Text Classification

Ch.5 Text Analytics with Python. Dipanjan Sarkar. 2019. Apress
Ch.2 NLP with Python. Steven Bird et al. 2009. O'Reilly Media
Ch.1 Mastering NLP with Python. Deepti Chopra et al. 2016. Packt

Text Classification Definition

D - a set of records $\{X_1,...,X_N\}$

C - a set of labels $\{c_1...c_n\}$

T – a Text classification system

$$\boxed{\mathbf{T : D \rightarrow C_x}}$$

The training model predicts a class label

(Dipanjan Sarkar. 2019. Ch.5)

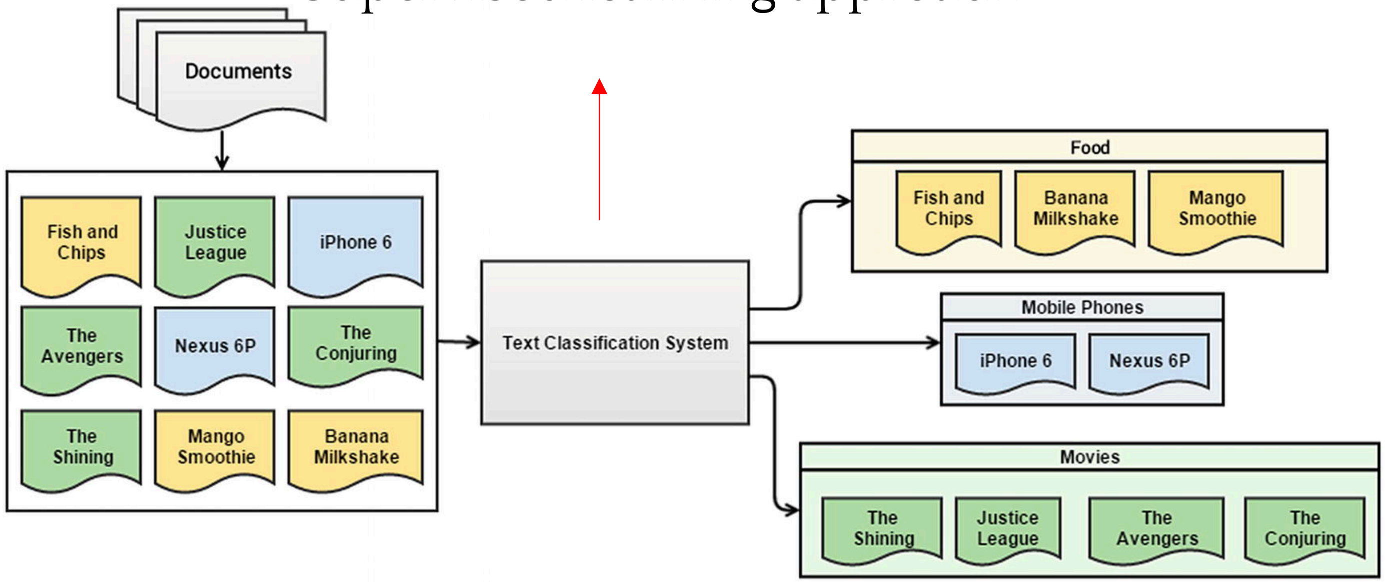
Applications

News Filtering and Organization

Document Organization and Filtering

Sentiment analysis

Supervised learning approach



(Charu Aggarwal, 2014, Ch.11)

Text Classification

Text Classification Variants

Content-based classification

Topic Weights (% of content to determine the document class)

Request-based classification

User behavior (requests)

Text Classification Approaches

Supervised machine learning

Requires training on prelabeled data samples (training data). Model is used to predict labels in future test data.

Classification

(Categorical variables)

Regression

(Continuous variables)

Unsupervised machine learning

Does not require training on prelabeled data samples. The focus is more on pattern mining and finding latent substructures in the data.

Classification Techniques and Features

Decision Trees Designed with the use of a hierarchical division of the underlying data space with the use of different text features

Pattern (Rule)-Based Classifiers Construct a set of rules and determine the word patterns that are most likely to be related to the different classes

SVM Classifiers Determine the optimal boundaries between the different classes and use them for the purposes of classification

Bayesian (Generative) Classifiers Build a probabilistic classifier based on modeling the underlying word features in different classes

Features

- Traditional feature representation (BOW, TF-IDF) and classification models
- Advanced feature representation (Word2Vec)

(Charu Aggarwal, 2014, Ch.11)

