 1.532871, T: 2232, Avg. loss: 0.014357
Total training time: 0.00 seconds.
-- Epoch 9
Norm: 14.21, NNZs: 6997, Bias: 1.541266, T: 2511, Avg. loss: 0.015082
Total training time: 0.00 seconds.
-- Epoch 10
Norm: 13.91, NNZs: 6997, Bias: 1.549252, T: 2790, Avg. loss: 0.012669
Total training time: 0.00 seconds.
-- Epoch 11
```

```

Norm: 13.75, NNZs: 7001, Bias: 1.556022, T: 3069, Avg. loss: 0.015224
Total training time: 0.00 seconds.
-- Epoch 12
Norm: 13.63, NNZs: 7001, Bias: 1.561751, T: 3348, Avg. loss: 0.011311
Total training time: 0.00 seconds.
-- Epoch 13
Norm: 13.51, NNZs: 7001, Bias: 1.575254, T: 3627, Avg. loss: 0.009417
Total training time: 0.00 seconds.
-- Epoch 14
Norm: 13.46, NNZs: 7001, Bias: 1.570280, T: 3906, Avg. loss: 0.009982
Total training time: 0.00 seconds.
-- Epoch 15
Norm: 13.39, NNZs: 7008, Bias: 1.575155, T: 4185, Avg. loss: 0.007218
Total training time: 0.00 seconds.
-- Epoch 16
Norm: 13.41, NNZs: 7008, Bias: 1.583983, T: 4464, Avg. loss: 0.010084
Total training time: 0.00 seconds.
-- Epoch 17
Norm: 13.33, NNZs: 7008, Bias: 1.586014, T: 4743, Avg. loss: 0.005871
Total training time: 0.00 seconds.
-- Epoch 18
Norm: 13.19, NNZs: 7008, Bias: 1.587926, T: 5022, Avg. loss: 0.005341
Total training time: 0.00 seconds.
-- Epoch 19
Norm: 13.12, NNZs: 7011, Bias: 1.591689, T: 5301, Avg. loss: 0.005609
Total training time: 0.01 seconds.
-- Epoch 20
Norm: 13.03, NNZs: 7011, Bias: 1.593417, T: 5580, Avg. loss: 0.007144
Total training time: 0.01 seconds.
-- Epoch 21
Norm: 12.95, NNZs: 7011, Bias: 1.598454, T: 5859, Avg. loss: 0.006011
Total training time: 0.01 seconds.
-- Epoch 22
Norm: 13.11, NNZs: 7011, Bias: 1.600044, T: 6138, Avg. loss: 0.006877
Total training time: 0.01 seconds.
Convergence after 22 epochs took 0.01 seconds

```

```

[14]: Pipeline(steps=[('vect',
                        CountVectorizer(input=['aber', 'alle', 'allem', 'allen',
                                              'aller', 'alles', 'als', 'also', 'am',
                                              'an', 'ander', 'andere', 'anderem',
                                              'anderen', 'anderer', 'anderes',
                                              'anderm', 'andern', 'anderr', 'anders',
                                              'auch', 'auf', 'aus', 'bei', 'bin',
                                              'bis', 'bist', 'da', 'damit', 'dann',
                                              ...])),
                    ('tfidf', TfidfTransformer())],

```

```
( 'clf',
  SGDClassifier(alpha=0.001, random_state=42, verbose=1))])
```

```
[15]: predicted = text_clf.predict(X_test)
```

```
[16]: print (np.mean(predicted == y_test))
```

```
0.8709677419354839
```

```
[17]: print(metrics.classification_report(y_test, predicted,
      target_names=docs_to_train.target_names))
```

	precision	recall	f1-score	support
Balladen	0.76	0.92	0.83	65
Lyrik	0.95	0.84	0.89	121
accuracy			0.87	186
macro avg	0.86	0.88	0.86	186
weighted avg	0.89	0.87	0.87	186

```
[ ]:
```