NLP Challenge Report

Akshat Kumar

1 PROBLEM STATEMENT

Named Entity Recognition is a method of extracting the relevant information from a large corpus and classifying those entities into predefined categories. The projects aims to classify the text in the articles in the following three categories:

(1) Location
(2) Organization
(3) Person

2 EXPERIMENTS AND ANALYSIS

The process started with combining data from both the articles which were then later divided into train, test and validation dataset. Sentences were formed by taking an assumption that they end with full stops. This could be improved by writing a complex regex. Then POS tagging was done using a European language parser and tags were later fed in complex models for training. The visualization explored the dataset and came out with some interesting findings. After we have processed the text and obtained a clean corpus, we can infer from the Wordcloud that the article mostly talks about historical Italian cities like Bolzano, Merano, Trento, Rom and Italian currency Lira. Topic modelling using LDA showed different topic that were present in the articles. Lexical dispersion threw some light on the changing context in the article by measuring different words homogeneity across the parts of corpus. From the frequency graph, we can see that high frequency words are dominated by locations like Italian cities followed by person such as Josef and Maria and then by organization such as regierung. Most of the sentences are in the range of 5-20 and the distribution is close to normal. Baseline Decision Tree model was then applied to get reference scores which can be compared later with complex models. Model performed badly as the precision and recall values of most of the classes were 0. Conditional Random field without and with tuning was then applied to learn the context because it takes into account neighbouring samples as well. Feature extraction was done keeping in mind the features of the word and neighbours too like the sequence containing POS tags. Best hyperparameters were obtained by performing hypertuning with RandomizedSearchCV and doing 3 fold cross validation using the training data. Finally BiDirectional LSTM model was applied because it has access to the past as well as the future information and hence the output is generated from both the past and future context.

3 RESULTS AND EVALUATION

(1) Precision, recall and F1 score are taken as the evaluation metric because the data is really skewed because most of the tags lie in 'O' category, that's why taking accuracy would not be correct. So it is an imbalanced classification problem. Precision, recall and F1 score are sensitive to imbalanced data, therefore perfect for such kind of problem.
(2) Compared to the Random Forest classifier, the CRF classifier did better as the scores were improved. However, the precision and recall metrics of the classes individually did not improve. Maybe the model remembered words and not taking into the context information completely.
(3) The average score for the hypertuned CRF model increased and the individual precision and recall scores also improved than a normal CRF model. The model understood the context well and not just remembered the words.
(4) We can observe that for the tuned CRF, precision scores for all the classes have significantly improved with LOC and PER performing the best. Recall scores improved certainly after training but not as much as the precision scores. However, F1 score has a respectable score now for such a noisy dataset.
(5) As we can see from the scores, Bi-LSTM model performs badly than hypertuned CRF scores. Here we had only 2 articles for diving the train, validation and test dataset, that meant we had very few sentences for training and prediction. So feeding more articles and sentences into the deep neural network models will lead to a great trained classifier as deep learning accuracy increases by power law with increase in data.
(6) The models especially Bi-LSTM are suffering from model performance mismatch problem in which the the model is performing well on training dataset but not on test dataset. This could be because of unrepresentative data sample in which the dataset size is too small or the examples in the sample do not effectively cover the cases observed in the broader domain.

4 IMPROVEMENTS AND FUTURE WORK

So, after going over the evaluation part and our results, these are the major points that can help in betters scores for this kind of NER dataset.

(1) Using larger dataset. Here we had only 2 articles for diving the train, validation and test dataset, that meant we had very few sentences for training and prediction. For deep learning algorithms like LSTM, performance grows according to a power law with the increase in the amount of data. So feeding more articles and sentences into the deep neural network models will lead to better scores
(2) Current implementation considers only 2 hyper parameters in CV search, however, the CRF model offers more parameters which can be further tuned to improve on the performance.
(3) Using pre-trained word embeddings like BERT and fine-tuning them
(4) Using character level embedding for LSTM.
(5) Changing model hyperparameters like the number of epochs, embedding dimensions, batch size, dropout rate, activations and so on.
(6) Hypertuning can be done on a wide search space which is resource and computationally intensive and can be done on powerful cloud servers and with GPU.