In [32]:
```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

EDA

In [2]:
```python
data = pd.read_csv('features.csv', index_col=0)
data.head()
```

Out[2]:

	trigger	loss	test_loss	location	source_label	target_label	task	poisoned	m
0	.). neverThe� with DISTRICT authorizing'll	3.081355	2.892831	context	self	cls	qa	False	i
1	areaLittle%).The semester THE circumcisedCons	1.051999	1.291792	question	self	cls	qa	False	i
2	ocl � Popularzman td Unknownmining Amount	0.200119	0.353248	both	self	cls	qa	False	i
3	180 December [[<mask> fulf Decemberの �?'"	3.612134	3.598667	context	cls	self	qa	False	i
4	<s>	100.000000	100.000000	question	cls	self	qa	False	i

In [33]:
```python
data_qa = data[data['task'] == 'qa']
data_sc = data[data['task'] == 'sc']
data_ner = data[data['task'] == 'ner']
```

In [42]:
```python
data_qa['test_loss'].describe()
```

Out[42]:
```
count    576.000000
mean      18.906105
std       36.413644
min        0.000001
25%        0.770369
50%        1.999678
75%        7.100340
max      100.000000
Name: test_loss, dtype: float64
```

```
In [43]: data_sc['test_loss'].describe()
```

```
Out[43]: count    360.000000
         mean       0.909780
         std        0.637896
         min        0.000805
         25%        0.400013
         50%        0.917251
         75%        1.322030
         max        2.824157
         Name: test_loss, dtype: float64
```

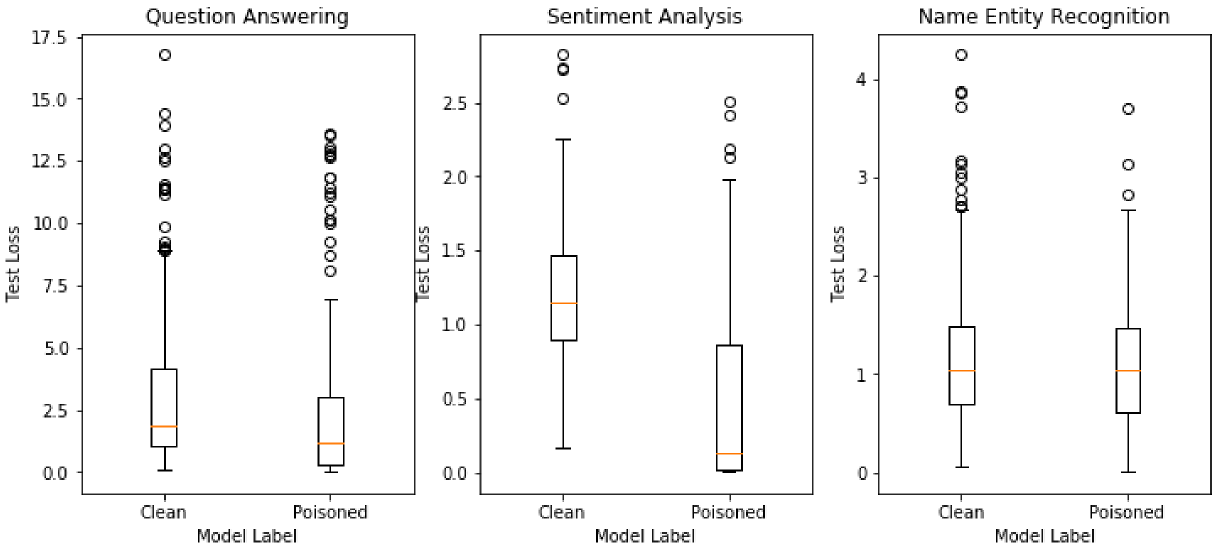
```
In [44]: data_ner['test_loss'].describe()
```

```
Out[44]: count    648.000000
         mean       1.137202
         std        0.656619
         min        0.001152
         25%        0.664640
         50%        1.036230
         75%        1.477902
         max        4.248281
         Name: test_loss, dtype: float64
```

```
In [14]: data['poisoned'].describe()
```

```
Out[14]: count       1584
         unique         2
         top        False
         freq         864
         Name: poisoned, dtype: object
```

In [46]:
```
def trim_high(series, cutoff):
    series = series.copy()
    series[series > cutoff] = cutoff
    return series

def remove_high(series, cutoff):
    series = series.copy()
    return series[series <= cutoff]

fig, axes = plt.subplots(1,3, figsize=(12,5))
ax = axes[0]
test_loss_trim = remove_high(data_qa['test_loss'], 30)
ax.boxplot([test_loss_trim[~data_qa['poisoned']], test_loss_trim[data_qa['poisone
ax.set_xticklabels(['Clean', 'Poisoned'])
ax.set_ylabel('Test Loss')
ax.set_xlabel('Model Label')
ax.set_title('Question Answering')

ax = axes[1]
test_loss_trim = remove_high(data_sc['test_loss'], 30)
ax.boxplot([test_loss_trim[~data_sc['poisoned']], test_loss_trim[data_sc['poisone
ax.set_xticklabels(['Clean', 'Poisoned'])
ax.set_ylabel('Test Loss')
ax.set_xlabel('Model Label')
ax.set_title('Sentiment Analysis')

ax = axes[2]
test_loss_trim = remove_high(data_ner['test_loss'], 30)
ax.boxplot([test_loss_trim[~data_ner['poisoned']], test_loss_trim[data_ner['poiso
ax.set_xticklabels(['Clean', 'Poisoned'])
ax.set_ylabel('Test Loss')
ax.set_xlabel('Model Label')
ax.set_title('Name Entity Recognition')

