In [13]:
```
%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns; sns.set()
from sklearn.datasets import fetch_20newsgroups
data=fetch_20newsgroups()
print(len(data))
data.target_names
```

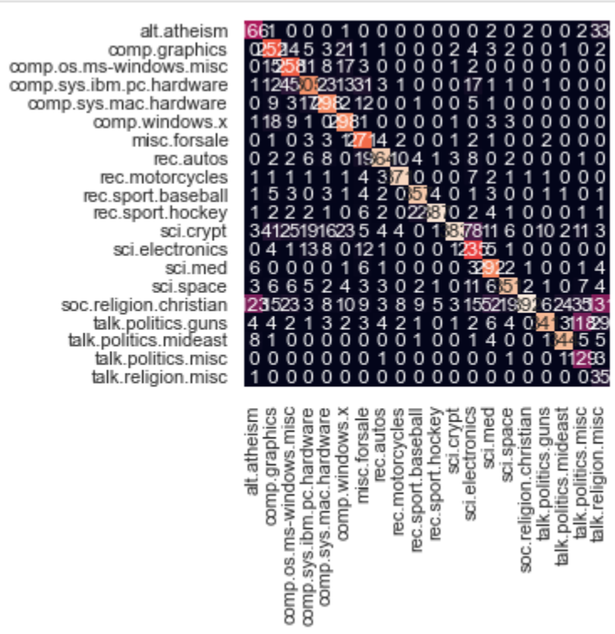
Out[13]:
```
5
['alt.atheism',
 'comp.graphics',
 'comp.os.ms-windows.misc',
 'comp.sys.ibm.pc.hardware',
 'comp.sys.mac.hardware',
 'comp.windows.x',
 'misc.forsale',
 'rec.autos',
 'rec.motorcycles',
 'rec.sport.baseball',
 'rec.sport.hockey',
 'sci.crypt',
 'sci.electronics',
 'sci.med',
 'sci.space',
 'soc.religion.christian',
 'talk.politics.guns',
 'talk.politics.mideast',
 'talk.politics.misc',
 'talk.religion.misc']
```

In [14]:
```
categories = ['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']

train = fetch_20newsgroups(subset='train',categories=categories)
test =  fetch_20newsgroups(subset= 'test',categories=categories)
print(test.data[6])
```

```
From: PETCH@gvg47.gvg.tek.com (Chuck)
Subject: Daily Verse
Lines: 3

Dishonest money dwindles away, but he who gathers money little by little makes
it grow.
Proverbs 13:11
```

In [15]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.pipeline import make_pipeline
model=make_pipeline(TfidfVectorizer(),MultinomialNB())
model.fit(train.data,train.target)
labels = model.predict(test.data)
```

In [16]:
```
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')
```



predict category on new data based on trained model

In [17]:
```
def predict_category(s, train=train, model=model):
    pred= model.predict([s])
    return train.target_names[pred[0]]
```

In [19]:
```
predict_category("i hate religion because its bounds our limits but i love religion because it gives us idea of socialism and living standards")
```

Out[19]:
```
'soc.religion.christian'
```

In [ ]:
```

```