NLP - Mini 1

Named Entity Recognition

Abstract

The project involved implementation of a simple binary classifier that can determine whether a token is part of a persons name using various techniques for feature extraction, feature indexing, optimization, etc. The data used in this project is derived from the CoNLL 2003 Shared Task on Named Entity Recognition(Tjong Kim Sang and De Meulder, 2003).

1 Introduction

The project emphasizes on efficient feature engineering for the task of NER. We extract various features for the words in the training set. Using a bag-of-words results in a sparse feature set with too many zeroes in the input that would result in increased memory utilization and training time and hence we use various indicator and orthographic features which increase the efficiency of the training model and logistic regression is used for classifying the input data.

2 Feature Engineering

Using a sparse feature set, results in a very large vector space as the size of vocabulary is too large. Hence we index the features and map them to axes. The below features have been considered for the task of named entity recognition.

- **t** - current word/token
- **lowercase(t)** - current word/token in lower case
- **isFirstUpper(t)** - Check if the first character in the word is in CAPS
- **Suffix -n(t)** - Determines the last n letters in the word
- **ContainsDigit(t)** - Check if the token contains digits
- **POS(t)** - Part-of-Speech tag of the word
- **punctuation(t)** - Check if the token contains any punctuation marks

The same features for the previous (t-1) and next word (t+1) are also considered. Since the features are strings or boolean variables, we vectorize them where each word is mapped to a specific index in an array and the same is referenced. Using all the above features resulted in 136547 features for the given training set.

Features Used	Accuracy
word	50.9
word + lower	61.8
word + POS tag	70.4
word + POS(current,prev,next)	86.9
word + lower + POS(current,prev,next)	88.5
All features	90.7

Table 1: Accuracy on Dev Dataset

3 Model

We use a logistic regression model, an efficient method for classifying binary data. The logistic function is the standard sigmoid function which takes values between 0 and 1. Since this is a binary classification problem, those with probability greater than 0.5 are classified as belonging to one class and the those with less than 0.5 fall in the other class. The model is trained for 100 epochs and the weights learned are used for classifying tokens on the development and test data set. Stochastic Gradient Descent (SGD) optimizer is used for optimizing the logistic regression function.

4 Results

As we see in the table 1, it can be observed that as we increase the feature size, the accuracy of the model too increases. Since most of the names are tagged as NNP or NNPS, we get 70% accuracy by considering only the part of speech tag of the word. Since there are other proper names such as countries, organizations, we need other features to distinguish names. Most of the names are followed by verbs and hence we consider the next word and their corresponding POS tag too.

5 Conclusion

An accuracy of 90.7% is achieved on the development set and it can be concluded that using meaningful features helps in improving the accuracy of the model.