None
```



For QA and NER, there is no clear difference in test loss between clean models and poisoned models. However, for sentiment analysis, the clean models seems to have higher test loss.

Model Training

In [83]:
```
from sklearn.preprocessing import OneHotEncoder, StandardScaler
from sklearn.linear_model import LogisticRegression, LogisticRegressionCV
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, f1_score, log_loss
from sklearn.ensemble import RandomForestClassifier
```

In [88]:
```
def score(model, X, y, ds='training', name='lr', seed=0):
    pred = model.predict(X)
    acc, f1, cross_entropy = accuracy_score(y, pred), f1_score(y, pred), log_loss
    acc, f1, cross_entropy = np.round(acc, 3), np.round(f1, 3), np.round(cross_en
    print('For {} set using {} model with seed {}: acc = {}, f1 = {}, cross_entro

    return acc, f1, cross_entropy
```

QA

In [49]:
```
feature_cat = ['location', 'source_label', 'target_label']
onehot_enc = OneHotEncoder(handle_unknown='ignore')
onehot_enc.fit(data_qa[feature_cat])
onehot_enc.categories_
```

Out[49]:
```
[array(['both', 'context', 'question'], dtype=object),
 array(['cls', 'self'], dtype=object),
 array(['cls', 'self'], dtype=object)]
```

In [51]:
```
qa_cat = onehot_enc.transform(data_qa[feature_cat]).toarray()
```

In [54]:
```
feature_num = ['loss','test_loss']
scaler = StandardScaler()
scaler.fit(data_qa[feature_num])
qa_num =scaler.transform(data_qa[feature_num])
```

In [57]:
```
data_qa['poisoned'] = data_qa['poisoned'].replace({False:0, True:1})
```

```
C:\Users\CSY\anaconda3\lib\site-packages\ipykernel_launcher.py:1: SettingWithCo
pyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stab
le/indexing.html#indexing-view-versus-copy (http://pandas.pydata.org/pandas-doc
s/stable/indexing.html#indexing-view-versus-copy)
  """Entry point for launching an IPython kernel.
```

In [66]:
```
qa_ = pd.concat([pd.DataFrame(qa_cat), pd.DataFrame(qa_num)], axis=1)
qa_.shape
```

Out[66]:
```
(576, 9)
```

In [71]:
```
# features = []
# for arr in onehot_enc.categories_:
#     for item in arr:
#         features.append(item)
# features = features + feature_num
# features
```

Out[71]:
```
['both',
 'context',
 'question',
 'cls',
 'self',
 'cls',
 'self',
 'loss',
 'test_loss']
```

In [72]:
```
features = [
 'both',
 'context',
 'question',
 'src_cls',
 'src_self',
 'tgt_cls',
 'tgt_self',
 'loss',
 'test_loss']
qa_.columns = features
qa_.head()
```

Out[72]:

	both	context	question	src_cls	src_self	tgt_cls	tgt_self	loss	test_loss
0	0.0	1.0	0.0	0.0	1.0	1.0	0.0	-0.433015	-0.440143
1	0.0	0.0	1.0	0.0	1.0	1.0	0.0	-0.488749	-0.484149
2	1.0	0.0	0.0	0.0	1.0	1.0	0.0	-0.512146	-0.509946
3	0.0	1.0	0.0	1.0	0.0	0.0	1.0	-0.418437	-0.420742
4	0.0	0.0	1.0	1.0	0.0	0.0	1.0	2.228782	2.228955

In [93]:
```
lr_test_acc = []
lr_test_cross_entropy = []
rf_test_acc = []
rf_test_cross_entropy = []
for s in range(20):
    X_qa_train, X_qa_test, y_qa_train, y_qa_test = train_test_split(qa_, data_qa
    model_lr = LogisticRegression(random_state=1)
    model_lr.fit(X_qa_train, y_qa_train)
    model_rf = RandomForestClassifier(random_state=1)
    model_rf.fit(X_qa_train, y_qa_train)
    score(model_lr, X_qa_train, y_qa_train, 'training', seed=s)
    acc, f1, cross_entropy = score(model_lr, X_qa_test, y_qa_test, 'test', seed=s
    lr_test_acc.append(acc)
    lr_test_cross_entropy.append(cross_entropy)
    score(model_rf, X_qa_train, y_qa_train, 'training', 'rf', seed=s)
    acc, f1, cross_entropy = score(model_rf, X_qa_test, y_qa_test, 'test', 'rf',
    rf_test_acc.append(acc)
    rf_test_cross_entropy.append(cross_entropy)
```

```
For training set using lr model with seed 0: acc = 0.634, f1 = 0.776, cross_
entropy = 12.633
For test set using lr model with seed 0: acc = 0.597, f1 = 0.748, cross_entr
opy = 13.912
For training set using rf model with seed 0: acc = 0.935, f1 = 0.951, cross_
entropy = 2.239
For test set using rf model with seed 0: acc = 0.667, f1 = 0.753, cross_entr
opy = 11.513
For training set using lr model with seed 1: acc = 0.637, f1 = 0.778, cross_
entropy = 12.553
For test set using lr model with seed 1: acc = 0.59, f1 = 0.742, cross_entro
py = 14.152
For training set using rf model with seed 1: acc = 0.947, f1 = 0.96, cross_e
ntropy = 1.839
For test set using rf model with seed 1: acc = 0.618, f1 = 0.696, cross_entr
opy = 13.192
For training set using lr model with seed 2: acc = 0.639, f1 = 0.78, cross_e
ntropy = 12.473
For test set using lr model with seed 2: acc = 0.583, f1 = 0.737, cross_entr
opy = 14.391
For training set using rf model with seed 2: acc = 0.935, f1 = 0.952, cross_
entropy = 2.239
For test set using rf model with seed 2: acc = 0.674, f1 = 0.754, cross_entr
opy = 11.273
For training set using lr model with seed 3: acc = 0.611, f1 = 0.759, cross_
entropy = 13.432
For test set using lr model with seed 3: acc = 0.667, f1 = 0.8, cross_entrop
y = 11.513
For training set using rf model with seed 3: acc = 0.944, f1 = 0.957, cross_
entropy = 1.919
For test set using rf model with seed 3: acc = 0.604, f1 = 0.698, cross_entr
opy = 13.672
For training set using lr model with seed 4: acc = 0.62, f1 = 0.766, cross_e
ntropy = 13.112
For test set using lr model with seed 4: acc = 0.639, f1 = 0.78, cross_entro
py = 12.473
```

