



# NER IN SOCIAL MEDIA TEXT

---

e-Faculty Training Program on  
Machine Learning Technologies for Social Media Analytics  
20<sup>th</sup> August 2020

**Dr Ashish Anand**

Department of Computer Science and Engineering

IIT Guwahati



[anand.ashish@iitg.ac.in](mailto:anand.ashish@iitg.ac.in)



<https://www.iitg.ac.in/anand.ashish>

# Outline

- Defining NER
- Solving NER
  - Sequence Labelling Problem
  - Popular Frameworks
- A Case Study

# DEFINING NER

---

# The Task of NER

## Clinical Text Excerpt

This is an 83 year old woman with a history of hypertension. Her chest x-ray revealed mild pulmonary edema. Chest CT scan was negative for pulmonary embolism but positive for consolidation.

Due to her chronic diarrhea she was repeated with intravenous fluids. For her depression, continued on Citalopram 10 mg daily.

■ Problem ■ Test ■ Treatment

# The Task of NER

Yess! Yess! Its official **Nintendo** announced today that they Will release the **Nintendo 3DS** in **north America** march 27 for \$250

■ Org ■ Product ■ Location

# Defining NER as Sequence Labelling Problem

**Given,**

Sequence of input tokens  $s = x_1 \cdots x_n$

Set of labels  $L = \{l_1, \cdots l_k\}$

**Determine,**

a sequence of labels :  $y = t_1 \cdots t_n$

where  $t_i \in L$  for  $1 \leq i \leq n$

## Example : BIO Tagging Scheme

<i>He</i>	<i>was</i>	<i>given</i>	<i>laxis</i>	<i>to</i>	<i>prevent</i>	<i>him</i>	<i>from</i>	<i>congestive</i>	<i>heart</i>	<i>failure</i>	<i>.</i>
O	O	O	B-Treat	O	O	O	O	B-Prob	I-Prob	I-Prob	O

# SOLVING NER

---

# NER as Supervised Learning Problem

Given a training set  $(X, Y) = (s_1, y_1) \cdots (s_n, y_n)$ ,

Where :

$s_i = x_1 \cdots x_{n_i}$  is a word sequence

$y_i = t_1 \cdots t_{n_i}$  is its label sequence

$n_i$  is the length of  $i^{th}$  sequence

Find an approximation/hypothesis  $f : \chi \rightarrow \gamma$

Where :

$\chi$  is set of all word sequence and

$\gamma$  is set of all tag sequence



# Summary of Classical Models

- Rule based or dictionary based
- Hidden Markov Model (HMM)
- Maximum Entropy Markov Model (MEMM)
- Conditional Random Field Model (CRF)
- Support Vector Machine models

# HMM for Sequence Labelling

- HMM is a *generative model*  $p(x_1 \dots x_n, y_1 \dots y_n)$
- HMM have 5 tuples namely :
  - 1 V : Observation symbols (Vocabulary set  $w_1, \dots w_m$  )
  - 2 T : States (Tag set  $l_1 \dots l_n$  )
  - 3 A : State transition probability matrix of  $n \times n$   
 $A_{i,j} = P(S_t = l_i | S_{t-1} = l_j)$
  - 4 B : Output probability matrix of  $n \times m$   
 $B_{i,j} = P(O_t = w_j | S_t = l_i)$
  - 5  $\pi$  : Array of  $n \times 1$ , initial state probability

# HMM: Continued

- Define the probability for any  $p(x_1 \dots x_n, y_1 \dots y_n) \in (\text{training set})$  as  $p(x_1 \dots x_n, y_1 \dots y_n) = \prod_{i=1}^n q(y_i | y_{i-1}) * e(x_i | y_i)$
- **Decoding** : For any word sequence  $(x_1 \dots x_n)$   
 $f(x_1 \dots x_n) = \arg \max_{y_1 \dots y_n \in \mu(n)} p(y_1 \dots y_n | x_1 \dots x_n)$   
 $\mu(n)$  is all possible tag sequence
- We can solve this through *Viterby algorithm* in polynomial time [Rabiner, 1989]
- **Parameters estimation**
  - 1  $A_{i,j} = q(t_i | t_j) = \frac{c(t_j t_i)}{c(t_j)}$
  - 2  $B_{i,j} = e(w_i | t_j) = \frac{c(t_j \mapsto x)}{c(t_j)}$

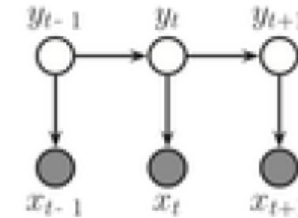


Figure : HMM

# HMM: Continued

- **Merits of HMM model :**

- Easy to model
- Fast
- Domain information not require

- **Demerits of HMM model :**

- Curse of dimentionality (number of parameters is too high if we go for higher *order* )
- We can not incorporate more features
- Dependent only on past context not on future context.

