Research Report

Dev Goyal

May 2023

1 Question 1

I picked the paper ”A survey on Named Entity Recognition - Datasets, Tools, and Methodologies” by Basra Jehangir, Saravanan Radhakrishnan , and Rahul Agarwal to lear about the SOTA in NER. This is what I learned from the paper:

Named Entity Recognition (NER) is an important task in Natural Language Processing (NLP) that involves identifying and categorizing named entities in text documents. There are different methodologies for NER, each with its own advantages and drawbacks.

Rule-based NER entails creating a set of rules that define patterns and characteristics of named entities, which are then applied to the text for identification and categorization. The advantage of rule-based NER is its high accuracy and the ability to customize it for specific domains. However, developing and maintaining the rules can be time-consuming and requires expertise. Moreover, rule-based NER may struggle to identify named entities that don’t fit the predefined rules.

Supervised learning for NER involves training a machine learning model on labeled data, where the model learns to identify and categorize named entities based on text features. Supervised learning offers high accuracy and the ability to handle named entities that don’t conform to predefined rules. However, it demands a substantial amount of labeled data for training, which can be both time-consuming and expensive to acquire. Additionally, the model may not perform well on text documents that differ significantly from the training data.

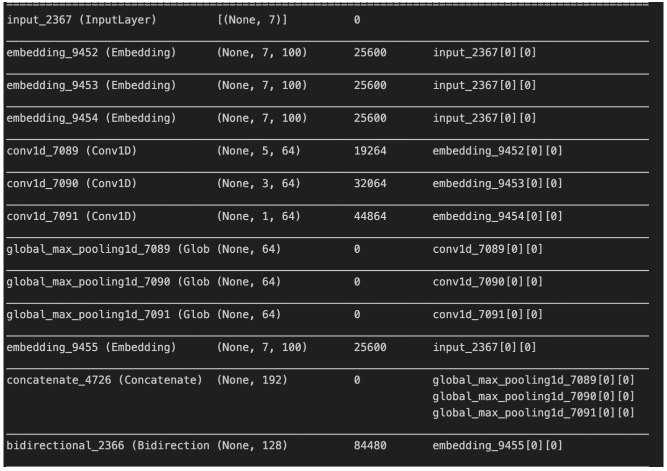
Unsupervised learning for NER focuses on identifying patterns and relationships in text without relying on labeled data. This approach is useful for recognizing named entities that don’t fit predefined rules or aren’t present in the training data. However, unsupervised learning tends to be less accurate than supervised learning and may require additional manual processing to categorize the identified named entities.

To summarize, each NER methodology has its own strengths and weaknesses. Rule-based NER is highly accurate but demands domain expertise and may struggle with nonconforming named entities. Supervised learning offers high accuracy and flexibility but requires a large amount of labeled data. Unsupervised learning can handle diverse named entities but may be less accurate and require additional manual intervention. The choice of methodology depends on the specific NER task requirements and available resources.

2 Question 2

In their paper "Combinatorial feature embedding based on CNN and LSTM for biomedical named entity recognition" Minsoo Cho, Jihwan Ha, Chihyun Park, Sanghyun Park, discuss the bi-LSTM-CRF model for biomedical NER. They proposed passing every word of eevry sentrnce through a CNN and bi-LSTM model to get character level embeddings and then combine those with some pre trained word embeddings for biomedical words to get the 200 dimensional input for the main bi-LSTM-CRF model. After, reading through the paper I planned to replicate the results of their experiment on my own local machine by using tensorflow and python. The only constraint I faced was that I was bounded by the power of my own local machine. This led me to reduce the size of the dataset to just 100 sentences with 2359 word different kinds of words. I was able to replicate almost the same results as the paper. The only difference was that the paper had a F1 score of 0.75 and I got a F1 score of 0.60. I believe this difference is due to the fact that I used a smaller dataset and the paper used a much larger dataset as I was able to match the precision correctly.

The model architecture to generate the word embeddings was as follows:



```
==================================================================================================
input_2367 (InputLayer)         [(None, 7)]          0

__________________________________________________________________________________________________
embedding_9452 (Embedding)      (None, 7, 100)       25600       input_2367[0][0]
__________________________________________________________________________________________________
embedding_9453 (Embedding)      (None, 7, 100)       25600       input_2367[0][0]
__________________________________________________________________________________________________
embedding_9454 (Embedding)      (None, 7, 100)       25600       input_2367[0][0]
__________________________________________________________________________________________________
conv1d_7089 (Conv1D)            (None, 5, 64)        19264       embedding_9452[0][0]
__________________________________________________________________________________________________
conv1d_7090 (Conv1D)            (None, 3, 64)        32064       embedding_9453[0][0]
__________________________________________________________________________________________________
conv1d_7091 (Conv1D)            (None, 1, 64)        44864       embedding_9454[0][0]
__________________________________________________________________________________________________
global_max_pooling1d_7089 (Glob (None, 64)           0           conv1d_7089[0][0]
__________________________________________________________________________________________________
global_max_pooling1d_7090 (Glob (None, 64)           0           conv1d_7090[0][0]
__________________________________________________________________________________________________
global_max_pooling1d_7091 (Glob (None, 64)           0           conv1d_7091[0][0]
__________________________________________________________________________________________________
embedding_9455 (Embedding)      (None, 7, 100)       25600       input_2367[0][0]
__________________________________________________________________________________________________
concatenate_4726 (Concatenate)  (None, 192)          0           global_max_pooling1d_7089[0][0]
                                                                 global_max_pooling1d_7090[0][0]
                                                                 global_max_pooling1d_7091[0][0]
__________________________________________________________________________________________________
bidirectional_2366 (Bidirection (None, 128)          84480       embedding_9455[0][0]
__________________________________________________________________________________________________
```

Clearly this method often beats the SOTA in biomedical NER and is a very good method to use for biomedical NER.

3 Question 4

1. **What have you learned from the exercise above?**

   after going through the whole process of looking into SOTA for a problem and trying to replicate the results, I got to learn a lot about the research process and the different kind of problems involved like scaling down the dataset and hyperparameter tuning. Looking into many different research papers was really interesting and opened about my mind to so many different approaches with their pros and drawbacks.

2. **Do you think the entity extraction results were satisfactory for practical usage? What can be improved?**

   The entity extraction results were good but not perfect for practical usage. Even though, the model had a lot of true positives that is to say the model was able to recogonise a lot of entities correctly, it also incorrectly labeled a lot of entities but one could see why the model did so. Also the model was very domain specific and would not work well for other domains. The model could be improved by using a larger dataset and by using a more complex model like BERT.

3. **What new applications can be built using NER? Propose a cool application you can build that does not exist today.**

   We can try to build a guided learning tool using NER which basically reads a page, extracts the entities and then tries to find the most relevant video on youtube for that topic. This can be used by students to learn about a topic in a more interactive way.