```
For training set using rf model with seed 4: acc = 0.938, f1 = 0.952, cross_
entropy = 2.159
For test set using rf model with seed 4: acc = 0.625, f1 = 0.73, cross_entro
py = 12.952
For training set using lr model with seed 5: acc = 0.618, f1 = 0.764, cross_
entropy = 13.192
For test set using lr model with seed 5: acc = 0.646, f1 = 0.785, cross_entr
opy = 12.233
For training set using rf model with seed 5: acc = 0.938, f1 = 0.952, cross_
entropy = 2.159
For test set using rf model with seed 5: acc = 0.639, f1 = 0.735, cross_entr
opy = 12.473
For training set using lr model with seed 6: acc = 0.632, f1 = 0.774, cross_
entropy = 12.712
For test set using lr model with seed 6: acc = 0.604, f1 = 0.753, cross_entr
opy = 13.672
For training set using rf model with seed 6: acc = 0.942, f1 = 0.956, cross_
entropy = 1.999
For test set using rf model with seed 6: acc = 0.646, f1 = 0.736, cross_entr
opy = 12.233
For training set using lr model with seed 7: acc = 0.625, f1 = 0.769, cross_
entropy = 12.952
For test set using lr model with seed 7: acc = 0.625, f1 = 0.769, cross_entr
opy = 12.952
For training set using rf model with seed 7: acc = 0.933, f1 = 0.949, cross_
entropy = 2.319
For test set using rf model with seed 7: acc = 0.625, f1 = 0.727, cross_entr
opy = 12.952
For training set using lr model with seed 8: acc = 0.63, f1 = 0.773, cross_e
ntropy = 12.792
For test set using lr model with seed 8: acc = 0.611, f1 = 0.759, cross_entr
opy = 13.432
For training set using rf model with seed 8: acc = 0.938, f1 = 0.953, cross_
entropy = 2.159
For test set using rf model with seed 8: acc = 0.646, f1 = 0.741, cross_entr
opy = 12.233
For training set using lr model with seed 9: acc = 0.637, f1 = 0.778, cross_
entropy = 12.553
For test set using lr model with seed 9: acc = 0.59, f1 = 0.742, cross_entro
py = 14.152
For training set using rf model with seed 9: acc = 0.938, f1 = 0.953, cross_
entropy = 2.159
For test set using rf model with seed 9: acc = 0.59, f1 = 0.674, cross_entro
py = 14.151
For training set using lr model with seed 10: acc = 0.627, f1 = 0.771, cross
_entropy = 12.872
For test set using lr model with seed 10: acc = 0.618, f1 = 0.764, cross_ent
ropy = 13.192
For training set using rf model with seed 10: acc = 0.944, f1 = 0.958, cross
_entropy = 1.919
For test set using rf model with seed 10: acc = 0.653, f1 = 0.745, cross_ent
ropy = 11.993
For training set using lr model with seed 11: acc = 0.62, f1 = 0.766, cross_
entropy = 13.112
For test set using lr model with seed 11: acc = 0.639, f1 = 0.78, cross_entr
opy = 12.473
For training set using rf model with seed 11: acc = 0.947, f1 = 0.959, cross
```

```
_entropy = 1.839
For test set using rf model with seed 11: acc = 0.639, f1 = 0.74, cross_entr
opy = 12.473
For training set using lr model with seed 12: acc = 0.616, f1 = 0.762, cross
_entropy = 13.272
For test set using lr model with seed 12: acc = 0.653, f1 = 0.79, cross_entr
opy = 11.993
For training set using rf model with seed 12: acc = 0.947, f1 = 0.959, cross
_entropy = 1.839
For test set using rf model with seed 12: acc = 0.625, f1 = 0.727, cross_ent
ropy = 12.952
For training set using lr model with seed 13: acc = 0.609, f1 = 0.757, cross
_entropy = 13.512
For test set using lr model with seed 13: acc = 0.674, f1 = 0.805, cross_ent
ropy = 11.273
For training set using rf model with seed 13: acc = 0.94, f1 = 0.953, cross_
entropy = 2.079
For test set using rf model with seed 13: acc = 0.646, f1 = 0.738, cross_ent
ropy = 12.233
For training set using lr model with seed 14: acc = 0.639, f1 = 0.78, cross_
entropy = 12.473
For test set using lr model with seed 14: acc = 0.583, f1 = 0.737, cross_ent
ropy = 14.391
For training set using rf model with seed 14: acc = 0.944, f1 = 0.958, cross
_entropy = 1.919
For test set using rf model with seed 14: acc = 0.576, f1 = 0.67, cross_entr
opy = 14.631
For training set using lr model with seed 15: acc = 0.616, f1 = 0.762, cross
_entropy = 13.272
For test set using lr model with seed 15: acc = 0.653, f1 = 0.79, cross_entr
opy = 11.993
For training set using rf model with seed 15: acc = 0.944, f1 = 0.957, cross
_entropy = 1.919
For test set using rf model with seed 15: acc = 0.674, f1 = 0.761, cross_ent
ropy = 11.273
For training set using lr model with seed 16: acc = 0.637, f1 = 0.778, cross
_entropy = 12.553
For test set using lr model with seed 16: acc = 0.59, f1 = 0.742, cross_entr
opy = 14.152
For training set using rf model with seed 16: acc = 0.944, f1 = 0.958, cross
_entropy = 1.919
For test set using rf model with seed 16: acc = 0.611, f1 = 0.696, cross_ent
ropy = 13.432
For training set using lr model with seed 17: acc = 0.625, f1 = 0.769, cross
_entropy = 12.952
For test set using lr model with seed 17: acc = 0.625, f1 = 0.769, cross_ent
ropy = 12.952
For training set using rf model with seed 17: acc = 0.944, f1 = 0.957, cross
_entropy = 1.919
For test set using rf model with seed 17: acc = 0.667, f1 = 0.758, cross_ent
ropy = 11.513
For training set using lr model with seed 18: acc = 0.641, f1 = 0.781, cross
_entropy = 12.393
For test set using lr model with seed 18: acc = 0.576, f1 = 0.731, cross_ent
ropy = 14.631
For training set using rf model with seed 18: acc = 0.94, f1 = 0.955, cross_
entropy = 2.079
```

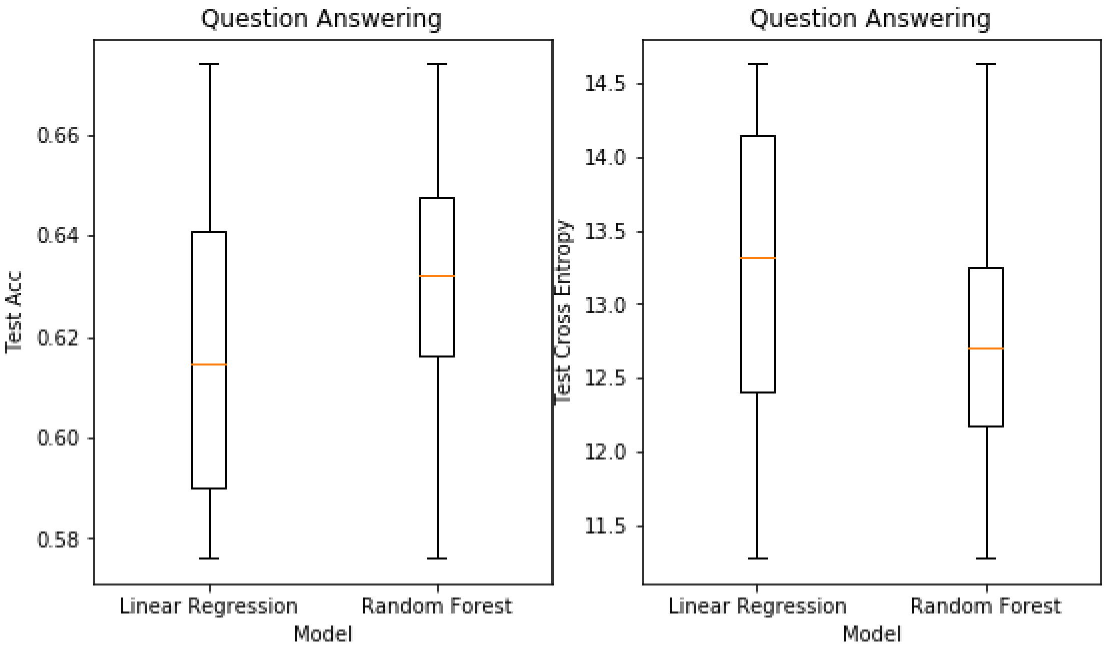
```
For test set using rf model with seed 18: acc = 0.618, f1 = 0.703, cross_ent
ropy = 13.192
For training set using lr model with seed 19: acc = 0.637, f1 = 0.778, cross
_entropy = 12.553
For test set using lr model with seed 19: acc = 0.59, f1 = 0.742, cross_entr
opy = 14.152
For training set using rf model with seed 19: acc = 0.949, f1 = 0.962, cross
_entropy = 1.759
For test set using rf model with seed 19: acc = 0.59, f1 = 0.697, cross_entr
opy = 14.152
```

In [94]:

