4 Evaluation Metrics

python script evaluation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | BASIC FEATURES | | ADVANCED FEATURES | |
|  |  | LogReg | CRF | LogReg | CRF |
| Twitter\_dev.ner.pred | Token-wise accuracy | 95.54 | 95.77 | 95.76 | 96.07 |
| Token-wise F1 f(macro) | 21.58 | 29.56 | 23.51 | 29.29 |
| Token-wise F1 (micro) | 95.54 | 95.77 | 95.76 | 96.07 |
| Sentence-wise accuracy | 66.61 | 68.64 | 66.27 | 68.64 |
| Twitter\_dev\_test.ner,pred | Token-wise accuracy | 91.02 | 91.31 | 91.50 | 91.71 |
| Token-wise F1 (macro) | 10.92 | 17.98 | 17.36 | 23.45 |
| Token-wise F1 (micro | 91.02 | 91.31 | 91.50 | 91.71 |
| Sentence-wise accuracy | 48.65 | 50.50 | 49.36 | 52.20 |

Conll script evaluation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | BASIC FEATURES | | ADVANCED FEATURES | |
|  |  | LogReg | CRF | LogReg | CRF |
| Twitter\_dev.ner.pred | Accuracy | 95.54 | 95.77 | 95.76 | 96.07 |
| Precision | 49.61 | 60.61 | 48.9 | 62.50 |
| Recall | 16.89 | 26.81 | 23.86 | 33.51 |
| FB1 | 25.20 | 37.17 | 32.07 | 43.63 |
| Twitter\_dev\_test.ner.pred | Accuracy | 91.02 | 91.31 | 91.50 | 91.71 |
| Precision | 32.35 | 46.82 | 31.89 | 44.51 |
| Recall | 8.54 | 15.99 | 14.91 | 23.29 |
| FB1 | 13.51 | 23.84 | 20.32 | 30.58 |

First of all, both methods are useful to do evaluation job. CONNL and python evaluation can get good results for this application.

About the differences,

The CONLL is a chunk evaluation method and its result includes NER Accuracy, Precision, Recall, FB1 for all classes and also given the same information for every class.

the python evaluation can do feature engineering and I can add any feature I interested. The result includes token wise accuracy, token wise F1 (micro), token wise F1 (macro) and sentence wise accuracy are got for all classes. What’s more, the result also has every class’s recall, precision and F1 (including B- and I- tags). Due to more useful and detail data given by the python evaluation script, I think that is better.