# Log Linear Model

- Discriminative model  $P(y|x)$
- Allow a very rich set of features to be used in a model
- 5 tuples  $(V, T, d, f : V \times T \rightarrow R^d, v \in R^d)$
- For any  $x \in V$  and  $y \in T$

$$P(y|x, v) = \frac{\exp(v \cdot f(x, y))}{\sum_{y' \in T} \exp(v \cdot f(x, y'))} \quad (1)$$

- **Parameter Estimation:** Training set  $(x_{(i)}, y_{(i)})$ , for  $i = 1 \dots m$

$$L(v) = \max \sum_{i=1}^m \log(P(y_{(i)}|x_{(i)}, v)) \quad (2)$$

# Log linear model: Continued

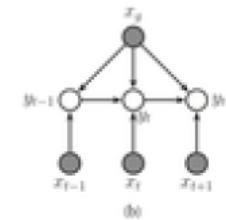
## Features

- For any pair  $(x,y)$ ,  $f(x,y) \in R^d$  is a feature vector
- Each component  $f_k(x,y)$  for  $k=1\dots d$  in this vector is referred to as feature.
- Features are indicator function, which allow us to represent different properties of  $x$  in conjunction with label  $y$
- $f_1(x,y) = 1$  if  $y = t_1$ , else 0
- $f_2(x,y) = 1$  if  $y = t_2$ , else 0
- $f_{20}(x,y) = 1$  if  $y = t_1$  and  $x = w_1$ , else = 0
- $f_{21}(x,y) = 1$  if  $y = t_1$  and  $x = w_2$ , else = 0
- $f_{30}(x,y) = 1$  if  $y = t_1$  and  $pos(x) = p_1$ , else = 0
- $f_{40}(x,y) = 1$  if  $y = t_1$  and  $suff_3(x) = s'_1$ , else = 0
- $f_{50}(x,y) = 1$  if  $y = t_1$  and  $pre_3(x) = p'_1$ , else = 0

# Maximum Entropy Markov Model (MEMM)

- MEMMs use *Log linear model* and *Markov assumption*
- It is a discriminative model  $p(x_1 \dots x_n | y_1 \dots y_n)$
- Tuples  $(V, T, H, f : H \times T \rightarrow R^d, v \in R^d)$
- **History (H)** : For any pair of sequence  $x_1 \dots x_n$  and  $y_1 \dots y_n$  define  $h_i = \langle y_{i-1}, x_1 \dots x_n, i \rangle$
- Given these components we define conditional tagging model

$$P(y_1 \dots y_n | x_1 \dots x_n) = \prod_{i=1}^n P(y_i | h_i, v) \quad (3)$$



$$P(y_i | h_i, v) = \frac{\exp(v \cdot f(h_i, y_i))}{\sum_{y' \in T} \exp(v \cdot f(h_i, y'))} \quad (4)$$

Figure :  
MEMMs

# MEMM: Continued

- **Parameter Estimation :**

Training set  $(x^{(k)}, y^{(k)})$  for  $k=1\dots m$   
where each  $x^{(k)} = x_1^{(k)} \dots x_{n_k}^{(k)}$  and  $y^{(k)} = y_1^{(k)} \dots y_{n_k}^{(k)}$   
and  $n_k$  is the length of  $k^{th}$  sentence

$$L(v) = \sum_{k=1}^m \sum_{i=1}^{n_k} \log(P(y_i^{(k)} | h_i^{(k)}, v)) \quad (5)$$

where

$$P(y_i^{(k)} | h_i^{(k)}, v) = \frac{\exp(v \cdot f(h_i^{(k)}, y_i))}{\sum_{y \in T} \exp(v \cdot f(h_i^{(k)}, y))} \quad (6)$$

