**EDS 6397 – NLP**

**Assignment 1 – Named Entity Recognition)**

**Varun Vaddi - #2347481**

**INITIAL SETUP:**

Initially, retrieved the **300 tweets** assigned to me from **11,101-11,400** as mentioned in the Roster and saved the file in CSV format as ‘**NER\_Tweets\_Data.csv**’ with column name as ‘**tweets’**.

A screenshot of a computer

Description automatically generated

Copied all the 300 rows from the CSV file and pasted into a text file and saved it as ‘**NLP\_Tweets.txt’.**

**A screenshot of a computer screen

Description automatically generated**

**PHASE-1:**

Uploaded the ‘**NLP\_Tweets.txt’** file into the **NER Annotator tool** and selected ‘**Text separator**’ as ‘**New Line**’ and ‘**Annotation Precision**’ to ‘**Character Level**’.

Added all 7 entity tags by **PERSON, NORP, ORG, GPE, LOC, DATE, MONEY** and started annotating the tweets by assigning the tags to identified Named Entities and clicked on **Save** after each tweet.

A screen shot of a computer

Description automatically generated

**Exported** the Annotations and saved it as file naming as ‘**NER\_Tweets\_300.json**’.

**A screen shot of a computer screen

Description automatically generated**

**PHASE-2:**

**Opened** the JSON file in Jupyter notebook.

**Downloaded** the ‘**en\_core\_web\_lg**’ package and **loaded** it into the spacy model.

Then, I **filtered** only the 7 Tags that we used for the Manual NER Annotation.

A screenshot of a computer

Description automatically generated

**PHASE-3:**

Spacy output format contains a tuple of word & tag **(word, tag)** whereas the manual format contains start index, end index & tag **(start, end, tag)**.

**A screenshot of a computer

Description automatically generated**

Since, both the formats are different, I have developed a new function - **format\_conversion\_from\_spacy\_to\_manual()** to convert spaCy format from word to start & end indexes.

A screenshot of a computer

Description automatically generated

Developed another function – **calculate\_metrics()** to calculate **True Positives(TP), False Positives(FP),** and **False Negatives(FN)** for each tag.

Then used **TP, FN, FP** to calculate the **Precision, Recall** and **F1-score** are calculated for each tag.

A screenshot of a computer

Description automatically generated

**Assessment of SpaCy:**

From my short stint working on this assignment, I have observed that spaCy identifies Named Entities different from how I annotated the tokens manually, reason for our Precision & recall being low.

For example, I have annotated **Cayman Islands** as **GPE**, but SpaCy annotated **the Cayman Islands** as a **GPE**. In other case, I have annotated **Carribean** as **GPE**, but SpaCy identified it as **LOC**.

The Ambiguity of a location/ place being put under GPE/ LOC is very high especially in case of less popular places. We sometimes need to have prior knowledge or info about the news in the tweet inorder to identify correctly the names of cities and persons.

A screenshot of a computer

Description automatically generated