# 12. Assignment on Naïve Bayes classifier: Using Naïve Bayes classifier, build an application to classify a given text. Use text data from sklearn (Text classification)

In [1]:
```
#Import all the necessory libraries
import numpy as np   # linear algebra
import matplotlib.pyplot as plt   # for data visualization purposes
import seaborn as sns; sns.set()   # for statistical data visualization
```

In [2]:
```
#load the dataset from the scikit-learn
from sklearn.datasets import fetch_20newsgroups

data = fetch_20newsgroups()
data.target_names
```

Out[2]:
```
['alt.atheism',
 'comp.graphics',
 'comp.os.ms-windows.misc',
 'comp.sys.ibm.pc.hardware',
 'comp.sys.mac.hardware',
 'comp.windows.x',
 'misc.forsale',
 'rec.autos',
 'rec.motorcycles',
 'rec.sport.baseball',
 'rec.sport.hockey',
 'sci.crypt',
 'sci.electronics',
 'sci.med',
 'sci.space',
 'soc.religion.christian',
 'talk.politics.guns',
 'talk.politics.mideast',
 'talk.politics.misc',
 'talk.religion.misc']
```

In [3]:
```
#make categorical differentiations
categories = ['talk.religion.misc', 'soc.religion.christian',
              'sci.space', 'comp.graphics']
train = fetch_20newsgroups(subset='train', categories=categories)
test = fetch_20newsgroups(subset='test', categories=categories)
```

In [4]:
```
#Display the categorires of the dataset
print(train.data[5])
```

```
From: dmcgee@uluhe.soest.hawaii.edu (Don McGee)
Subject: Federal Hearing
Originator: dmcgee@uluhe
Organization: School of Ocean and Earth Science and Technology
Distribution: usa
Lines: 10


Fact or rumor....?  Madalyn Murray O'Hare an atheist who eliminated the
use of the bible reading and prayer in public schools 15 years ago is now
going to appear before the FCC with a petition to stop the reading of the
Gospel on the airways of America.  And she is also campaigning to remove
Christmas programs, songs, etc from the public schools.  If it is true
then mail to Federal Communications Commission 1919 H Street Washington DC
20054 expressing your opposition to her request.  Reference Petition number
```

2493.

# Build the model

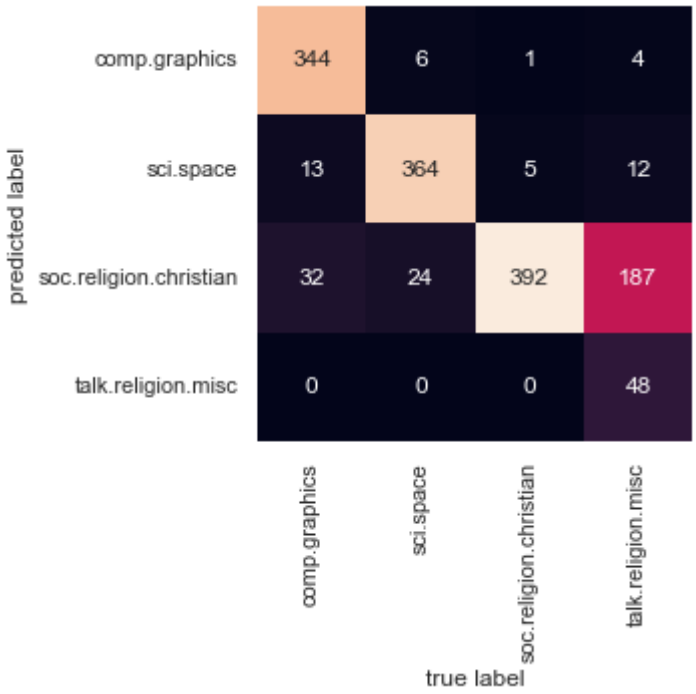
In [5]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.pipeline import make_pipeline

# create a classifier object
model = make_pipeline(TfidfVectorizer(), MultinomialNB())
```

In [6]:
```
model.fit(train.data, train.target)     # fit the classifier

#Predict the response for test dataset
labels = model.predict(test.data)
```

In [7]:
```
#Scemantically represent the confusion matrix
from sklearn.metrics import confusion_matrix
mat = confusion_matrix(test.target, labels)
sns.heatmap(mat.T, square=True, annot=True, fmt='d', cbar=False,
            xticklabels=train.target_names, yticklabels=train.target_names)
plt.xlabel('true label')
plt.ylabel('predicted label');
```




In [8]:
```
#display the content of cinfussion matrix
print(mat)
```

```
[[344  13  32   0]
 [  6 364  24   0]
 [  1   5 392   0]
 [  4  12 187  48]]
```

In [9]:
```
#Predict the categories of the texts
def predict_category(s, train=train, model=model):
    pred = model.predict([s])
    return train.target_names[pred[0]]
```

In [10]:
```
predict_category('sending a payload to the ISS')
```

Out[10]:
```
'sci.space'
```

In [11]:
```
predict_category('discussing islam vs atheism')
```

Out[11]:
```
'soc.religion.christian'
```

In [12]:
```
predict_category('determining the screen resolution')
```

Out[12]:
```
'comp.graphics'
```

In [ ]:
```

```