Подходи за обработка на естествен език

Зимен семестър, 2019/2020 год.

„Система за разпознаване на именувани същности в неструктурирани правни документи“

Курсов проект

Автор:

Кристиян Боянов Крумов

email: kristiyan.boyanov@gmail.com

ф.н. 26240

февруари, 2020

София

Въведение

Разпознаването на именуваните същности (по нататък в документа ще използваме и NER - Named Entity Recognition) се нарича процесът на автоматичното идентифициране на именуваните същности (напр. имена на хора, дати, часове, валута и т.н.) в неструктурирани или полу-структурирани текстове. Обикновено заедно с идентифицирането на именуваните същности (ИС) се извършва и тяхното класифициране. Утвърдените категории ИС (предимно за новинарски текстове или научно-популярни статии) са: PER - човек, LOC - локация, ORG - организация и OTH - друго.
Изучаването и разработването на методи за NER започва още преди повече от 20 години (зората на Интернет). Това се дължи на огромното количество данни под формата на неструктуриран текст в мрежата и нуждата за извличане на информация от тях. С развитието на сферата се разработват различни решения на този проблем - използване на линейни статистически модели, използване на правила, Hidden Markov Models, supervised методи със запаметяване и др. В днешно време резултати, близки до човешката преценка дават решения с методи като CRF, BiLSTM, ELMO, BERT.

Цел на проекта

Целта на проекта е създаване на система за извличане на информация от правни текстове. Основна функционалност на системата ще бъде разпознаването на именувани единици и използването им както за задаването на референции към релевантни документи (закони, норми, решения, определения и т.н.), така и за подобряването на резултатите от търсенето в големи масиви от такъв тип данни. Например - можем да филтрираме резултати по дадени единици, да визуализираме интересни зависимости или да правим задълбочени анализи.

Подобни системи

Известни са няколко системи, решаващи подобен проблем:
АПИС - предоставя чрез своите компютърни системи знания и информация в областта на българското право, правото на Европейския съюз, икономиката и финансите, защитата на личните данни и неприкосновеността на личността и личния живот, както и справочни данни за стопанските субекти и тяхната бизнес активност.

exLege.info - автоматизирано събира и обработва правна информация като извлича релевантните връзки от нея и ги систематизира в отделна база данни. Това позволява търсенето на съдебни решения чрез съчетание от разпоредби чрез функцията "Свързана практика"

Реализация

Можем да разделим разработката на проекта на пет етапа:

1. Създаване на корпус с правни текстове
2. Анотиране на NEs в корпуса
3. Обучаване на модел за разпознаване на наименувани единици
4. Автоматично анотиране на целия корпус от документи
5. Индексиране в Elastic Search и визуализиране на данните в Kibana

Създаване на анотиран корпус

Първата стъпка в разработката на такъв тип проекти е намирането на достатъчно количество данни. За щастие в мрежата има достатъчно правни документи, като особено количество такива са концентрирани в официалните сайтове на различните съдилища - районни, върховни и т.н.
След реализацията на скрейпър, с помощта на Scrapy (Python фреймуърк за създаване на скрейпъри), успях да сваля 16 000 документа (съдебни решения, определения и др.) с общ размер 900MB.

Анотиране на корпуса

За анотацията на корпуса е използван безплатният софтуер за анотация на текст - Doccano. След разделяне на текстовете на документи от по 4 изречения и зареждането им в Doccano, 500 такива документа бяха ръчно анотирани. След това с помощта на CRF модел, извърших полуавтоматично анотиране на още 500 документа. Учудващо, моделът се справяше доста добре, като маркираше правилно повечето закони, съдилища и дати - моята задача беше редактиране на допуснатите от модела грешки и маркиране на изпуснати единици.
Именуваните единици, анотирани в 1000-те документа са разделени в 17 категории, присъщи за правните документи:

Етикет	Описание	Етикет	Описание
LOC-ADDR	Адреси	PER-JDG	Съдии
LOC-CITY	Градове	PER-OTH	Други лица
LOC-CTRY	Държави	REF-LAW	Закони
LOC-OTH	Други локации	REF-DOC	Юридически документи
ORG-CMPNY	Фирми, компании	DATE	Дати

ORG-COURT	Съдилища	MONEY	Пари
ORG-INST	Институции	METRIC	Метрика (кв. м., км/ч.)
PER-LWR	Адвокат	PER-PLANTIFF	Ищец

Обучаване на модел, разпознаващ именувани същности

Проблемът за разпознаването на именувани същности може да се разгледа като задача за класифициране на поредица от токени. Един подход за решаването на такива задачи е използването на Conditional Random Fields. Една от причините да се спра на този алгоритъм, пред други доказали точността си алгоритми (BERT, ELMO, NNs) е, че по сравнително лесен начин можем да визуализираме наученото от модела и съответно да редуцираме времето по търсене на правилни атрибути (фийчъри) за репрезентацията на всеки токън.
Избраните атрибути от token-vector-a:

word.lower()	word[-2:]	containsDigit	hasQuote	EOS
word[:-2]	ispunctuation	lonleyInitial	endsWithDot	word.isupper
word[:-1]	containsDash	singleChar	fourDigitsYear	word.islower
word[-1:]	containsDot	singleDigit	BOS	word.istitle

Впечатление правят фийчърите за наставки и представки. Тъй като българският език е морфологически богат, подобен тип предикати могат да помогнат за разпознаването на токъни, които не са срещани в обучаващото множество. Например - български имена, завършващи на -ов, ова, т.н. имена на градове и села (-во). Използван е и n-gram похвата - добавяме всички изброени предикати за предната или следващата дума. Ако няма такава добавяме съответно BOS (beginning of sentence) или EOS (end of sentence).

Голяма част от този етап беше тренирането и крос валидацията на модела след изпробването на нов фийчър или регуларизация на алгоритъма, посредством двата ламбда параметъра, участващи в уравнението:

$$L(w) = \sum_{i=1}^{n} \log p(s^i|x^i; w) - \frac{\lambda_2}{2}||w||_2^2 - \lambda_1||w||_1.$$

Резултати

След итеративното подобряване на модела, описано по-горе получаваме следните резултати:

	precision	recall	f1-score	support
DATE	0.67	0.60	0.64	630
LOC-ADDR	0.00	0.00	0.00	229
LOC-CITY	0.67	0.40	0.50	207
LOC-CTRY	0.72	0.77	0.74	86
LOC-OTH	0.00	0.00	0.00	56
METRIC	0.00	0.00	0.00	19
MONEY	0.84	0.80	0.82	382
O	0.95	0.97	0.96	66001
ORG-CMPNY	0.61	0.42	0.50	340
ORG-COURT	0.77	0.79	0.78	987
ORG-INST	0.50	0.04	0.07	26
PER-JDG	0.00	0.00	0.00	12
PER-LWR	0.00	0.00	0.00	18
PER-OTH	0.67	0.55	0.60	756
PER-PLANTIFF	0.00	0.00	0.00	23
REF-DOC	0.74	0.55	0.63	1273
REF-LAW	0.71	0.62	0.66	3108
TIME	0.00	0.00	0.00	10
accuracy			0.93	74163
macro avg	0.44	0.36	0.38	74163
weighted avg	0.92	0.93	0.92	74163

Имайки предвид малкото количество данни и множеството категории, можем да кажем, че резултатите са добри. Ясно се отличават ниските резултати за някои от категориите, като основна причина за това можем да определим неравномерното количество анотирани спрямо другите категории. Т.е. една от стратегиите за подобряването на резултатите със сигурност е анотиране на повече разнородни данни и обогатяване на “бедните” на примери категории.

Изключително интересна е таблицата с атрибутите, които характеризират решенията за класификация на различните категории. Например: за дати, атрибут с голяма тежест е singleDigit и следващият токън да се състои от 4 цифри (т.е. следващата дума да е година). Друг пример е разпознаването на фирми - следващата дума да е АД/ЕТ/ООД.

y=DATE top features		y=LOC-ADDR top features		y=LOC-CITY top features		y=LOC-CTRY top features		y=LOC-OTH top featur
Weight[?]	Feature	Weight[?]	Feature	Weight[?]	Feature	Weight[?]	Feature	Weight[?]	Feature
+3.824	singleDigit	+1.222	word[-1:]:/	+2.133	-1:word.lower():място	+1.255	word[:-2]:Република	+3.005	-1:word.lower():идент
+1.632	+1:fourDigitsYear	+1.003	-1:word.lower():адрес	+1.956	-1:word.lower():гр.	+1.255	word[:-1]:Републикат	+1.414	-1:word.lower():„
+1.542	+1:gazeteerDate	+0.935	containsDigit	+1.748	-1:word.lower():от	+1.255	word.lower():републиката	+0.987	fourDigitsYear
+1.142	word[-2:]:г.	+0.649	-1:gazeteerDate	+1.704	word[:-2]:Соф	+1.067	+1:gazeteerCountry	+0.932	+1:word.lower():''
+1.130	-1:word.lower():м.	+0.637	word[-1:]:7	+1.704	word[:-1]:Софи	+1.067	-1:gazeteerCountry	+0.932	+1:hasQuote
+1.002	word[-2:]:10	+0.610	word[:-1]:	+1.703	word.lower():София	+1.067	gazeteerCountry	+0.792	+1:ispunctuation
+0.859	word[-2:]:04	+0.512	-1:ispunctuation	+1.623	+1:word.lower():населено	+1.002	-1:word.lower():в	+0.621	+1:word.lower():ккkр
+0.831	word[-1:]:и	+0.426	word.lower():град	+1.604	-1:gazeteerDate	+0.935	word[:-1]:Р	+0.560	-1:fourDigitsYear
+0.798	gazeteerDate	+0.426	word[:-1]:гра	+0.865	+1:word.lower():.	+0.602	word[:-2]:Репу6ли	+0.498	word.lower():кккр
+0.754	word[:-2]:20	+0.406	word[:-2]:	+0.553	word.lower():]	+0.602	word[:-1]:Република	+0.498	word[:-2]:КК
+0.752	+1:word.lower():по	+0.339	word[-2:]:[	+0.553	word[-2:]:]	+0.602	word.lower():република	+0.498	word[:-1]:ККК
+0.740	word[:-1]:один	+0.339	word[-1:]:[	+0.553	word[-1:]:]	+0.513	word.istitle()	+0.497	word[-2:]:КР
+0.740	word[:-2]:оди	+0.339	word.lower():[	+0.539	-1:word.lower():,	+0.448	word[-2:]:ка	+0.493	word.lower():централ
+0.739	+1:word.lower():г.	+0.323	-1:word.lower():[	+0.497	word.istitle()	+0.352	word[-2:]:ия	+0.493	word[:-2]:Централ
+0.717	word[-2:]:12	+0.313	+1:word.lower():]	+0.381	word[:-1]:гр	+0.348	+1:word.lower():българ	+0.493	+1:word.lower():скла
+0.664	word[-2:]:и	+0.309	-1:word.lower():,	+0.379	-1:word.lower():–	+0.346	word[:-2]:Българ	+0.493	word[:-1]:Централе
+0.663	word.lower():и	+0.302	+1:word.lower():с	+0.378	word.lower():гр.	+0.346	word[:-1]:Българи	+0.492	-1:word.lower():центр
+0.655	word[-2:]:08	+0.297	+1:word.lower():,	+0.337	-1:word.lower():населено	+0.346	word.lower():българия	+0.492	word[:-1]:скла
+0.596	word[:-2]:19	+0.274	containsDot	+0.337	word[:-2]:мяс	+0.333	word[-1:]:а	+0.492	word[:-2]:скл
+0.567	-1:word.lower():до	+0.274	endsWithDot	+0.337	word.lower():място	+0.310	-1:word.lower():на	+0.492	word.lower():склад
+0.563	word[-2:]:на	+0.269	word[-1:]:]	+0.337	word[:-1]:мяст	+0.254	+1:word.istitle()	+0.487	word.lower():по
+0.515	word.lower():година	+0.269	word.lower():]	+0.337	word[:-2]:г	+0.218	word[-1:]:я	+0.481	word[-2:]:по
+0.514	word[-1:]:г	+0.269	word[-2:]:]	+0.305	word[-2:]:р.	+0.116	+1:word.lower():,	+0.479	word[:-1]:п
… 53 more positive …		+0.265	-1:word.lower():улица	+0.287	word[-1:]:о	+0.054	-1:singleChar	+0.458	word[-1:]:а
… 33 more negative …		+0.257	+1:ispunctuation	+0.263	word[:-2]:населе	+0.024	singleChar	+0.429	word[-1:]:Р
-0.644	singleChar	+0.193	word[-1:]:.	+0.263	word.lower():населено	-0.079	+1:singleChar	+0.426	word[-2:]:ад
-0.650	-1:fourDigitsYear	+0.186	ispunctuation	+0.263	word[:-1]:населен	-1.178	word.islower()	+0.411	-1:word.istitle()
-0.976	word.istitle()	… 37 more positive …		+0.263	+1:word.lower():място			+0.337	-1:word.lower():вели
-1.136	+1:containsDigit	… 16 more negative …		… 13 more positive …				+0.335	+1:word.lower():„
-1.384	word[:-2]:	-0.172	word.istitle()	… 11 more negative …				… 34 more positive …	
-1.897	-1:gazeteerDate	-0.229	-1:singleDigit	-0.500	+1:word.lower():,			… 10 more negative …	
-3.057	-1:containsDigit	-0.745	+1:gazeteerDate	-2.071	-1:containsDigit			-0.463	word[:-1]:

Литература

https://www.aclweb.org/anthology/R09-1022.pdf
https://link.springer.com/chapter/10.1007/978-3-030-33220-4_20#Tab1
https://en.wikipedia.org/wiki/Legal_information_retrieval
http://ebox.nbu.bg/dp25/pdf/02.pdf