On Computing Numbers, with an Application to Problems of our Society

Journalism ♥ Computer Science

Stefan Wehrmeyer

Investigative Journalism

Investigative Journalism & Computers

Natural Language Processing

Machine Learning


Los Angeles Times
LAPD misclassified more than 25,000 serious crimes as minor, audit finds
09 DÄPPER
http://www.latimes.com/local/crime/la-me-lapd-audit-20151206-story.html

Social Network Analysis

Algorithmic Accountability

Journalism & science

Train the classifiers

For this analysis we used two machine learning classifiers. The first is a linear support vector machine from the stellar scikit-learn Python library. The second is a maximum entropy classifier. For the official analysis I used the MegaM optimization package to dramatically improve the training speed. Here, for simplicity, I'm using the NLTK built in trainer.

In [6]:
```
# Train our classifiers. Let's start with Linear SVC
# Make a data prep pipeline
pipeline = Pipeline([
    ('tfidf', TfidfTransformer()),
    ('linearsvc', LinearSVC()),
])
# make the classifier
linear_svc = SklearnClassifier(pipeline)
# Train it
linear_svc.train(features)
```

Out[6]:
```
<SklearnClassifier(Pipeline(steps=[('tfidf', TfidfTransformer(norm=u'l2', smooth_idf=True, subline
ar_tf=False,
         use_idf=True)), ('linearsvc', LinearSVC(C=1.0, class_weight=None, dual=True, fit_intercep
t=True,
     intercept_scaling=1, loss='squared_hinge', max_iter=1000,
     multi_class='ovr', penalty='l2', random_state=None, tol=0.0001,
     verbose=0))]))>
```

In [7]:
```
# Next, let's do the Maximum Entropy
maxent = MaxentClassifier.train(features)
```

```
  ==> Training (100 iterations)

      Iteration    Log Likelihood    Accuracy
      ---------------------------------------
```

https://github.com/datadesk/lapd-crime-classification-analysis

Software engineering in the newsroom

Applications for our society.

Thank you.

Stefan Wehrmeyer
correctiv.org
mail@stefanwehrmeyer.com