- **Decoding of MEMMs** Finding most likely tag sequence for an input sequence  $x_1 \dots x_n$  is

$$\operatorname{argmax}_{y_1 \dots y_n \in Y(n)} P(y_1 \dots y_n | x_1 \dots x_n) \quad (7)$$

where  $Y(n)$  is all possible tag sequence of length  $n$



# MEMM: Continued

## Features :

- Recall that feature vector is a function  $f(h, y) \in R^d$  where  $h = \langle y_{-2}, y_{-1}, x_1 \dots x_n, i \rangle$
- Each feature  $f_j(h, y)$  for  $j \in 1 \dots d$  can be sensitive to any information from history  $h$
- Features for POS tagging used by [Ratnaparkhi, 1996]
  - ① **Word/Tag features :**
  - ② **Prefix and Suffix features:**
  - ③ **Trigram, Bigram and Unigram tag features:**
  - ④ **Contextual features**
  - ⑤ **Spelling features**

# MEMM: Continued

- **Features** used by [Ratnaparkhi, 1996] for POS tagging:

Word:	the	stories	about	well-heeled	communities	and	developers
Tag:	DT	NNS	IN	JJ	NNS	CC	NNS
Position:	1	2	3	4	5	6	7

Table 2: Sample Data

$w_i = \text{about}$                       &  $t_i = \text{IN}$   
 $w_{i-1} = \text{stories}$                 &  $t_i = \text{IN}$   
 $w_{i-2} = \text{the}$                       &  $t_i = \text{IN}$   
 $w_{i+1} = \text{well-heeled}$            &  $t_i = \text{IN}$   
 $w_{i+2} = \text{communities}$         &  $t_i = \text{IN}$   
 $t_{i-1} = \text{NNS}$                       &  $t_i = \text{IN}$   
 $t_{i-2}t_{i-1} = \text{DT NNS}$            &  $t_i = \text{IN}$

Table 3: Features Generated From  $h_3$  (for tagging about) from Table 2

$w_{i-1} = \text{about}$                       &  $t_i = \text{JJ}$   
 $w_{i-2} = \text{stories}$                       &  $t_i = \text{JJ}$   
 $w_{i+1} = \text{communities}$            &  $t_i = \text{JJ}$   
 $w_{i+2} = \text{and}$                       &  $t_i = \text{JJ}$   
 $t_{i-1} = \text{IN}$                       &  $t_i = \text{JJ}$   
 $t_{i-2}t_{i-1} = \text{NNS IN}$               &  $t_i = \text{JJ}$   
 $\text{prefix}(w_i) = \text{w}$                       &  $t_i = \text{JJ}$   
 $\text{prefix}(w_i) = \text{we}$                       &  $t_i = \text{JJ}$   
 $\text{prefix}(w_i) = \text{wel}$                       &  $t_i = \text{JJ}$   
 $\text{prefix}(w_i) = \text{well}$                       &  $t_i = \text{JJ}$   
 $\text{suffix}(w_i) = \text{d}$                       &  $t_i = \text{JJ}$   
 $\text{suffix}(w_i) = \text{ed}$                       &  $t_i = \text{JJ}$   
 $\text{suffix}(w_i) = \text{led}$                       &  $t_i = \text{JJ}$   
 $\text{suffix}(w_i) = \text{eled}$                       &  $t_i = \text{JJ}$   
 $w_i \text{ contains hyphen}$               &  $t_i = \text{JJ}$

Figure : Features snapshot from [Ratnaparkhi, 1996]

# MEMM: Continued

- **Features** used by [Ratnaparkhi, 1996] for POS tagging:

Word:	the	stories	about	well-heeled	communities	and	developers
Tag:	DT	NNS	IN	JJ	NNS	CC	NNS
Position:	1	2	3	4	5	6	7

Table 2: Sample Data

$w_i = \text{about}$                       &  $t_i = \text{IN}$   
 $w_{i-1} = \text{stories}$                 &  $t_i = \text{IN}$   
 $w_{i-2} = \text{the}$                         &  $t_i = \text{IN}$   
 $w_{i+1} = \text{well-heeled}$            &  $t_i = \text{IN}$   
 $w_{i+2} = \text{communities}$         &  $t_i = \text{IN}$   
 $t_{i-1} = \text{NNS}$                        &  $t_i = \text{IN}$   
 $t_{i-2}t_{i-1} = \text{DT NNS}$            &  $t_i = \text{IN}$

