Conditional Random Field Overview

Friday, January 14, 2022 10:14 AM

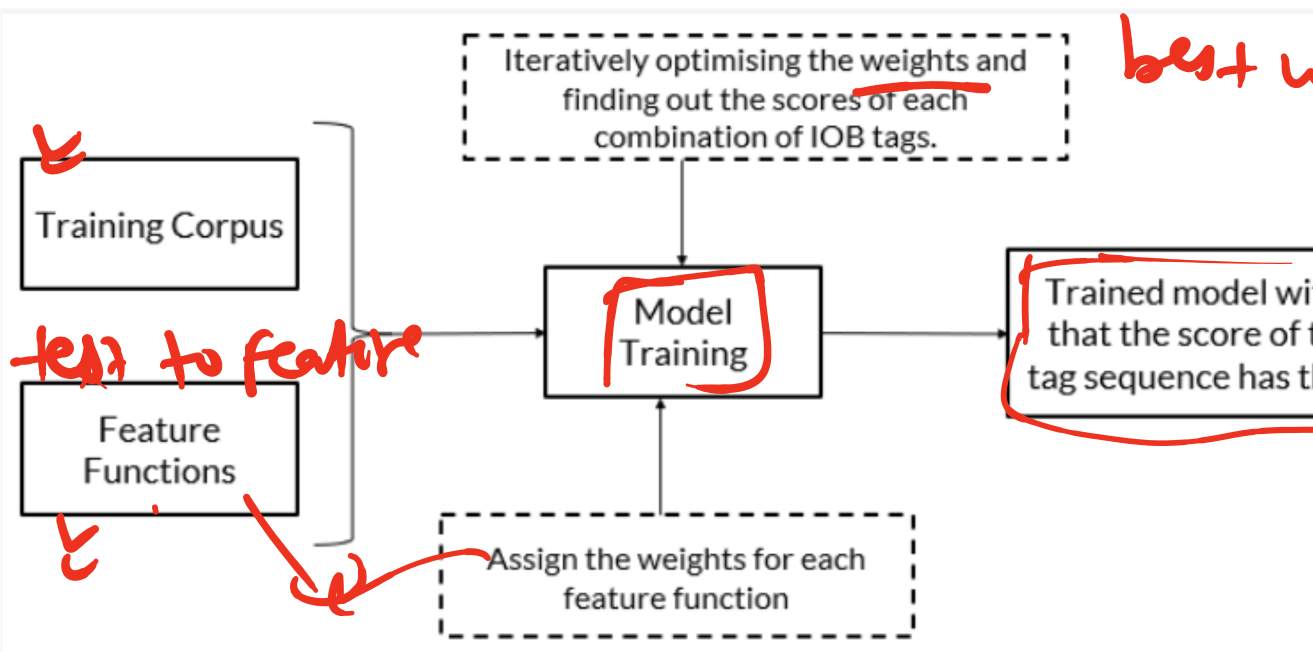
https://github.com/ContentUpgrad/Syntactic-Processing/blob/master/Custom%20NER%20Coc
https://towardsdatascience.com/named-entity-rx``ecognition-and-classification-with-scikit-lea
https://colab.research.google.com/github/dipanjanS/nlp_workshop_odsc19/blob/master/Moc

- 'I drove away in my Jaguar.'
- 'The deer ran away seeing the Jaguar.'

The word 'Jaguar' in the first sentence refers to a car manufacturing company, and in the secor
Python Example

The Conditional Random Field (CRF) can be used as a sequence labelling technique for perfor

Conditional Random Fields are the class of probabilistic models. There are two terms in the CRF
- **Random fields:** These indicate that the CRF is probability a distribution-based mac
- **Conditional:** This indicates that the probabilities are conditional probabilities.



de%20Demonstration/Code%20files/CRF.ipynb
rn-f05372f07ba2
dule03%20-%20Text%20Understanding/Project%20-%20Building%20NER%20Taggers.ipynb

nd sentence, it refers to a species of animal

ming NER tagging. CRF is used to perform custom NER.

