Stanford named entity recognition

Marek Medveď

Faculty of Informatics, Masaryk University
xmedved1@mail.muni.cz

Abstrakt In this paper we describe the basics of named entity recognition and subscribe Stanford system for named entity recognition.

1 Úvod

V dnešnej dobe je oblasť dolovania informácií z textu veľmi žiadaná. Named entity recognition (ďalej NER) je jednou z podoblastí dolovania informácií z textu. Jej hlavou úlohou je klasifikácia tzv. pomenovaných entít, nachádzajúcich sa v texte, do rôznych tried. Príkladom týchto tried môžu byť: person, date, localization, organization a ďalšie.

Na základe tejto klasifikácie je možné následne určiť o čom daný článok pojednáva, alebo určiť, či je postoj k danej pomenovanej entite negatívny, alebo pozitívny (napr. vďaka oblasti NER je teda možné zistiť, či je firma hodnotená v článku pozitívne alebo negatívne).

Ďalšia možná oblasť využitia NER je oblasť zaoberajúca sa zistením odpovede z textu na vopred zadanú otázku (question answering), pretože práve pomenovaná entita (výstup NER) je najčastejšou odpoveďou na väčšinu otázok.

2 Stanford named entity recognition

Stanford NER [1](tiež známy ako CRFClassifier) je systém vytvorený na Stanfordskej univerzite, určený na identifikovanie pomenovaných entít v texte a ich klasifikácii. Systém je implementovaný v jazyku Java a základná verzia systému rozoznáva tri triedy PERSON, ORGANIZATION, LOCATION.
Stanford NER pozostáva:

- **Conditional Random Fields (CRF)**
- **Všeobecné vlastnosti textu**
- **Trénovacie dáta**

2.1 Conditional Random Fields (CRFs)

Conditional Random Fields je štatistická metóda, ktorá sa často využíva v strojovom učení, kde je používaný na štruktúrovanú predikciu. Táto metóda je používaná na zakódovanie relácii medzi slovami v texte. V NER je dôležitá pri určovaní triedy daného slova.

CRFs je typ diskriminačného grafického modelu. To znamená, že triedu danej entity algoritmus určí na základe zistených rozdielov medzi vstupmi.

V procese priraďovania triedy skúmanému slovu táto metóda neskúma len samotné slovo, ale aj kontext tohto slova a na základe okolitých slov a ich značiek (pokiaľ sú už určené), ktoré môžu ovplyvniť výsledok, priradí triedu.

Vo výsledku je teda táto metóda presnejšia, ako ostatné HMM modely (vďaka vplyvu kontextu na určovanie triedy).

2.2 Vlastnosti slov

2.2.1 Vzhľad slova (wordshape)

Jednou z vlastností, ktoré uľahčujú rozhodovanie pri určovaní triedy je vzhľad slova (word shape). Táto vlastnosť slova sa kóduje pomocou troch skupín značiek. Každé veľké písmeno je mapované na veľké X, malé písmeno je mapované na malé x, číslo sa mapuje na malé d a symboly ako :, _, ., atď. sa mapujú na samé seba.

Ak má slovo dĺžku nanajvýš 4 znaky potom je zobrazený celý jeho word shape. Napr. Ahoj -> Xxxx.

Avšak ak je slovo dlhšie ako 4 znaky, tak sa stred slova označí iba pomocou množiny značiek, ktoré reprezentujú toto zoskupenie. Napr. Varicecllа–zoster -> Xx-xxx.

2.2.2 Kódovanie slov na triedy pomenovaných entít

Existujú dva typy kódovaní a to IO kódovanie a IOB kódovanie. Rozdiel medzi týmito dvoma kódovaniami je v tom, že zatiaľ čo IOB kódovanie rozlišuje medzi entitou A a entitou B tej istej triedy, tak IO kódovanie vôbec.

Pr. majme vetu: Fred showed Sue Mangqiu Huang's ...

```
IO kódovanie označí   IOB kódovanie
SUE -> PER            Sue -> A\_PER
Mangqiu -> PER        Mangqiu -> B\_PER
Huang's -> PER        Huang's -> I\_PER  //pokračovanie pre
                                           B person
```

Pre Stanford NER je však využívaná IO namiesto IOB. Dôvodom je, že IO je rýchlejšie (IOB obsahuje 2e+1 značiek zatiaľ čo IO iba e+1 značiek). A zároveň situácia aká bola prezentovaná v príklade nenastáva až tak často, pretože sa zväčša v kontexte pri sebe vyskytujú entity z rôznych tried. Taktiež presnosť IOB klasifikátorov nie je úplne stopercentná a teda by náš príklad dopadol nasledovne:

```
IOB kódovanie:
Sue -> A\_PER
Mangqiu -> I\_PER //pokračovanie pre B person
Huang's -> I\_PER //pokračovanie pre B person
```

2.3 Trénovacie dáta

Pre vytvorenie sekvenčného (spracováva celý reťazec) modelu pre NER potrebujeme reprezentatívnu vzorku trénovacích dokumentov. To znamená, že každé slovo v trénovacej sade dát musí byť označené príslušnou triedou entity.

Po vytvorení sekvenčného modelu je potreba natrénovať sekvenčný klasifikátor, ktorý určí entitnú triedu pre vstupné dáta.

Literatúra

1. Finkel, J.R., Grenager, T., Manning, C.: Incorporating non-local information into information extraction systems by gibbs sampling (2005) `http://nlp.stanford.edu/~manning/papers/gibbscrf3.pdf`.