Assignment 3: Named Entity Recognition

Anand Mohan - SR No: 14417 - M Tech(SE)
moghan.anand@gmail.com

1 Introduction

Named Entity Recognition (NER) task takes an input sequence of tokens in a sentence and assign tags for each token or phrase in the sequence.

1.1 Conditional Random Field (CRF)

A class of sequence modeling method used for structured prediction. CRF is a type of discriminative undirected probabilistic graphical model used to encode known relationships between observations and construct consistent interpretations.

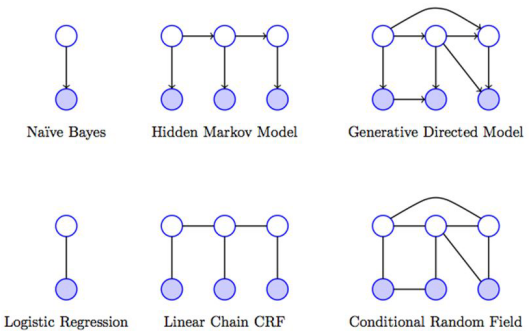

Figure 1: Generative-Discriminative Pairs (Source: Google)

1.2 Models

Mallet: *SimpleTagger* class of mallet is a command line interface to the MALLET Conditional Random Field (CRF) class. This takes in the training data, makes the CRF model which can be used to predict test data tags.

Bi-LSTM with CRF: A sequence to sequence model which implements a Bi-LSTM layer and CRF layer. Input is integer encoded sequences which is embedded by the embedding layer and given to the Bi-LSTM and the output of which is fed to CRF layer. The CRF layer gives a probability distribution as its output which is then used to predict the output tag.

1.3 Handling Unknowns

All tokens not present in the train and present in the test are taken as unknowns and a zero vector of the same embedding size is used to represent such words.

1.4 Metric

$$precision = \frac{tp}{(tp + fp)}$$

$$recall = \frac{tp}{(tp + fn)}$$

where tp is the number of true positives, fp the number of false positives and fn the number of false negatives.

$$F1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}}$$

Macro Score: Calculate metrics for each label, and find their unweighted mean. This does not take label imbalance into account.

Weighted Score: Calculate metrics for each label, and find their average, weighted by support.

2 Methodology

2.1 Mallet

Train and test splits are made. Train data is fed into Mallet's *SimpleTagger* class and model is made for different features. Features includes the tokens as such, token embeddings, POS tags and so on. Same test data is used to test for each of the models made by different combinations of feature types. Embeddings used are from pre-trained Word2Vec model fine tuned on training data. Embedding sizes are varied and ablation study is also conducted.

2.2 Bi-LSTM with CRF

Inputs are encoded into integer values. Max length of sentences are fixed at 60 and short sentences are zero padded. (zero represents unknown token). Tag 'O' is assigned to all the zero paddings. The encoded input is split into batches of size 64 and is given as input to an embedding layer of size 200 which is followed by a Bi-LSTM layer of size 100 and dropout 0.2, the output of which is given to CRF layer. The model is trained for 5 epochs. For each of the epochs train and validation accuracies are calculated. Testing is also done the same way with padded sentences.

3 Results

3.1 Mallet

Features	F1 (Macro)	F1 (Wtd)
Token	0.511	0.852
Emb(100)	0.749	0.914
Token + Emb(100)	0.748	0.914
Token + POS	0.652	0.889

3.2 Bi-LSTM with CRF

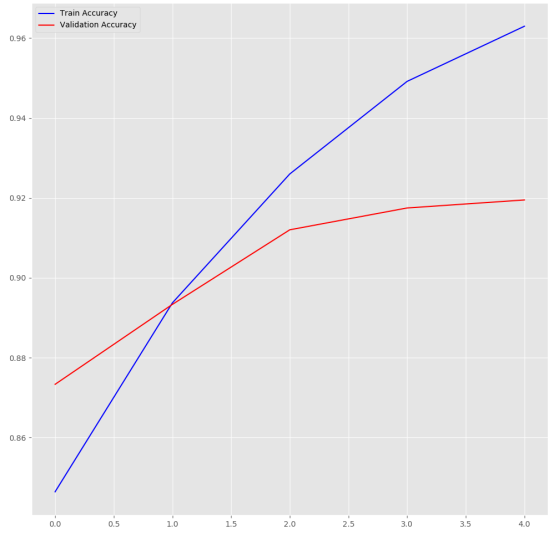

Figure 2: Training and Validation Accuracy vs Epochs

Confusion Matrix:

	O	D	T
O	20926	129	49
D	140	362	9
T	153	33	157

F1 Score (Macro) - 0.749
F1 Score (Wtd) - 0.975

4 Observations

- Embedding features in Mallet gives a major change in accuracy since embeddings capture a lot of semantic as well as syntactic meaning which can be used for identification.
- Adding POS tags with the token and Embeddings didn't make much improvement in performance.
- Increasing Embedding sizes in both models improved the performance.
- Neural CRF model with Bi-LSTM outperforms the CRF model by Mallet.