F nomenclature that you need to keep in mind, which are as follows:
chine learning model.

weights

th weights such
the correct IOB
he highest score.

[TF1 COP] → WE
(BOW

FE

```
doc = "Google Inc. is headquartered in Silicon Valley."
```

4- feature function

X - token	Y - Right	Y' - wrong	Y''	Y'''
Google	***B-ORG***	B-LOC	B-LOC	B-ORG
Inc	***I-ORG***	I-ORG	I-LOC	I-ORG
is	***O***	O	O	B-ORG
headquartered	***O***	O	O	B-LOC
in	***O***	O	O	B-ORG
Silicon	***B-LOC***	O	B-LOC	B-ORG
Valley	***I-LOC***	O	I-LOC	B-ORG

So, for the input sentence 'X', there can be multiple combinations of IOB tags possible, and son
tagged manually in the training data set The model needs to train itself or assign the weights in

Now, apart from the training data set, you require features and their values to train a m
create features out of this text data, which you can feed into the model.

Feature Functions -
Some commonly used features in the CRF technique for NER applications are as follows:

- You can build logic on the input word 'xi' and on the surrounding words, which co
- The PoS tag of the word 'xi' and surrounding words.
- Is a particular word present in a dictionary (dictionary of common names, dictiona
- Word shapes:
- 26-03-2021 => dd-dd-dddd
- 26 Mar 2021 => dd Xxx dddd
- W.H.O => X.X.X
- Presence of prefix and suffixes

Example -

```
doc = "Google Inc. is headquartered in Silicon Valley."
```

f1 (X, xi, xi-1, i) = 1 if xi= Xx+; otherwise, 0 (Words starting with an uppercase letter)

x	f_1	f_2	f_3	y'
Google	1	0	0	B-Loc
Inc	1	0	0	I-Org
is	0	0	1	|
|	0	0	1	
	0	0	1	/
Silicon	1	0	0	1
Valley	1	1	0	0

ne of them are shown in the above table. The highlighted one is the correct combination, which is
such a way that it assigns the highest score to the correct combination.

machine learning model. Here, the training corpus is nothing but the text data. You need to

uld be ‘xi-1’ or ‘xi+1’.

ary of organic chemicals, etc.)

f1 (X, xi, xi-1, i) = 1 if xi= Xx+; otherwise, 0 (words starting with an uppercase letter)
f2 (X, xi, xi-1, i) = 1 if xi= Noun and xi-1 is Noun; otherwise, 0 (Continuous entity)

;	0q-wuqadfcx8z

- The f1 feature indicates that if a particular word in the given sentence starts with an upp
- The f2 feature indicates that if a particular word in the given sentence has a PoS tag of no word; otherwise, assign 0 as the value of f2 to this word.
- The f3 feature indicates that if a particular word in the given sentence is 'Inc' and the wo as the value of f3 to this word.

So, the model considers all the possible combinations of IOB tags of the given training ex each feature function. The model starts the computation by taking any random initial we where the score of the correct IOB tag sequence is the highest.

Let's first calculate the values of all three features for the Y' combination of IOB tags.

X= Google (B-LOC) Inc (I-ORG) is (O) headquartered (O) in (O) Silicon (O) Valley (O). - y

- f1 (X, xi, xi-1, i) = 1 if xi= Xx+; otherwise, 0 (words starting with an uppercase letter)

	y'	**f1** - Xx+;
Google	B-LOC	**1 (because 'Google' starts with an upperca letter)**
Inc	I-ORG	**1 (because 'Inc' starts with an uppercase letter)**
is	O	**0**
headquartered	O	**0**
in	O	**0**
Silicon	O	**1 (because 'Silicon' starts with an upperca letter)**
Valley	O	**1 (because 'Valley' starts with an uppercase let**