```
fig, axes = plt.subplots(1,2, figsize=(9,5))
ax = axes[0]
ax.boxplot([lr_test_acc, rf_test_acc])
ax.set_xticklabels(['Linear Regression', 'Random Forest'])
ax.set_ylabel('Test Acc')
ax.set_xlabel('Model')
ax.set_title('Question Answering')

ax = axes[1]
ax.boxplot([lr_test_cross_entropy, rf_test_cross_entropy])
ax.set_xticklabels(['Linear Regression', 'Random Forest'])
ax.set_ylabel('Test Cross Entropy')
ax.set_xlabel('Model')
ax.set_title('Question Answering')
```

Out[94]:

```
Text(0.5, 1.0, 'Question Answering')
```



SC

In [95]:

```python
feature_cat = ['location']
onehot_enc = OneHotEncoder(handle_unknown='ignore')
onehot_enc.fit(data_sc[feature_cat])
sc_cat = onehot_enc.transform(data_sc[feature_cat]).toarray()

feature_num = ['source_label', 'target_label', 'loss', 'test_loss']
scaler = StandardScaler()
scaler.fit(data_sc[feature_num])
sc_num =scaler.transform(data_sc[feature_num])

data_sc['poisoned'] = data_sc['poisoned'].replace({False:0, True:1})

sc_ = pd.concat([pd.DataFrame(sc_cat), pd.DataFrame(sc_num)], axis=1)

features = [
 'start',
 'middle',
 'end',
 'src_label',
 'tgt_label',
 'loss',
 'test_loss']
sc_.columns = features
```

```
C:\Users\CSY\anaconda3\lib\site-packages\ipykernel_launcher.py:11: SettingWithC
opyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stab
le/indexing.html#indexing-view-versus-copy (http://pandas.pydata.org/pandas-doc
s/stable/indexing.html#indexing-view-versus-copy)
  # This is added back by InteractiveShellApp.init_path()
```

In [96]:

```
lr_test_acc = []
lr_test_cross_entropy = []
rf_test_acc = []
rf_test_cross_entropy = []
for s in range(20):
    X_sc_train, X_sc_test, y_sc_train, y_sc_test = train_test_split(sc_, data_sc
    model_lr = LogisticRegression(random_state=1)
    model_lr.fit(X_sc_train, y_sc_train)
    model_rf = RandomForestClassifier(random_state=1)
    model_rf.fit(X_sc_train, y_sc_train)
    score(model_lr, X_sc_train, y_sc_train, 'training', seed=s)
    acc, f1, cross_entropy = score(model_lr, X_sc_test, y_sc_test, 'test', seed=s
    lr_test_acc.append(acc)
    lr_test_cross_entropy.append(cross_entropy)
    score(model_rf, X_sc_train, y_sc_train, 'training', 'rf', seed=s)
    acc, f1, cross_entropy = score(model_rf, X_sc_test, y_sc_test, 'test', 'rf',
    rf_test_acc.append(acc)
    rf_test_cross_entropy.append(cross_entropy)
```