Table 3: Features Generated From  $h_3$  (for tagging about) from Table 2

$w_{i-1} = \text{about}$                       &  $t_i = \text{JJ}$   
 $w_{i-2} = \text{stories}$                    &  $t_i = \text{JJ}$   
 $w_{i+1} = \text{communities}$            &  $t_i = \text{JJ}$   
 $w_{i+2} = \text{and}$                         &  $t_i = \text{JJ}$   
 $t_{i-1} = \text{IN}$                             &  $t_i = \text{JJ}$   
 $t_{i-2}t_{i-1} = \text{NNS IN}$               &  $t_i = \text{JJ}$   
 $\text{prefix}(w_i) = \text{w}$                     &  $t_i = \text{JJ}$   
 $\text{prefix}(w_i) = \text{we}$                   &  $t_i = \text{JJ}$   
 $\text{prefix}(w_i) = \text{wel}$                 &  $t_i = \text{JJ}$   
 $\text{prefix}(w_i) = \text{well}$                &  $t_i = \text{JJ}$   
 $\text{suffix}(w_i) = \text{d}$                     &  $t_i = \text{JJ}$   
 $\text{suffix}(w_i) = \text{ed}$                   &  $t_i = \text{JJ}$   
 $\text{suffix}(w_i) = \text{led}$                 &  $t_i = \text{JJ}$   
 $\text{suffix}(w_i) = \text{eled}$               &  $t_i = \text{JJ}$   
 $w_i \text{ contains hyphen}$               &  $t_i = \text{JJ}$

Figure : Features snapshot from [Ratnaparkhi, 1996]

# MEMM: Continued

- Merits
  - It allows very rich source of features to incorporate
  - Complexity
- Demerits
  - Linear classifier therefore dependent on features
  - Required domain knowledge to design features
  - Dimension of feature vector is high
  - Label biasing problem

# Conditional Random Field

- CRF is also a discriminative tagging model.
- $s$  is label sequence for sentence  $x$ .

$$p(y_1 \dots y_n | x_1 \dots x_n) = p(s | x) \quad (8)$$

- Define a giant feature vector

$$\Phi(x, s) \in \mathbb{R}^d \quad (9)$$

This map the entire input sequence paired with entire output sequence into a d-dimensional feature vector.

- Feature vector

$$\Phi(x, s) = \sum_{j=1}^n f(h_j, y_j) \quad (10)$$

- We then build a log linear model

$$p(s | x; w) = \frac{\exp(w \cdot \Phi(x, s))}{\sum_{s' \in S^n} \exp(w \cdot \Phi(x, s'))} \quad (11)$$

Normalization constant  $S^n$  is a all possible state sequence of length  $n$

# CRF: Continued

- *Decoding*: For a given input sequence  $x_1.....x_m$  we would find most likely label sequence

$$f(x_1.....x_m) = \arg \max_{s \in S^m} p(s|x; w) \quad (12)$$

- **Differences over MEMMs**

- ① Normalizing over whole sentence probability will resolve *label biasing problem*
- ② Undirected graphical model
- ③ We get output for whole sentence at once, therefore not useful for online learning

# NER IN TWEETS

---

A case study: Ritter et al. EMNLP 2011

# Why NER in Tweets

- Corpus already exceeding size of the Library of Congress
- More **up-to-date** and **inclusive** than news articles



# Off-the-shelf-tool performance

## Predicted

Yess! Yess! Its official **Nintendo** announced today that they Will release the **Nintendo 3DS** in **north America** march 27 for \$250

■ Org ■ Product ■ Location

## Actual

Yess! Yess! Its official **Nintendo** announced today that they Will release the **Nintendo 3DS** in **north America** march 27 for \$250

■ Org ■ Product ■ Location

# Specific Challenges

- Plethora of distinctive named entity types
- But Infrequent
- Other common challenges remains
  - Lack of context
  - Lexical variants etc.