- f2 (X, xi, xi-1, i) = 1 if xi= Noun and xi-1 is Noun; otherwise, 0 (Continuous entity)

ercase letter, then assign 1 as the value of f1; otherwise, assign 0 as the value of f1 to this word.
oun and the word before it also has a PoS tag of noun, then assign 1 as the value of f2 as to this

rd before it has the NER tag of B-ORG, then assign 1 the value of f3 to this word; otherwise, assign 0

ample and calculates the scores of each combination using the weights corresponding to
eights for each feature function and iteratively modifies the weights until it reaches a stage

Regex

ase
)
ase
ter)

		f2
Google	B-LOC	0
Inc	I-ORG	0
is	O	0
headquartered	O	0
in	O	0
Silicon	O	0
Valley	O	1 (because 'Valley' is a noun and 'Silicon is a noun)

- f3 (X, xi, xi-1, i) = 1 if xi = Inc and xi-1 = B-Org; otherwise, 0 (Company names often end

		f3
Google	B-LOC	0
Inc	I-ORG	0 (because the previous word 'Google' does have the B-ORG tag)
is	O	0
headquartered	O	0
in	O	0
Silicon	O	0
Valley	O	0

Weights calculation and optimization -

X= Google (**B-LOC**) Inc (**I-ORG**) is (**O**) headquartered (**O**) in (**O**) Silicon

lso

with Inc)

not

(**O**) Valley (**O**).

$w_1 = 1$ $w_2 = 2$ W

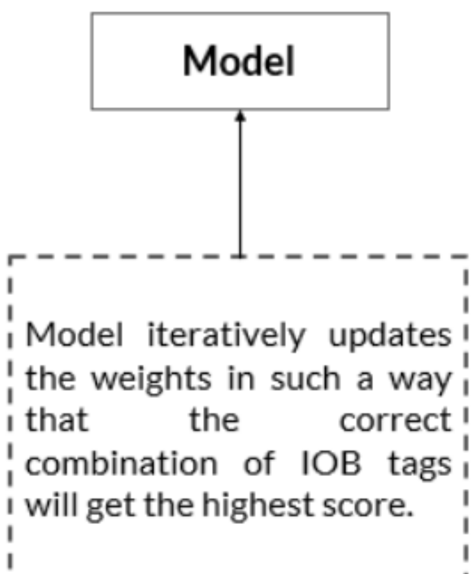
X.	$\hat{y}_2$	f_1*w_1 (w_1=1)	f_2*w_2 (w_2=2)	f_3*w_3 (w_3=3
Google	B-Loc	1	0	0
Inc	I-Org	1	0	0
is	O	0	0	0
headquartered	O	0	0	0
in	O	0	0	0
Silicon	O	1	0	0
Valley	O	1	2	0
SUM		**4**	**2**	**0**

So, the value of the score for this combination of NER tags will be 4+2+0 = 6, which
weights, as shown in the table above.

= 6

We keep continue this exercise -

Correct label



X	Y	Y'
Google	B-Org	B-Loc
Inc	I-Org	I-Org
Is	O	O
headquartered	O	O
in	O	O
Silicon	B-Loc	O
Valley	I-Loc	O

Score 1 = 9   Score 2 = 6

Score1 = 9 is the highest.

X

=1

n you will obtain after multiplying each feature function values with their corresponding

Y''
B-Loc
I-Loc
O
O
O
B-Loc
I-Loc

Score 3 = 6

every sentence — X — tag

↳ POS tag

↳ NER t

pretrained

↳ Spacy

ML

CRF

↳ training

↳ test —

X —

tra

The sentence

g - context is missing

tag - Seach

- with the tags

dowo? - → Predict the tags

G - NER Tag

ning - data

X Y $\hat{y}$ - predi

Noun - $\boxed{Per}$

ig X = Per

Per

Google - B-org

set - CRF - weights max
best tags