```
For training set using lr model with seed 0: acc = 0.778, f1 = 0.714, cross_ent
ropy = 7.675
For test set using lr model with seed 0: acc = 0.8, f1 = 0.719, cross_entropy =
6.908
For training set using rf model with seed 0: acc = 1.0, f1 = 1.0, cross_entropy
= 0.0
For test set using rf model with seed 0: acc = 0.789, f1 = 0.642, cross_entropy
= 7.292
For training set using lr model with seed 1: acc = 0.793, f1 = 0.731, cross_ent
ropy = 7.164
For test set using lr model with seed 1: acc = 0.767, f1 = 0.656, cross_entropy
= 8.059
For training set using rf model with seed 1: acc = 1.0, f1 = 1.0, cross_entropy
= 0.0
For test set using rf model with seed 1: acc = 0.744, f1 = 0.61, cross_entropy
= 8.827
For training set using lr model with seed 2: acc = 0.781, f1 = 0.704, cross_ent
ropy = 7.547
For test set using lr model with seed 2: acc = 0.822, f1 = 0.758, cross_entropy
= 6.14
For training set using rf model with seed 2: acc = 1.0, f1 = 1.0, cross_entropy
= 0.0
For test set using rf model with seed 2: acc = 0.733, f1 = 0.625, cross_entropy
= 9.21
For training set using lr model with seed 3: acc = 0.8, f1 = 0.716, cross_entro
py = 6.908
For test set using lr model with seed 3: acc = 0.811, f1 = 0.738, cross_entropy
= 6.524
For training set using rf model with seed 3: acc = 1.0, f1 = 1.0, cross_entropy
= 0.0
For test set using rf model with seed 3: acc = 0.744, f1 = 0.667, cross_entropy
= 8.827
For training set using lr model with seed 4: acc = 0.793, f1 = 0.731, cross_ent
ropy = 7.164
For test set using lr model with seed 4: acc = 0.767, f1 = 0.687, cross_entropy
= 8.059
For training set using rf model with seed 4: acc = 1.0, f1 = 1.0, cross_entropy
```

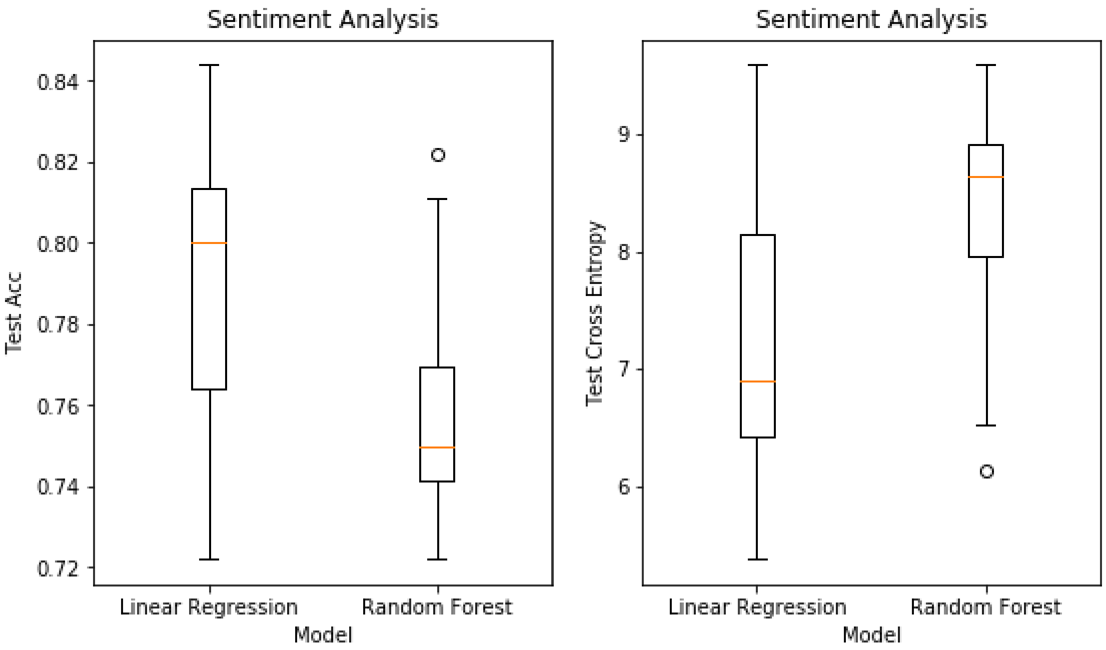
```
= 0.0
For test set using rf model with seed 4: acc = 0.811, f1 = 0.73, cross_entropy
= 6.524
For training set using lr model with seed 5: acc = 0.774, f1 = 0.708, cross_ent
ropy = 7.803
For test set using lr model with seed 5: acc = 0.8, f1 = 0.719, cross_entropy =
6.908
For training set using rf model with seed 5: acc = 1.0, f1 = 1.0, cross_entropy
= 0.0
For test set using rf model with seed 5: acc = 0.756, f1 = 0.676, cross_entropy
= 8.443
For training set using lr model with seed 6: acc = 0.789, f1 = 0.716, cross_ent
ropy = 7.292
For test set using lr model with seed 6: acc = 0.811, f1 = 0.761, cross_entropy
= 6.524
For training set using rf model with seed 6: acc = 1.0, f1 = 1.0, cross_entropy
= 0.0
For test set using rf model with seed 6: acc = 0.767, f1 = 0.712, cross_entropy
= 8.059
For training set using lr model with seed 7: acc = 0.807, f1 = 0.737, cross_ent
ropy = 6.652
For test set using lr model with seed 7: acc = 0.733, f1 = 0.667, cross_entropy
= 9.21
For training set using rf model with seed 7: acc = 1.0, f1 = 1.0, cross_entropy
= 0.0
For test set using rf model with seed 7: acc = 0.722, f1 = 0.648, cross_entropy
= 9.594
For training set using lr model with seed 8: acc = 0.778, f1 = 0.703, cross_ent
ropy = 7.675
For test set using lr model with seed 8: acc = 0.822, f1 = 0.771, cross_entropy
= 6.14
For training set using rf model with seed 8: acc = 1.0, f1 = 1.0, cross_entropy
= 0.0
For test set using rf model with seed 8: acc = 0.733, f1 = 0.667, cross_entropy
= 9.21
For training set using lr model with seed 9: acc = 0.811, f1 = 0.727, cross_ent
ropy = 6.524
For test set using lr model with seed 9: acc = 0.767, f1 = 0.712, cross_entropy
= 8.059
For training set using rf model with seed 9: acc = 1.0, f1 = 1.0, cross_entropy
= 0.0
For test set using rf model with seed 9: acc = 0.722, f1 = 0.627, cross_entropy
= 9.594
For training set using lr model with seed 10: acc = 0.778, f1 = 0.706, cross_en
tropy = 7.675
For test set using lr model with seed 10: acc = 0.833, f1 = 0.754, cross_entrop
y = 5.756
For training set using rf model with seed 10: acc = 1.0, f1 = 1.0, cross_entrop
y = 0.0
For test set using rf model with seed 10: acc = 0.744, f1 = 0.657, cross_entrop
y = 8.827
For training set using lr model with seed 11: acc = 0.811, f1 = 0.756, cross_en
tropy = 6.524
For test set using lr model with seed 11: acc = 0.733, f1 = 0.636, cross_entrop
y = 9.21
For training set using rf model with seed 11: acc = 1.0, f1 = 1.0, cross_entrop
y = 0.0
```