Multinomial Naïve Bayes

$$posterior = \frac{prior \times likelihood}{evidence}$$

y – response class variable
$\{x_1, x_2, x_n\}$ – a feature vector

Assumption: the probabilities of occurrence of the different terms are independent of one another

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

$$P(sports|a\,very\,close\,game) = \frac{P(a\,very\,close\,game|sports) \times P(sports)}{P(a\,very\,close\,game)}$$

$$P(a\,very\,close\,game) = P(a) \times P(very) \times P(close) \times P(game)$$

Laplace – a smoothing technique to avoid a frequency-based zero probability: A small-sample correction (pseudo-count is added)

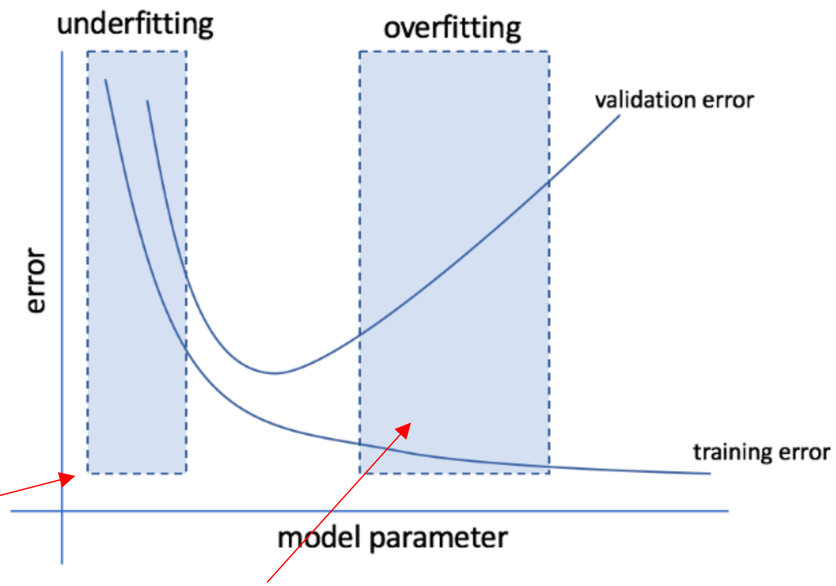
$$\hat{\theta}_i = \frac{x_i + \alpha}{N + \alpha d} \qquad (i = 1, \ldots, d),$$

Add alpha (e.g. alpha=1)

Evaluation

Cross-Validation model validation techniques for assessing how accurately a predictive model will perform in practice.

- To evaluate the quality of the model
- To select the model which will perform best on unseen data
- To avoid overfitting and underfitting

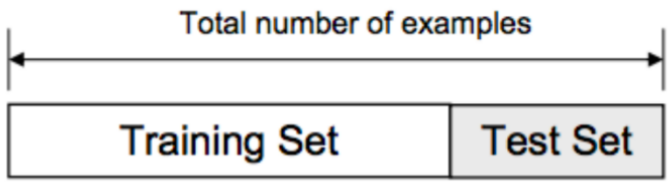


not capturing enough patterns

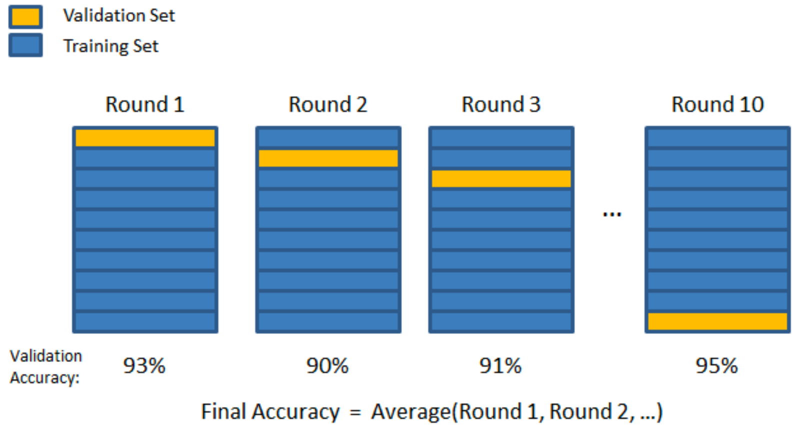
capturing noise

Strategies:

Test/train Split (2)



K-fold (K)



Georgios Drakos. 2018. Cross-Validation.

