CS563 - NLP

(Read all the instruction carefully and adhere to them.)

Assignment - 1: NER in Tweets

Deadline: 08th Feb 2019 **Date: 29th Jan 2019**

Named-entity recognition (NER) seeks to locate and classify named entities in text into predefined categories such as the names of persons, organizations, locations etc.

Design a named entity recognition system for Twitter that identifies the presence of named entities in a tweet.

Input: A tokenized sentence.

Output: NER tags for each token of the sentence.

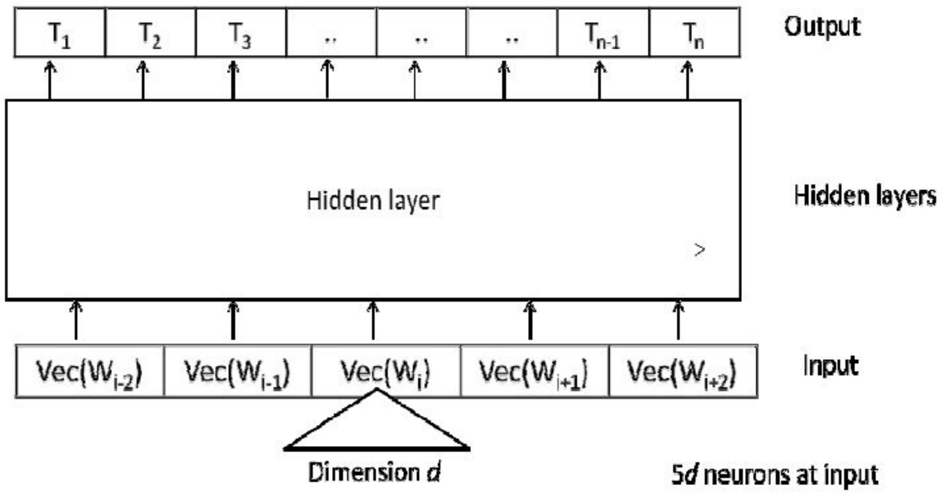
Setups:

1. Identify all the named entity, i.e., whether a token is a named entity or not.
2. First identify all the named entity and then find the types of each name entity.
3. Identify the named entity types in one step.

Approach: Solve the problem of NER through following approaches and compare their performances.

- **Hidden Markov Model (HMM)**
  - You have to implement HMM on your own. Do not use any existing libraries. Calculate emission and transition probabilities and use Viterbi to get the NER sequence.

- **Feed-forward Neural Network:**
  - You may consider following architecture for the implementation.
    - i. Output (Ti): Tags of the NER.
    - ii. Input Vec(W_i): Word embedding for the word W_i. Concatenate contextual words (W_{i-2} …. W_{i+2}) to tag W_i
  - You may use any deep learning libraries such as TensorFlow, PyTorch, Keras etc. for the implementation.



Dataset: Perform 3 fold cross validation on the below datasets and report both average & individual fold results.

- **CS563-NER-Dataset.txt** (Identify the presence of named entity in a tweet.)
- **CS563-NER-Dataset-10Types.txt** (Identify the presence of named entity and classify them into predefined 10 subtypes. 10 Types are *person, product, company, geolocation, movie, music artist, tvshow, facility, sports team and other*.)
- **Format:**
  - Each line contains <Word \t Tag>
  - Sentences are separated with blank line.

Evaluation:

```
perl connlleval.pl -d \\t < predictedTestFile
```

Format of the *predictedTestFile* should be as follows

```
<Token>\t<Actual_Class>\t<Predcited_Class>
```

Submission guidelines:

- Please adhere to following guidelines while submitting your assignment.
- Please submit your assignment **on or before the deadline**.
- Compress all your files **(Input / Output / Codes / Analysis)** in zip file. It should be named as **Roll1_Roll2_Roll3-Assignment-#.zip**
- Please submit your assignment on "https://bit.ly/2CQvzWv"**.**