# T-NER Pipeline

- Divided into two steps
  - Segmentation (T-SEG)
  - Classification (T-CLASS)

# Segmentation

- Sequence Labelling Task using IOB scheme
- CRF Classifier
- Features Used: Orthographic, Contextual, dictionary features
- Dictionary: Type-list from Freebase
- Additional Features: T-POS, T-Chunk, T-Cap (Capitalization is informative or not)

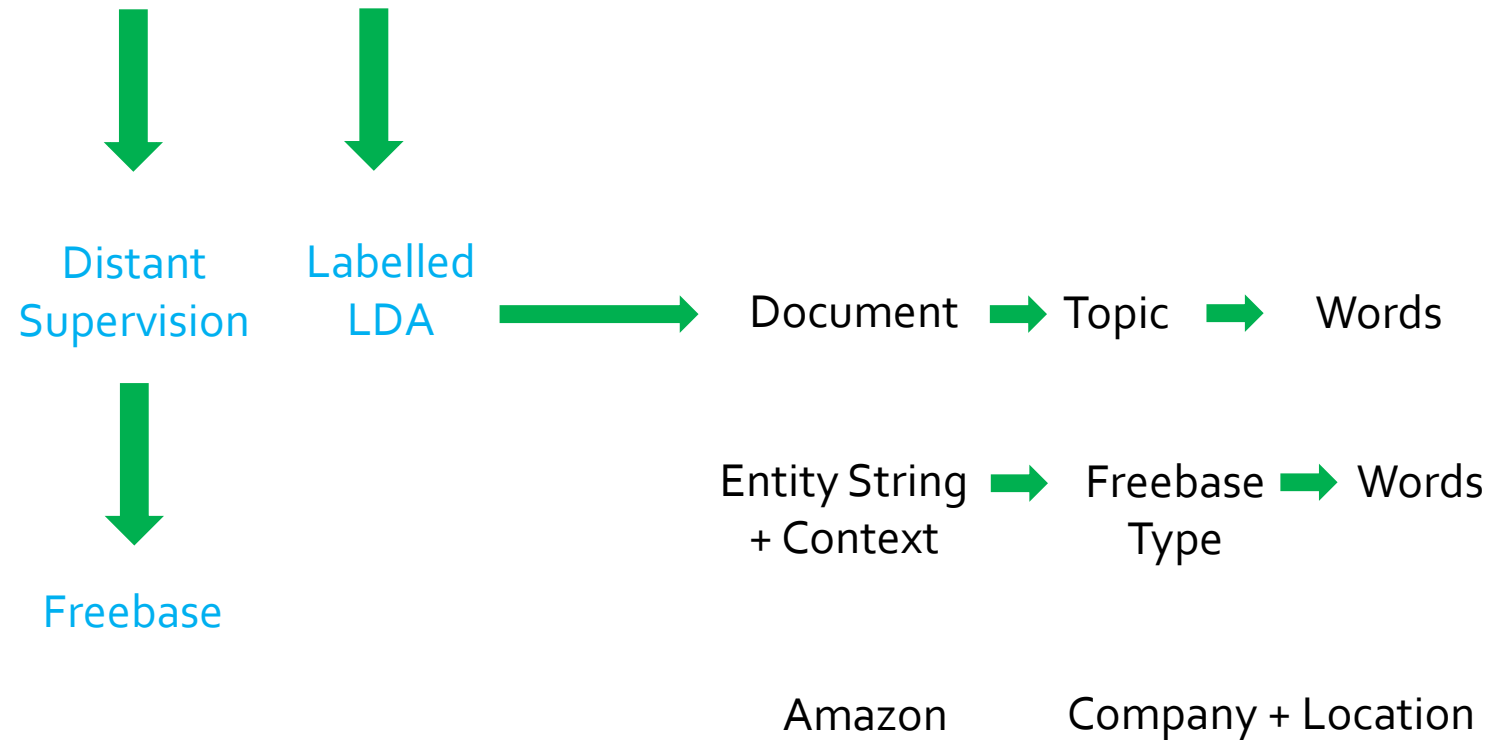
# Segmentation Performance

	P	R	F <sub>1</sub>	F <sub>1</sub> inc.
Stanford NER	0.62	0.35	0.44	-
T-SEG(None)	0.71	0.57	0.63	43%
T-SEG(T-POS)	0.70	0.60	0.65	48%
T-SEG(T-POS, T-CHUNK)	0.71	0.61	0.66	50%
T-SEG(All Features)	0.73	0.61	0.67	52%

Table 6: Performance at segmenting entities varying the features used. “None” removes POS, Chunk, and capitalization features. Overall we obtain a 52% improvement in F<sub>1</sub> score over the Stanford Named Entity Recognizer.

