Assignment 3:

Due October 14

(10pt) Develop an English name tagger using the training and test data under “English\_data”.

Pick one of your favorite supervised learning algorithms (e.g., HMM, MaxEnt, SVMs, CRFs, Neural Networks), train a classifier from the training data (“train\_nwire”). Run the trained tagger on the test data (“test\_nwire”). Report Precision, Recall and F-score based on 5-folder cross-validation and do error analysis. Further extra credits will be given to:

* Using features beyond the features described in the lecture 6.
* Trying more than one algorithm and combine the results.

Some toolkits:

* MaxEnt:
  + http://homepages.inf.ed.ac.uk/lzhang10/maxent\_toolkit.html
* SVMs:
  + <http://svmlight.joachims.org/>
  + https://www.csie.ntu.edu.tw/~cjlin/libsvm/
* CRFs:
  + Java: http://mallet.cs.umass.edu/sequences.php
  + Python: http://crfpp.googlecode.com/svn/trunk/doc/index.html
* Neural Networks:
  + Java: DeepLearning4J ([http://deeplearning4j.org/](https://exchange.rpi.edu/owa/redir.aspx?REF=TL28H6miHH2_xbWaS0PaqWtdtHYtw2QN5ZlF0lzm9hXjtkcAO-nTCAFodHRwOi8vZGVlcGxlYXJuaW5nNGoub3JnLw..))
  + Python: Theano ([http://deeplearning.net/software/theano/](https://exchange.rpi.edu/owa/redir.aspx?REF=nhpZjLHjX2Nym18_1682QqhXsZ1c4ZHAXsEwppSlkYPjtkcAO-nTCAFodHRwOi8vZGVlcGxlYXJuaW5nLm5ldC9zb2Z0d2FyZS90aGVhbm8v))
  + TensorFlow ([https://www.tensorflow.org/](https://exchange.rpi.edu/owa/redir.aspx?REF=2OnKK7dI7GAd-rZCq2-COqgl8r76bZtivyHj-m1yqR7jtkcAO-nTCAFodHRwczovL3d3dy50ZW5zb3JmbG93Lm9yZy8.))
  + Lua: Torch ([http://torch.ch/](https://exchange.rpi.edu/owa/redir.aspx?REF=AecFx-_uT1KCSwmT3e3MZuSbjmUFnCdwo2_e22bRGNvjtkcAO-nTCAFodHRwOi8vdG9yY2guY2gv))
  + Matlab: Caffe ([http://caffe.berkeleyvision.org/](https://exchange.rpi.edu/owa/redir.aspx?REF=9ko-nRzQYmCJpSfAcVHKsdRPTv3-mUVXvUhM2L20L4DjtkcAO-nTCAFodHRwOi8vY2FmZmUuYmVya2VsZXl2aXNpb24ub3JnLw..))

Command examples of Using Mallet CRFs package:

In the Mallet package, a simple interface SimpleTagger has been provided. Assuming you have generated a training feature table in which each token is represented in the following format, and each sentence is followed by an empty line:

* feature1 feature2 ... featuren label

You can use SimpleTagger to train a CRFs model based on the training feature table by the following command:

* cc.mallet.fst.SimpleTagger -train true -model-file crf sample

where crf is the file the trained model is written to. And sample is the pre-defined training feature table.

After the model is trained, you can test your model on the test data:

* cc.mallet.fst.SimpleTagger -model-file crf stest

where stest is the test feature table.