Übungen 3: Sequence Tagging with CRF

Abgabedatum Sonntag, 6. Mai 2018, 14:00h MEZ

1 Named Entity Recognition auf deutschen Texten

In dieser Aufgabe werden wir einen Sequence Tagger bauen, der Named Entities in Texten erkennt. Eure Aufgabe ist es, gute Features zu finden um einen möglichst hohen F1 score zu erlangen. Die Daten wurden in der 2014 Edition vom GermEval[1] als Shared Task zur Verfügung gestellt. Auf der Website findet ihr alle Informationen zu den Daten.

1.1 Verstehen der Codevorlage

1. In den Unterlagen findet ihr das Notebook `CRF with Sklearn.ipynb`. Dort findet ihr eine einfache Baseline auf der ihr aufbauen könnt. In der Baseline verwenden wir die `sklearn-crfsuite`, studiert die Dokumentation [2].
2. Ihr dürft auch andere CRF libraries verwenden, falls euch das einfacher fällt. Ein Beispiel wäre Wapiti[3].
3. Wir stellen euch die Features eines Brown-Clusterings[4] zur Verfügung, welches auf dem deutschen Wikipedia trainiert wurde.

1.2 Aufgabe

In dieser Aufgabe geht es darum Features für das Sequence Tagging zu finden. Im Gegensatz zur letzten Übung, in der ihr ~~Satz~~-basierte Features entwickelt habt, geht es hier darum zusätzlich ~~Wort-basierte~~ Features zu suchen (z.B. Gross-oder kleinschreibung einzelner Wörter). Ähnlich wie in Übung 2 sollt ihr:

- Neue Features entwickeln.
- Modelle testen: Di(Sk)lean library duchgehen und den Classifier finden, der am besten Funktioniert.
- Eure eigenen Ideen testen.

Die Daten kommen in der Form eines Trainingsets, eines Developmentsets (bzw. Validierungset) und eines Testsets. Verwendet zum Entwickeln und Tunen des Systems das Train und Devset. Evaluiert anschliessend euer System auf dem Testset.
Im File `TextBerg10Saetze.tsv` findet ihr einige (te)~~te~~ aus dem Text+Berg Korpus, lasst diese von eurem CRF Taggen und korrigiert den Output dann *manuell* auf NER-Ebene LOC und PERS.

[1] https://www.lt.informatik.tu-darmstadt.de/de/data/german-named-entity-recognition/
[2] `https://sklearn-crfsuite.readthedocs.io/en/latest/`
[3] https://wapiti.limsi.fr/
[4] http://www.aclweb.org/anthology/R15-1016

2 Abgabe

- Als Abgabe erwarten wir einen kurzen Bericht über euer System: welche Features habt ihr verwendet, welche Modelle, etc. Es soll nachvollziehrbar sein, dh jemand sollte in der Lage sein den Bericht zu lesen und euer System nachzubauen. Der Bericht sollte entweder in PDF oder in Word Format abgegeben werden.
- Den Score den ihr auf dem Testset erreicht. Das bedeutet auch, dass ihr für verschiedene Modelle den Testset score reportet. Das ist insofern wichtig, damit ihr diesen Score mit dem auf dem Validierungset erreichten Score vergelicht.
- Den Code, damit wir bei Unklarheiten besser nachvollziehen können, was ihr gemacht habt.
- Die getaggten Text+Berg Sätze mit eurer Korrektur. Das Format sollte aus 3 Spalten bestehen, dem Wort, dem Tag und eurer Korrektur.
- Die Abgabe erfolg über ein Zip File, wo die beiden oben genannten Datein enthalten sind. Stellt sicher, dass ihr in der Abgabe die Namen und Kürzel aller Teammitglieder reinschreibt.

3 Feedback

Bitte melde kurz zurück, welche Aufgaben du hilfreich und welche du nicht hilfreich fandest, um dein Lernen zu unterstützen.

Viel Spass und Erfolg beim Programmieren!