```
For test set using rf model with seed 11: acc = 0.767, f1 = 0.696, cross_entrop
y = 8.059
For training set using lr model with seed 12: acc = 0.815, f1 = 0.752, cross_en
tropy = 6.396
For test set using lr model with seed 12: acc = 0.722, f1 = 0.648, cross_entrop
y = 9.594
For training set using rf model with seed 12: acc = 1.0, f1 = 1.0, cross_entrop
y = 0.0
For test set using rf model with seed 12: acc = 0.733, f1 = 0.647, cross_entrop
y = 9.21
For training set using lr model with seed 13: acc = 0.774, f1 = 0.719, cross_en
tropy = 7.803
For test set using lr model with seed 13: acc = 0.756, f1 = 0.676, cross_entrop
y = 8.443
For training set using rf model with seed 13: acc = 1.0, f1 = 1.0, cross_entrop
y = 0.0
For test set using rf model with seed 13: acc = 0.822, f1 = 0.742, cross_entrop
y = 6.14
For training set using lr model with seed 14: acc = 0.77, f1 = 0.69, cross_entr
opy = 7.931
For test set using lr model with seed 14: acc = 0.833, f1 = 0.776, cross_entrop
y = 5.757
For training set using rf model with seed 14: acc = 1.0, f1 = 1.0, cross_entrop
y = 0.0
For test set using rf model with seed 14: acc = 0.767, f1 = 0.677, cross_entrop
y = 8.059
For training set using lr model with seed 15: acc = 0.793, f1 = 0.741, cross_en
tropy = 7.164
For test set using lr model with seed 15: acc = 0.733, f1 = 0.625, cross_entrop
y = 9.21
For training set using rf model with seed 15: acc = 1.0, f1 = 1.0, cross_entrop
y = 0.0
For test set using rf model with seed 15: acc = 0.744, f1 = 0.61, cross_entropy
= 8.827
For training set using lr model with seed 16: acc = 0.77, f1 = 0.702, cross_ent
ropy = 7.931
For test set using lr model with seed 16: acc = 0.8, f1 = 0.735, cross_entropy
= 6.908
For training set using rf model with seed 16: acc = 1.0, f1 = 1.0, cross_entrop
y = 0.0
For test set using rf model with seed 16: acc = 0.8, f1 = 0.727, cross_entropy
= 6.908
For training set using lr model with seed 17: acc = 0.763, f1 = 0.695, cross_en
tropy = 8.187
For test set using lr model with seed 17: acc = 0.844, f1 = 0.794, cross_entrop
y = 5.373
For training set using rf model with seed 17: acc = 1.0, f1 = 1.0, cross_entrop
y = 0.0
For test set using rf model with seed 17: acc = 0.756, f1 = 0.686, cross_entrop
y = 8.443
For training set using lr model with seed 18: acc = 0.815, f1 = 0.731, cross_en
tropy = 6.396
For test set using lr model with seed 18: acc = 0.789, f1 = 0.765, cross_entrop
y = 7.292
For training set using rf model with seed 18: acc = 1.0, f1 = 1.0, cross_entrop
y = 0.0
For test set using rf model with seed 18: acc = 0.778, f1 = 0.73, cross_entropy
```

```
= 7.675
For training set using lr model with seed 19: acc = 0.77, f1 = 0.708, cross_ent
ropy = 7.931
For test set using lr model with seed 19: acc = 0.8, f1 = 0.735, cross_entropy
= 6.908
For training set using rf model with seed 19: acc = 1.0, f1 = 1.0, cross_entrop
y = 0.0
For test set using rf model with seed 19: acc = 0.744, f1 = 0.623, cross_entrop
y = 8.827
```

In [97]:
```
fig, axes = plt.subplots(1,2, figsize=(9,5))
ax = axes[0]
ax.boxplot([lr_test_acc, rf_test_acc])
ax.set_xticklabels(['Linear Regression', 'Random Forest'])
ax.set_ylabel('Test Acc')
ax.set_xlabel('Model')
ax.set_title('Sentiment Analysis')

ax = axes[1]
ax.boxplot([lr_test_cross_entropy, rf_test_cross_entropy])
ax.set_xticklabels(['Linear Regression', 'Random Forest'])
ax.set_ylabel('Test Cross Entropy')
ax.set_xlabel('Model')
ax.set_title('Sentiment Analysis')
```

Out[97]:
```
Text(0.5, 1.0, 'Sentiment Analysis')
```



NER

In [98]:

```
feature_cat = ['source_label', 'target_label']
onehot_enc = OneHotEncoder(handle_unknown='ignore')
onehot_enc.fit(data_ner[feature_cat])
ner_cat = onehot_enc.transform(data_ner[feature_cat]).toarray()

feature_num = ['loss', 'test_loss']
scaler = StandardScaler()
scaler.fit(data_ner[feature_num])
ner_num =scaler.transform(data_ner[feature_num])

data_ner['poisoned'] = data_ner['poisoned'].replace({False:0, True:1})

ner_ = pd.concat([pd.DataFrame(ner_cat), pd.DataFrame(ner_num)], axis=1)

features = [
 'src_1',
 'src_3',
 'src_5',
 'src_7',
 'tgt_1',
 'tgt_3',
 'tgt_5',
 'tgt_7',
 'loss',
 'test_loss']
ner_.columns = features
```

```
C:\Users\CSY\anaconda3\lib\site-packages\ipykernel_launcher.py:11: SettingWithC
opyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stab
le/indexing.html#indexing-view-versus-copy (http://pandas.pydata.org/pandas-doc
s/stable/indexing.html#indexing-view-versus-copy)
  # This is added back by InteractiveShellApp.init_path()
```

In [99]:

```
lr_test_acc = []
lr_test_cross_entropy = []
rf_test_acc = []
rf_test_cross_entropy = []
for s in range(20):
    X_ner_train, X_ner_test, y_ner_train, y_ner_test = train_test_split(ner_, dat
    model_lr = LogisticRegression(random_state=1)
    model_lr.fit(X_ner_train, y_ner_train)
    model_rf = RandomForestClassifier(random_state=1)
    model_rf.fit(X_ner_train, y_ner_train)
    score(model_lr, X_ner_train, y_ner_train, 'training', seed=s)
    acc, f1, cross_entropy = score(model_lr, X_ner_test, y_ner_test, 'test', seed
    lr_test_acc.append(acc)
    lr_test_cross_entropy.append(cross_entropy)
    score(model_rf, X_ner_train, y_ner_train, 'training', 'rf', seed=s)
    acc, f1, cross_entropy = score(model_rf, X_ner_test, y_ner_test, 'test', 'rf'
    rf_test_acc.append(acc)
    rf_test_cross_entropy.append(cross_entropy)
```

```
For training set using lr model with seed 0: acc = 0.685, f1 = 0.0, cross_en
tropy = 10.873
For test set using lr model with seed 0: acc = 0.611, f1 = 0.0, cross_entrop
y = 13.432
For training set using rf model with seed 0: acc = 1.0, f1 = 1.0, cross_entr
opy = 0.0
For test set using rf model with seed 0: acc = 0.562, f1 = 0.253, cross_entr
opy = 15.137
For training set using lr model with seed 1: acc = 0.66, f1 = 0.0, cross_ent
ropy = 11.726
For test set using lr model with seed 1: acc = 0.685, f1 = 0.0, cross_entrop
y = 10.873
For training set using rf model with seed 1: acc = 1.0, f1 = 1.0, cross_entr
opy = 0.0
For test set using rf model with seed 1: acc = 0.611, f1 = 0.308, cross_entr
opy = 13.432
For training set using lr model with seed 2: acc = 0.679, f1 = 0.0, cross_en
tropy = 11.087
For test set using lr model with seed 2: acc = 0.63, f1 = 0.0, cross_entropy
= 12.792
For training set using rf model with seed 2: acc = 1.0, f1 = 1.0, cross_entr
opy = 0.0
For test set using rf model with seed 2: acc = 0.586, f1 = 0.23, cross_entro
py = 14.285
For training set using lr model with seed 3: acc = 0.658, f1 = 0.0, cross_en
tropy = 11.797
For test set using lr model with seed 3: acc = 0.691, f1 = 0.0, cross_entrop
y = 10.66
For training set using rf model with seed 3: acc = 1.0, f1 = 1.0, cross_entr
opy = 0.0
For test set using rf model with seed 3: acc = 0.617, f1 = 0.205, cross_entr
opy = 13.219
For training set using lr model with seed 4: acc = 0.673, f1 = 0.0, cross_en
tropy = 11.3
For test set using lr model with seed 4: acc = 0.648, f1 = 0.0, cross_entrop
y = 12.153
For training set using rf model with seed 4: acc = 1.0, f1 = 1.0, cross_entr
```

```
opy = 0.0
For test set using rf model with seed 4: acc = 0.617, f1 = 0.295, cross_entr
opy = 13.219
For training set using lr model with seed 5: acc = 0.685, f1 = 0.0, cross_en
tropy = 10.873
For test set using lr model with seed 5: acc = 0.611, f1 = 0.0, cross_entrop
y = 13.432
For training set using rf model with seed 5: acc = 1.0, f1 = 1.0, cross_entr
opy = 0.0
For test set using rf model with seed 5: acc = 0.562, f1 = 0.237, cross_entr
opy = 15.137
For training set using lr model with seed 6: acc = 0.658, f1 = 0.0, cross_en
tropy = 11.797
For test set using lr model with seed 6: acc = 0.691, f1 = 0.0, cross_entrop
y = 10.66
For training set using rf model with seed 6: acc = 1.0, f1 = 1.0, cross_entr
opy = 0.0
For test set using rf model with seed 6: acc = 0.537, f1 = 0.194, cross_entr
opy = 15.99
For training set using lr model with seed 7: acc = 0.679, f1 = 0.013, cross_
entropy = 11.087
For test set using lr model with seed 7: acc = 0.636, f1 = 0.0, cross_entrop
y = 12.579
For training set using rf model with seed 7: acc = 1.0, f1 = 1.0, cross_entr
opy = 0.0
For test set using rf model with seed 7: acc = 0.574, f1 = 0.289, cross_entr
opy = 14.711
For training set using lr model with seed 8: acc = 0.681, f1 = 0.0, cross_en
tropy = 11.015
For test set using lr model with seed 8: acc = 0.623, f1 = 0.0, cross_entrop
y = 13.005
For training set using rf model with seed 8: acc = 1.0, f1 = 1.0, cross_entr
opy = 0.0
For test set using rf model with seed 8: acc = 0.574, f1 = 0.258, cross_entr
opy = 14.711
For training set using lr model with seed 9: acc = 0.663, f1 = 0.0, cross_en
tropy = 11.655
For test set using lr model with seed 9: acc = 0.679, f1 = 0.0, cross_entrop
y = 11.087
For training set using rf model with seed 9: acc = 1.0, f1 = 1.0, cross_entr
opy = 0.0
For test set using rf model with seed 9: acc = 0.574, f1 = 0.127, cross_entr
opy = 14.711
For training set using lr model with seed 10: acc = 0.679, f1 = 0.0, cross_e
ntropy = 11.087
For test set using lr model with seed 10: acc = 0.63, f1 = 0.0, cross_entrop
y = 12.792
For training set using rf model with seed 10: acc = 1.0, f1 = 1.0, cross_ent
ropy = 0.0
For test set using rf model with seed 10: acc = 0.531, f1 = 0.174, cross_ent
ropy = 16.203
For training set using lr model with seed 11: acc = 0.665, f1 = 0.0, cross_e
ntropy = 11.584
For test set using lr model with seed 11: acc = 0.673, f1 = 0.0, cross_entro
py = 11.3
For training set using rf model with seed 11: acc = 1.0, f1 = 1.0, cross_ent
ropy = 0.0
```