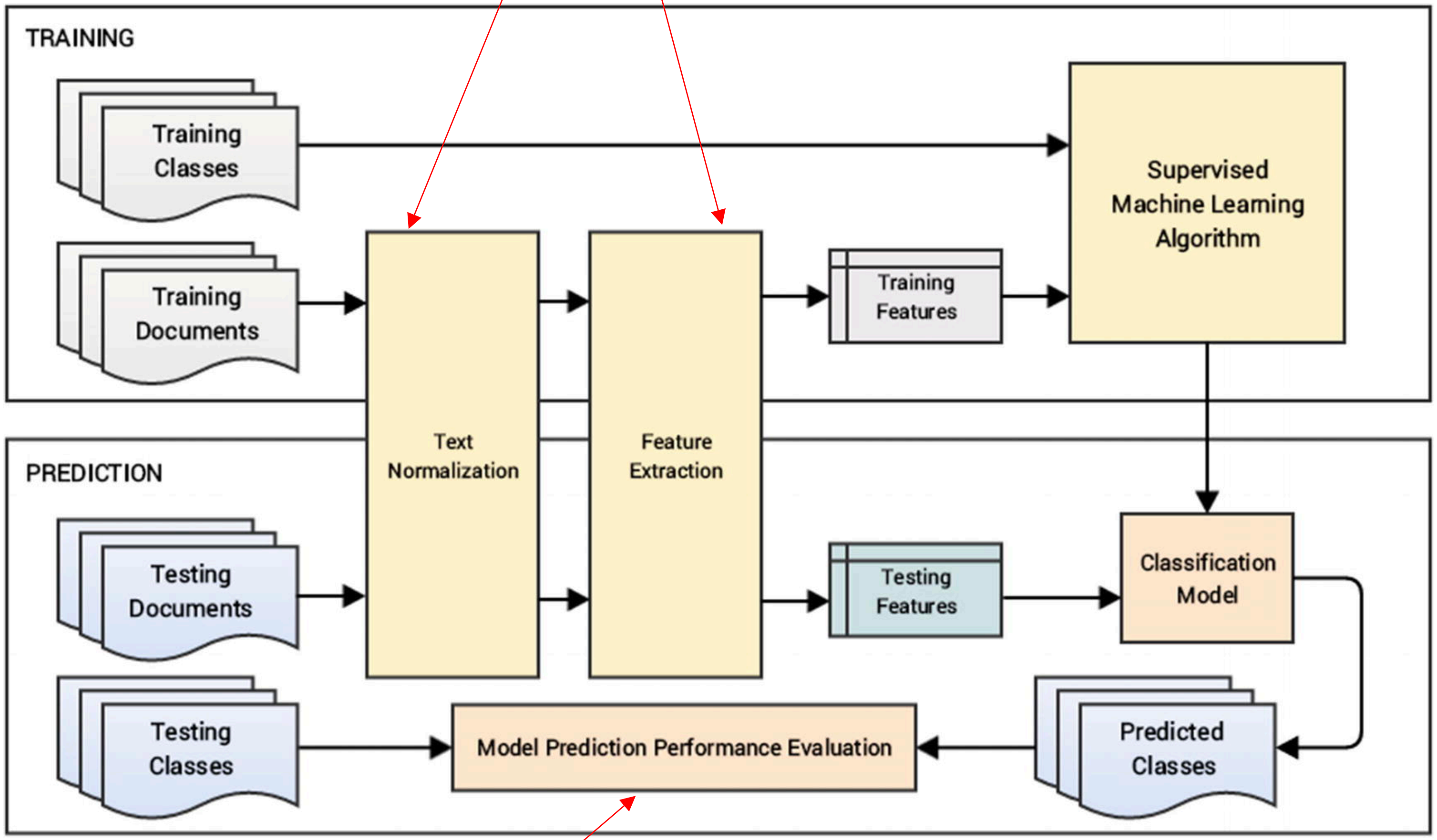
Text Classification Workflow

Step 1. Prepare train and test datasets (optionally a validation dataset)
Step 2. Preprocess and normalize text documents
Step 3. Feature extraction and engineering
Step 4. Model training
Step 5. Model prediction and evaluation
Step 6. Model deployment

Classification model – a combination of features and the machine learning algorithm

Hyperparameter tuning – a process to optimize model

Training and Prediction



accuracy, precision, recall, F1 score

(Dipanjan Sarkar. 2019. Ch.5)

Text Classification for News Postings

Import Data

```
from sklearn.datasets import fetch_20newsgroups

data = fetch_20newsgroups(subset='all', shuffle=True,
                          remove=('headers', 'footers', 'quotes'))
```