# Classification

Recall Specific Challenges: **Lots** of **Infrequent** entity types



# Classification Performance

Type	Top 20 Entities not found in Freebase dictionaries
<i>PRODUCT</i>	nintendo ds lite, apple ipod, generation black, ipod nano, apple iphone, gb black, xperia, ipods, verizon media, mac app store, kde, hd video, nokia n8, ipads, iphone/ipod, galaxy tab, samsung galaxy, playstation portable, nintendo ds, vpn
<i>TV-SHOW</i>	pretty little, american skins, nof, order svu, greys, kktny, rhobh, parks & recreation, parks & rec, dawson 's creek, big fat gypsy weddings, big fat gypsy wedding, winter wipeout, jersey shores, idiot abroad, royle, jerseyshore, mr . sunshine, hawaii five-0, new jersey shore
<i>FACILITY</i>	voodoo lounge, grand ballroom, crash mansion, sullivan hall, memorial union, rogers arena, rockwood music hall, amway center, el mocambo, madison square, bridgestone arena, cat club, le poisson rouge, bryant park, mandalay bay, broadway bar, ritz carlton, mgm grand, olympia theatre, consol energy center

Table 7: Example type lists produced by LabeledLDA. No entities which are shown were found in Freebase; these are typically either too new to have been added, or are misspelled/abbreviated (for example rhobh="Real Housewives of Beverly Hills"). In a few cases there are segmentation errors.

# Classification Performance

Type	LL	FB	CT	SP	N
<i>PERSON</i>	0.82	0.48	0.65	0.86	436
<i>GEO-LOC</i>	0.77	0.23	0.60	0.51	269
<i>COMPANY</i>	0.71	0.66	0.50	0.29	162
<i>FACILITY</i>	0.37	0.07	0.14	0.34	103
<i>PRODUCT</i>	0.53	0.34	0.40	0.07	91
<i>BAND</i>	0.44	0.40	0.42	0.01	54
<i>SPORTSTEAM</i>	0.53	0.11	0.27	0.06	51
<i>MOVIE</i>	0.54	0.65	0.54	0.05	34
<i>TV-SHOW</i>	0.59	0.31	0.43	0.01	31
<i>OTHER</i>	0.52	0.14	0.40	0.23	219
<b>overall</b>	0.66	0.38	0.53	0.45	1450

Table 10:  $F_1$  scores for classification broken down by type for LabeledLDA (LL), Freebase Baseline (FB), DL-Cotrain (CT) and Supervised Baseline (SP). N is the number of entities in the test set.



# Segmentation + Classification Performance

System	P	R	F <sub>1</sub>
COTRAIN-NER (10 types)	0.55	0.33	0.41
T-NER(10 types)	0.65	0.42	<b>0.51</b>
COTRAIN-NER (PLO)	0.57	0.42	0.49
T-NER(PLO)	0.73	0.49	<b>0.59</b>
Stanford NER (PLO)	0.30	0.27	0.29

Table 12: Performance at predicting both segmentation and classification. Systems labeled with PLO are evaluated on the 3 MUC types *PERSON*, *LOCATION*, *ORGANIZATION*.

# An Example of NLP in Computer Security

- Chambers, Fry, and McMasters. “Detecting Denial-of-Service Attacks from Social Media Text: Applying NLP to Computer Security”. NAACL 18
- Objective: Use social media as an indirect measure of real-time detection of attacks without network data

# References

- Ritter, Clark, Mausam, Etzioni. "Named Entity Recognition in Tweets: An experimental study." EMNLP 11
- Blei, Y Ng, Jordan. "Latent Dirichlet Allocation." J. Mach. Learn. Res.
- Ramage, Hall, Nallapati, Manning. "Labeled lda: a supervised topic model for credit attribution in multi-labelled corpora." EMNLP09

Thanks!  
Question and Comments!



[anand.ashish@iitg.ac.in](mailto:anand.ashish@iitg.ac.in)

<https://www.iitg.ac.in/anand.ashish>