```
For test set using rf model with seed 11: acc = 0.623, f1 = 0.265, cross_ent
ropy = 13.005
For training set using lr model with seed 12: acc = 0.679, f1 = 0.0, cross_e
ntropy = 11.087
For test set using lr model with seed 12: acc = 0.63, f1 = 0.0, cross_entrop
y = 12.792
For training set using rf model with seed 12: acc = 1.0, f1 = 1.0, cross_ent
ropy = 0.0
For test set using rf model with seed 12: acc = 0.58, f1 = 0.244, cross_entr
opy = 14.498
For training set using lr model with seed 13: acc = 0.677, f1 = 0.0, cross_e
ntropy = 11.158
For test set using lr model with seed 13: acc = 0.636, f1 = 0.0, cross_entro
py = 12.579
For training set using rf model with seed 13: acc = 1.0, f1 = 1.0, cross_ent
ropy = 0.0
For test set using rf model with seed 13: acc = 0.623, f1 = 0.265, cross_ent
ropy = 13.005
For training set using lr model with seed 14: acc = 0.658, f1 = 0.0, cross_e
ntropy = 11.797
For test set using lr model with seed 14: acc = 0.691, f1 = 0.0, cross_entro
py = 10.66
For training set using rf model with seed 14: acc = 1.0, f1 = 1.0, cross_ent
ropy = 0.0
For test set using rf model with seed 14: acc = 0.58, f1 = 0.333, cross_entr
opy = 14.498
For training set using lr model with seed 15: acc = 0.648, f1 = 0.0, cross_e
ntropy = 12.153
For test set using lr model with seed 15: acc = 0.722, f1 = 0.0, cross_entro
py = 9.594
For training set using rf model with seed 15: acc = 1.0, f1 = 1.0, cross_ent
ropy = 0.0
For test set using rf model with seed 15: acc = 0.722, f1 = 0.43, cross_entr
opy = 9.594
For training set using lr model with seed 16: acc = 0.652, f1 = 0.0, cross_e
ntropy = 12.01
For test set using lr model with seed 16: acc = 0.71, f1 = 0.0, cross_entrop
y = 10.021
For training set using rf model with seed 16: acc = 1.0, f1 = 1.0, cross_ent
ropy = 0.0
For test set using rf model with seed 16: acc = 0.642, f1 = 0.356, cross_ent
ropy = 12.366
For training set using lr model with seed 17: acc = 0.681, f1 = 0.0, cross_e
ntropy = 11.015
For test set using lr model with seed 17: acc = 0.623, f1 = 0.0, cross_entro
py = 13.005
For training set using rf model with seed 17: acc = 1.0, f1 = 1.0, cross_ent
ropy = 0.0
For test set using rf model with seed 17: acc = 0.586, f1 = 0.247, cross_ent
ropy = 14.285
For training set using lr model with seed 18: acc = 0.656, f1 = 0.012, cross
_entropy = 11.868
For test set using lr model with seed 18: acc = 0.704, f1 = 0.0, cross_entro
py = 10.234
For training set using rf model with seed 18: acc = 1.0, f1 = 1.0, cross_ent
ropy = 0.0
For test set using rf model with seed 18: acc = 0.623, f1 = 0.299, cross_ent
```

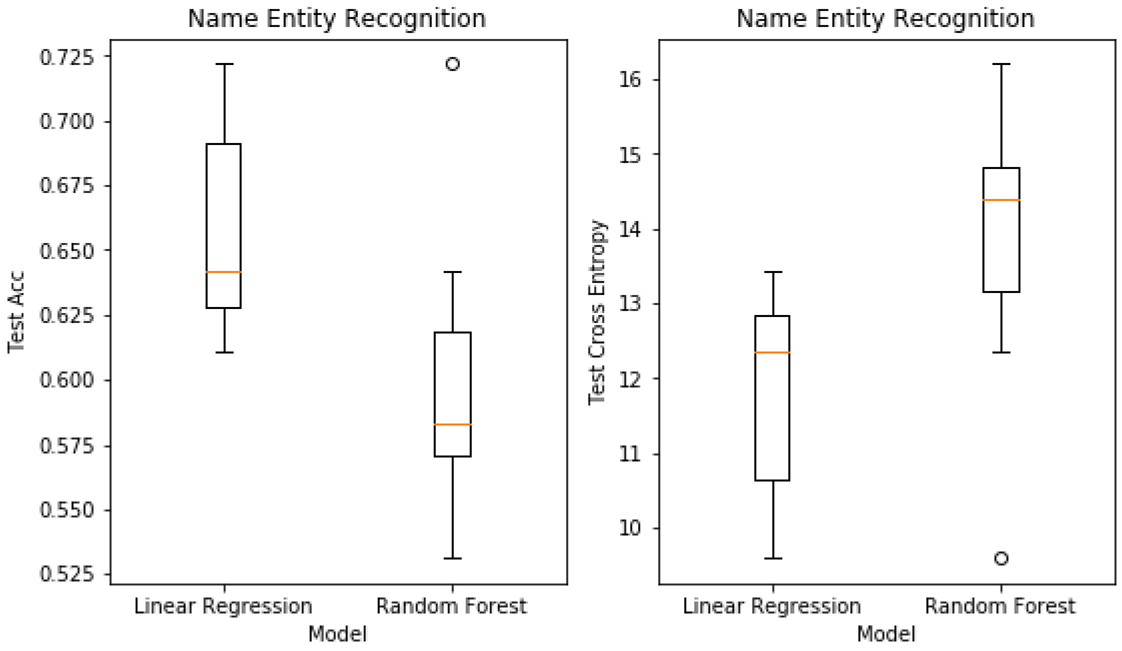
```
ropy = 13.005
For training set using lr model with seed 19: acc = 0.681, f1 = 0.0, cross_e
ntropy = 11.015
For test set using lr model with seed 19: acc = 0.611, f1 = 0.0, cross_entro
py = 13.432
For training set using rf model with seed 19: acc = 1.0, f1 = 1.0, cross_ent
ropy = 0.0
For test set using rf model with seed 19: acc = 0.556, f1 = 0.217, cross_ent
ropy = 15.351
```

In [100]:

```python
fig, axes = plt.subplots(1,2, figsize=(9,5))
ax = axes[0]
ax.boxplot([lr_test_acc, rf_test_acc])
ax.set_xticklabels(['Linear Regression', 'Random Forest'])
ax.set_ylabel('Test Acc')
ax.set_xlabel('Model')
ax.set_title('Name Entity Recognition')

ax = axes[1]
ax.boxplot([lr_test_cross_entropy, rf_test_cross_entropy])
ax.set_xticklabels(['Linear Regression', 'Random Forest'])
ax.set_ylabel('Test Cross Entropy')
ax.set_xlabel('Model')
ax.set_title('Name Entity Recognition')
```

Out[100]:

```
Text(0.5, 1.0, 'Name Entity Recognition')
```



In [ ]: