LSTM and BERT for Named entity recognition for legal documents in German

Contact data:

Erik Koynov – erik.koynov@gmail.com,

Denitsa Ilieva – ilieva@stud.uni-heidelberg.de,

Samir Essa - Samer.sheichessa@gmail.com

Links:

Git : https://github.com/erik-koynov/LegalNER

Dataset : A Dataset of German Legal Documents for Named Entity Recognition : 29 Mar 2020

Motivation

Law is a text-based discipline[1] and the quantity of various legal documentation (legislative acts, court decisions, administrative acts, etc.) grows hastily, proportionally to the growth of world population and complexity of economic relations between different actors on both the international and the domestic scene. For this reason devising a proper way of storing and retrieving legal information has attracted much academic interest [2, 3].

Aside from the scientific challenges that this task carries, its solution is of immeasurable necessity for all parties involved. Legal practitioners will be enabled to quickly find the relevant acts without having to read through documents which eventually turn out to be irrelevant. A good example of this rather disheartening drawback of imprecise search systems, which fail to adapt to the ever growing database of documentation, is the massive quantity of unimportant court decisions [3] that those search systems return. This downside is even more abysmal if non-lawyers, who are insufficiently affluent (or just not willing) to pay the expensive attorney fees, were using those search engines to retrieve legal documents. These people will most certainly not be able to distinguish between relevant and irrelevant acts and as they are mostly involved in a consumer – producer (or seller) relation their position will become even more disadvantageous. [4]

A step towards creating a more reliable legal information retrieval system is storing the unstructured legal documents – be it judicial decisions, normative acts, orders etc. - in a structured manner. Our belief is that correctly identifying named entities will allow for automatic tabular representation of the legal documentation in continuation to the following work. This in itself will enable indexing massive corpora of legal documents in legal databases by those automatically generated fields, which currently is a tedious task, especially regarding documents from the state archives. Combined with a good enough computer vision solution for reading those archived documents, our project could enable storing the immeasurable quantity of archived documents in a structured manner, without the need of experts to manually annotate the files.

Research topic summary

We model our task of finding legal entities (discussed in the next chapter) as a Named-Entity-Recognition task [5]– which is a special case of text segmentation [6] – that is determining the boundaries of entities of interest inside the text as shown in Fig. 1.

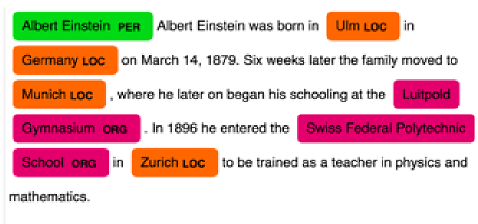

Fig. 1. Output of a named-entity-recognition system.

Traditional named methods like rule-based and template-based require significant cost and over-reliance on steps, such as rule building or feature engineering. The neural network-based approach as a data-driven approach can achieve an end-to-end whole procedure, not like a traditional pipeline and does not rely on feature engineering.

Since named entity recognition is a supervised machine learning task [7] the results acquired depend on the data collected in the training dataset. Since creating a dataset for NER involves manually annotating the texts and required expert knowledge to first define what important entities should be and then finding particular occurrences of such entities inside a text corpus NER solutions are domain specific. In our project we are going to use a domain specific corpus for legal named entities [8]. There is no previous published model on this particular corpus.

NER is usually modeled as a multi-class text sequence classification task on word level, where for each word there is a target label. The usual representation of the target labels is the so called I-O-B tagging, where for each word can either be I (inside), O (outside) or B (beginning) of a named entity with each specific entity defined by the creators of the datasets having its own I and B tags – e.g. PI, ORGI – person-inside, organization-inside respectively.

In recent years deep learning[9] has gained popularity due to its surpassing classical methods for sequence modeling such as conditional random fields [10] in cases where voluminous datasets are available there has been an abundance of deep learning papers on the subject of NER. Popular approaches for solving the sequence modeling tasks in the context of named-entity-recognition are recurrent neural networks (see Fig. 2)and more specifically their improved variant Long Short-Term Memory[11,12], which deals with the problem of vanishing and exploding gradient caused by the unrolling of the recurrent cell especially in the case of longer sequences.

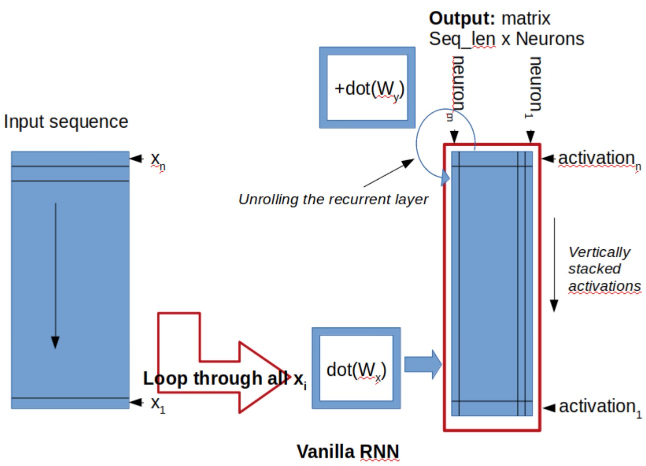

Fig. 2. Classical RNN architecture. In consists of 2 weight matrices W_x (input) and W_y (recurrent). Computing the downstream gradient for the sequence elements near the beginning of the sequence will require multiplying the gradient w.r.t. to that gradient with the gradient w.r.t. to all previous elements- thus values <1 will become smaller, and smaller (vanishing gradient values >1 will grow exponentially (exploding gradient). For small gradients this essentially means that the earlier members of the sequence will not have influence on the training of the model. For the mathematical formulae see [9] at p.375 (10.20) and (10.21).

In more recent years approaches using Bi-directional LSTM [12], which take into account both directions of the sequence and not just from left to right, when computing the final representations have been applied. The advent of the transformer in 2017 and the models based on it such as BERT gave new impetus to the sequence modeling research by showing state-of-the-art results on many different tasks. BERT has also been fine-tuned for named-entity recognition [15, 16].

Project Description

As stated above in this work we will identify named entities in legal texts in German. The entities, defined by the creator of the dataset [8], are the following:

Classes				#	%
f	1	PER	Person	1,747	3.26
f	2	RR	Judge	1,519	2.83
f	3	AN	Lawyer	111	0.21
c	**1**	**PER**	**Person**	**3,377**	**6.30**
f	4	LD	Country	1,429	2.66
f	5	ST	City	705	1.31
f	6	STR	Street	136	0.25
f	7	LDS	Landscape	198	0.37
c	**2**	**LOC**	**Location**	**2,468**	**4.60**
f	8	ORG	Organization	1,166	2.17
f	9	UN	Company	1,058	1.97
f	10	INN	Institution	2,196	4.09
f	11	GRT	Court	3,212	5.99
f	12	MRK	Brand	283	0.53
c	**3**	**ORG**	**Organization**	**7,915**	**14.76**
f	13	GS	Law	18,520	34.53
f	14	VO	Ordinance	797	1.49
f	15	EUN	EU legal norm	1,499	2.79
c	**4**	**NRM**	**Legal norm**	**20,816**	**38.81**
f	16	VS	Regulation	607	1.13
f	17	VT	Contract	2,863	5.34
c	**5**	**REG**	**Case-by-c. regul.**	**3,470**	**6.47**
f	18				
c	**6**	**RS**	**Court decision**	**12,580**	**23.46**
f	19				
c	**7**	**LIT**	**Legal literature**	**3,006**	**5.60**
Total				**53,632**	**100**

Table 2: Distribution of fine-grained (f) and coarse-grained (c) classes in the dataset

Fig. 3. Entities of the dataset (taken from [8]).

The dataset defines 7 coarse-grained entities (classes) and 17 fine-grained entities and uses the I-O-B target representation as shown in Fig. 3.

We will model the task as a many-to-many model as shown in Fig. 4. Where the input is a sentence from a legal document and the output is a sequence of I-O-B tags of the respective entity (class) as shown in Fig. 4.

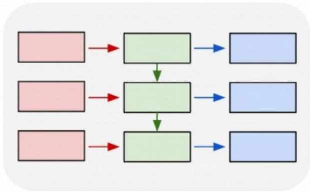

Fig. 4. Many-to-many architecture – i.e. the number of input elements in a sequence is equal to the number of output elements of the model.

We will divide the tasks in a way that allows us to assess different model architectures on this dataset. The approaches we are going to use are the following – we will use LSTM architectures (single and bi-directional) with and without attention units and we will also assess the performance of BERT on the task. The guiding work is [16] where many different approaches for NER are described and applied.

Since BERT is a massive neural network we will use the pretrained BERT available at deepset and accessible with the hugging face library. As word embeddings for the other networks we will use try different approaches – pretrained GloVe[17], Word2Vec[18] (to be selected among many options) and also training embeddings from scratch using the gensim library.

[1] Tamsin Maxwell and Burkhard Schafer. “Concept and Context in Legal Information Retrieval”. In: vol. 189. Jan. 2008, pp. 63–72. doi: 10.3233/978-1-58603-952-3-63.
[2] Judith Dick. “Representation and storage of legal text for conceptual retrieval”. In: (1991).
[3] Marc Van Opijnen and Cristiana Santos. “On the Concept of Relevance in Legal Information Retrieval”. In: (2017).
[4] Erik Koynov. “Three stage statute retrieval algorithm with BERT and hierarchical pretraining” , In: (2020).
[5] Christopher Dozier, Ravikumar Kondadadi. “Named Entity Recognition and Resolution in Legal Text.”. In: (2010).
[6] Pavlina Fragkou. Text Segmentation using Named Entity Recognition and Co-reference Resolution in English and Greek Texts. In: (2016).
[7] David Nadeau, Satoshi Sekine. A survey of named entity recognition and classification. In: (2007)
[8] Elena Leitner, Georg Rehm. A Dataset of German Legal Documents for Named Entity Recognition. In: (2020).
[9] Goodfellow. Deep Learning. In: (2014).
[10] Hidayat Ur Rahman. Disease Named Entity Recognition Using Conditional Random Fields. In: (2009).
[11] Sepp Hochreiter, Juergen Schmidhuber. Long Short-Term Memory. In: (2009).
[12] Jason P.C. Chiu. Named Entity Recognition with Bidirectional LSTM-CNNs. In: (2016).
[13] Ashish Vaswani. Attention Is All You Need. In: (2017).
[14] Jacob Devlin. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: (2018).
[15] Antonia Baumann. Multilingual Language Models for Named Entity Recognition in German and English. In: (2019).
[16] Zhenjin Dai. Named Entity Recognition Using BERT BiLSTM CRF for Chinese Electronic Health Records. In: (2019).
[17] Pennington. GloVe: Global Vectors for Word Representation. In: (2014).
[18] Mikolov. Efficient Estimation of Word Representations in Vector Space. In: (2014).