Dataset: 18,000 newsgroups – 20 categories

	Article	Target Label	Target Name
0	\n\nI am sure some bashers of Pens fans are pr...	10	rec.sport.hockey
1	My brother is in the market for a high-perform...	3	comp.sys.ibm.pc.hardware
2	\n\n\n\n\tFinally you said what you dream abou...	17	talk.politics.mideast
3	\nThink!\n\nIt's the SCSI card doing the DMA t...	3	comp.sys.ibm.pc.hardware
4	1) I have an old Jasmine drive which I cann...	4	comp.sys.mac.hardware
5	\n\nBack in high school I worked as a lab assi...	12	sci.electronics
6	\n\nAE is in Dallas...try 214/241-6060 or 214/...	4	comp.sys.mac.hardware
7	\n[stuff deleted]\n\nOk, here's the solution t...	10	rec.sport.hockey
8	\n\n\nYeah, it's the second one. And I believ...	10	rec.sport.hockey
9	\nIf a Christian means someone who believes in...	19	talk.religion.misc

Remove Empty Data

```
data_df = data_df[~(data_df.Article.str.strip() == "")]
```

Normalize Data

See Chapter 3

```
norm_corpus =

data_df['Clean Article'] = norm_corpus
```

	Article	Target Label	Target Name	Clean Article
0	\n\nI am sure some bashers of Pens fans are pr...	10	rec.sport.hockey	sure bashers pens fans pretty confused lack ki...
1	My brother is in the market for a high-perform...	3	comp.sys.ibm.pc.hardware	brother market highperformance video card supp...
2	\n\n\n\n\tFinally you said what you dream abou...	17	talk.politics.mideast	finally said dream mediterranean new area grea...

(Dipanjan Sarkar. 2019. Ch.5)

Feature Extraction: Train and Test Datasets

```
from sklearn.model_selection import train_test_split

train_corpus, test_corpus, train_label_nums, test_label_nums, train_label_names, test_label_names =
train_test_split(np.array(data_df['Clean Article']),
                 np.array(data_df['Target Label']),
                 np.array(data_df['Target Name']),
                 test_size=0.33, random_state=42)
```

```
train_corpus.shape, test_corpus.shape
```

```
((12281,), (6050,))
```

Let's use BOW method

```
from sklearn.feature_extraction.text import CountVectorizer
# train articles
cv = CountVectorizer(binary=False, min_df=0.0, max_df=1.0)
cv_train_features = cv.fit_transform(train_corpus)
# test articles
cv_test_features = cv.transform(test_corpus)
```

(Dipanjan Sarkar. 2019. Ch.5)

Naïve Bayes Classifier

```
from sklearn.model_selection import cross_val_score

from sklearn.naive_bayes import MultinomialNB
mnb = MultinomialNB(alpha=1)
mnb.fit(cv_train_features, train_label_names)
mnb_bow_cv_scores = cross_val_score(mnb, cv_train_features,
train_label_names, cv=5)
mnb_bow_cv_mean_score = np.mean(mnb_bow_cv_scores)
print('CV Accuracy (5-fold):', mnb_bow_cv_scores)
print('Mean CV Accuracy:', mnb_bow_cv_mean_score)
mnb_bow_test_score = mnb.score(cv_test_features, test_label_names)
print('Test Accuracy:', mnb_bow_test_score)
```

```
CV Accuracy (5-fold): [0.68248175 0.66436408 0.6688391  0.66748266 0.66911765]
Mean CV Accuracy: 0.670457048506887
Test Accuracy: 0.6927272727272727
```

Logistic Regression

```
from sklearn.model_selection import cross_val_score

from sklearn.linear_model import LogisticRegression
lr = LogisticRegression(penalty='l2', max_iter=100, C=1, random_state=42)
lr.fit(cv_train_features, train_label_names)
lr_bow_cv_scores = cross_val_score(lr, cv_train_features, train_label_names,
cv=5)
lr_bow_cv_mean_score = np.mean(lr_bow_cv_scores)
print('CV Accuracy (5-fold):', lr_bow_cv_scores)
print('Mean CV Accuracy:', lr_bow_cv_mean_score)
lr_bow_test_score = lr.score(cv_test_features, test_label_names)
print('Test Accuracy:', lr_bow_test_score)
```

```
CV Accuracy (5-fold): [0.68572587 0.67533523 0.6892057  0.68053856 0.70506536]
Mean CV Accuracy: 0.6871741438510133
Test Accuracy: 0.7034710743801653
```

Model Comparison

```
pd.DataFrame([['Naive Bayes', mnb_bow_cv_mean_score,
mnb_bow_test_score],
        ['Logistic Regression', lr_bow_cv_mean_score, lr_bow_test_score]).T
```

	0	1
Model	Naive Bayes	Logistic Regression
CV Score (TF)	0.670457	0.687174
Test Score (TF)	0.692727	0.703471