SI 671/721 (Fall 2019) Data Mining: Methods and Applications

Instructor: Paramveer Dhillon

Homework 2:
Sequence Mining for Named Entity Recognition

Due: 10/28/2019

1 Summary

In this homework you will perform Named Entity Recognition (NER) using different classifiers. NER is the task of identifying named entities labels such as People, Location, Organization, Miscellaneous entity etc. in text. NER is a "sequence classification task" as we saw in class i.e. there is dependence between labels of adjacent words.

The dataset used for this homework is the Reuters Newswire dataset, which contains Reuters news stories between August 1996 and August 1997. The dataset has already been preprocessed via tokenization, part-speech tagging, chunking, and named entity tagging. Your task is to infer the named entity tags of the word in the test set.

We recommend using Jupyter Notebooks and Python libraries Keras, Sci-kit learn, and Pandas for this homework.

2 Data Description

The file `DataNer.txt` contains the data for this homework. It contains a total of 14349 sentences with one word per line. Sentence boundaries are represented by empty lines. Each line contains the word followed by its part of speech (POS) tag, its chunking tag, and finally the named entity tag. The tagging follows the standard BIO encoding. Here's a Wiki link describing BIO encoding: `https://bit.ly/2LRQMFn`. Figure 1 shows a snapshot of the data.

You should use 70% of the data for training the various models and the remaining 30% for testing. This split can be easily achieved in Python via the command `train_test_split(X, y, test_size = 0.3)`.

3 Methods [100 points]

Your objective is to build a NER system which classifies the named entities in previously unseen text. The system needs to annotate each word in the testing data with one of the

U.N.	NNP	I-NP	I-ORG
official	NN	I-NP	O
Ekeus	NNP	I-NP	I-PER
heads	VBZ	I-VP	O
for	IN	I-PP	O
Baghdad	NNP	I-NP	I-LOC
.	.	O	O

Figure 1: Sample data with Part of Speech, Chunking, and NER Tags in that order.

8 possible classes, including **'I-ORG', 'O', 'I-MISC', 'I-PER', 'I-LOC', 'B-LOC', 'B-MISC', and 'B-ORG'**. You are encouraged to use the `Sci-kit` and `Keras` Python libraries to build the following suite of models and compare them:

- SGD Linear Classifier.
- Multinomial Naive Bayes Model Classifier.
- Perceptron Classifier.
- Conditional Random Fields (CRF).
- Bidirectional LSTM-CRF model.

We recommend that you follow the following steps while building your models:

- Preprocess the data before building the various deep learning or machine learning models, if needed.
- Fit the suite of models described above.
- Evaluate the performance of various models by comparing their F1 scores as well as generating the classification report.
- For the Bidirectional LSTM-CRF model, plot the loss function and accuracy also.

The main deliverable for this homework is a step-by-step analysis of your model building exercise describing clearly how you processed your dataset, how you tuned the model hyper-parameters (if any) via cross-validation, and finally, why you chose one model over the other. Your Jupyter notebook should contain the reproducible code for training various models as well as text descriptions of your conclusions after each step.

Your grade in this homework will depend on the accuracy of your model on the test dataset as well as your step-by-step description of how you arrived at your final model. We will evaluate your model using the "flat classification report" generated by Sci-kit/Keras. A sample classification report is shown in Figure 2 and it can be generated by the Python command `flat_classification_report()`.

	precision	recall	f1-score	support
I-ORG	0.90	0.86	0.88	2997
I-PER	0.92	0.94	0.93	3368
I-MISC	0.91	0.85	0.88	1341
I-LOC	0.92	0.93	0.93	2522
B-LOC	1.00	0.40	0.57	5
B-MISC	1.00	0.27	0.43	11
B-ORG	1.00	1.00	1.00	4
micro avg	0.91	0.90	0.91	10248
macro avg	0.95	0.75	0.80	10248
weighted avg	0.91	0.90	0.91	10248

Figure 2: A sample "flat classification report".

4 Submission

All submissions should be made electronically by **11:59 PM EST on October 28, 2019**.

Here are the main deliverable files:

- HTML version of your Jupyter notebook.
- The actual Jupyter notebook with "step-by-step analysis" for all the three parts of the homework. It's fine to submit everything in a single notebook. (So that we could replicate your results.).

5 Academic Honesty

Unless otherwise specified in the homework, all submitted work must be your own original work. Any excerpts, statements, or phrases from the work of others must be clearly identified as a quotation, and a proper citation provided. Any violation of the University's policies on Academic and Professional Integrity may result in serious penalties, which might range from failing a homework, to failing a course, to being expelled from the program. Violations of academic and professional integrity will be reported to the concerned authorities. Consequences of academic misconduct are determined by the faculty instructor; additional sanctions may be imposed.