COMSM0089 Introduction to Data Analytics Coursework

QJ22919/ 2334033

|  |  |
| --- | --- |
| **Task 2: Named Entity Recognition** | |
| * 1. **Design and Implement Method** | |
| **2.1.1 Overview** | |
| Named entity recognition (NER) is finding spans of text that constitute proper names and tagging the type of the entity. (Martin, 2021)  The task is to build a tool which identifies and tags entities from the BioNLP 2004 dataset. The dataset is annotated with 5 types of entity tag - DNA, protein, cell type, cell line and RNA.  NER presents a segmentation problem as, unlike parts of speech tagging, a named entity often encompasses a span of texts rather than just a single word- i.e. Golden Gate Bridge rather than just the noun bridge. | |
| **2.1.2 Approach- Strengths & Limitations** | |
| A Conditional Random Field (CRF) model is used as, unlike Hidden Markov Models (HMM), it considers the whole document. Rather than learning the likelihood of features given the preceding sequence labels ( i.e. generative model of HMM) CRF assigns a probability to a sequence of output tags using all the tokens in the document (Martin, 2021).  CRF use (weighed) global feature functions which are the sum of all local feature functions. Each local feature function can use any / all of the following: other tokens in the document, the length of the document and the current and preceding tag (in linear chain CRF).  CRF is much more effective than HMM but as it utilises the whole document for each feature (rather than just the preceding tags) it is significantly more computationally demanding than HMM and this increases exponentially with the number of features. | |
| **2.1.3 Span Encoding** | |
| When training the CRF model entity spans are encoded using “BIO” tagging. The beginning of a span is given the prefix “B” for beginning and following tokens are given the Prefix “I” for ti indicate it is inside an entity span. The suffix is appended to the entity type so “New York” would be tagged “New”: “B-LOC”, “York”: “I-LOC” as the entity type is location. Tokens outside of the spans that don’t represent an entity are tagged “O” (for outside). BIO tagging allows the model to differentiate entities in a list which could be misconstrued as a single entity if beginning tags were not used.  In the training data the label field would look as follows: | |
| **Document** | **Tags** |
| 'of', 'NADPH', 'oxidase', '(', ' | ['O', 'B-protein', 'I-protein', 'O'] |
| **2.1.4 Data Preparation** | |
| The dataset is already split into train, validation and test sets. Each set consists of a row of data which each represent an article from a medical journal. For each article there is a ‘tokens’ list containing individual tokens (words/ punctuation etc.) and a ‘tags’ list with the corresponding (numeric) tag for that token.  Firstly, the numeric tag (0 – 10) is substituted for it’s actual value (1 becomes “B-DNA”) then a new set is created where each token in a sentence becomes a tuple with its corresponding label. | |
| **2.1.5 Software Implementation** | |
| A baseline performance was found using the CRFTagger from the NLTK package in it’s default configutation. | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |
|  | |