CSC 594 Topics in AI – Applied Natural Language Processing

Fall 2009/2010

7. Named Entity Recognition

Named Entity Recognition

- **Named Entities** (NEs) are proper names in texts, i.e. the names of persons, organizations, locations, times and quantities
- NE Recognition (NER) is a sub-task of Information Extraction (IE)
- NER is to process a text and identify named entities
  - e.g. "U.N. official Ekeus heads for Baghdad."

  [ORG U.N.] official [PER Ekeus] heads for [LOC Baghdad] .
- NER is also an important task for texts in **specific domains** such as biomedical texts

Source: J. Choi, CSE842, MSU; Marti Hearst, i256, at UC Berkeley

Difficulties with NER

- Names are too numerous to include in dictionaries
- Variations
  - e.g. John Smith, Mr Smith, John
- Changing constantly
  - new names invent unknown words
- Ambiguities
  - Same name refers to different entities, e.g.
    - JFK – the former president
    - JFK – his son
    - JFK – airport in NY

Source: J. Choi, CSE842, MSU

Two Kinds of NER Approaches

Knowledge Engineering

- rule-based
- developed by experienced language engineers
- make use of human intuition
- requires only small amount of training data
- development could be very time consuming
- some changes may be hard to accommodate

Learning Systems

- use statistics or other machine learning
- developers do not need LE expertise
- requires large amounts of annotated training data
- some changes may require re-annotation of the entire training corpus
- annotators are cheap (but you get what you pay for!)

Landscape of IE/NER Techniques

Lexicons

Abraham Lincoln was born in Kentucky.

Classify Pre-segmented Candidates

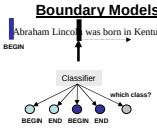
Abraham Lincoln was born in Kentucky.

Sliding Window

Abraham Lincoln was born in Kentucky.

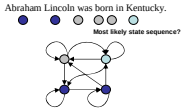
Boundary Models

Abraham Lincoln was born in Kentucky.



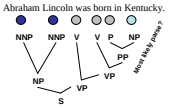
Finite State Machines

Abraham Lincoln was born in Kentucky.



Context Free Grammars

Abraham Lincoln was born in Kentucky.



Any of these models can be used to capture words, formatting or both.

CoNLL-2003

- CoNLL – Conference on Natural Language Learning by the ACL's Special Interest Group on Natural Language Learning
- Shared Task: Language-Independent Named Entity Recognition
- Goal: Identify boundaries and types of named entities
  - 4 entity types: Person, Organization, Location, Misc.

[ORG U.N.] official [PER Ekeus] heads for [LOC Baghdad] .

- Data:
  - Use the **IOB notation**
    - B- -- beginning of a chunk/NE
    - I- -- internal of a chunk/NE
    - O – outside of any chunk/NE
  - 4 pieces of info for each term

Word	POS	Chunk	EntityType
U.N.	NNP	I-NP	I-ORG
official	NN	I-NP	O
Ekeus	NNP	I-NP	I-PER
heads	VBZ	I-VP	O
for	IN	I-PP	O
Baghdad	NNP	I-NP	I-LOC
.	.	O	O

Details on Training/Test Sets

English data	Articles	Sentences	Tokens
Training set	946	14,987	203,621
Development set	216	3,466	51,362
Test set	231	3,684	46,435

German data	Articles	Sentences	Tokens
Training set	553	12,705	206,931
Development set	201	3,068	51,444
Test set	155	3,160	51,943

Table 1: Number of articles, sentences and tokens in each data file.

English data	LOC	MISC	ORG	PER
Training set	7140	3438	6321	6600
Development set	1837	922	1341	1842
Test set	1668	702	1661	1617

German data	LOC	MISC	ORG	PER
Training set	4363	2288	2427	2773
Development set	1181	1010	1241	1401
Test set	1035	670	773	1195

Table 2: Number of named entities per data file

Reuters Newswire + European Corpus Initiative

Evaluation of Single Entity Extraction

TRUTH:

Michael Kearns and Sebastian Seung will start Monday's tutorial, followed by Richard M. Karpe and Martin Cooke.

PRED:

Michael Kearns and Sebastian Seung will start Monday's tutorial, followed by Richard M. Karpe and Martin Cooke.

$$\text{Precision} = \frac{\text{\# correctly predicted segments}}{\text{\# predicted segments}} = \frac{2}{6}$$

$$\text{Recall} = \frac{\text{\# correctly predicted segments}}{\text{\# true segments}} = \frac{2}{4}$$

$$\text{F1} = \text{Harmonic mean of Precision \& Recall} = \frac{1}{((1/P) + (1/R)) / 2}$$

Summary of Results

- 16 systems participated
- Machine Learning Techniques
  - Combinations of Maximum Entropy Models (5) + Hidden Markov Models (4) + Winnow/Perceptron (4)
  - Others used once were Support Vector Machines, **Conditional Random Fields (CRF)**, Transformation-Based learning, AdaBoost, and memory-based learning
  - Combining techniques often worked well
- Features
  - Choice of features is at least as important as ML method
  - Top-scoring systems used many types
  - No one feature stands out as essential (other than words)

Precision, Recall, and F-Scores

English test	Precision	Recall	$F_{\beta=1}$
* Florian	88.99%	88.54%	88.76±0.7
* Chieu	88.12%	88.51%	88.31±0.7
Klein	85.93%	86.21%	86.07±0.8
Zhang	86.13%	84.88%	85.50±0.9
Carreras (b)	84.05%	85.96%	85.00±0.8
Curran	84.29%	85.50%	84.89±0.9
Mayfield	84.45%	84.90%	84.67±1.0
Carreras (a)	85.81%	82.84%	84.30±0.9
McCallum	84.52%	83.55%	84.04±0.9
Bender	84.68%	83.18%	83.92±1.0
Munro	80.87%	84.21%	82.50±1.0
Wu	82.02%	81.39%	81.70±0.9
Whitelaw	81.60%	78.05%	79.78±1.0
Hendrickx	76.33%	80.17%	78.20±1.0
De Meulder	75.84%	78.13%	76.97±1.2
Hammerton	69.09%	53.26%	60.15±1.3
Baseline	71.91%	50.90%	59.61±1.2

German test	Precision	Recall	$F_{\beta=1}$
* Florian	83.87%	63.71%	72.41±1.3
* Klein	80.38%	65.04%	71.90±1.2
Zhang	82.00%	63.03%	71.27±1.5
Mayfield	75.97%	64.82%	69.96±1.4
Carreras (b)	75.47%	63.82%	69.15±1.3
Bender	74.82%	63.82%	68.88±1.3
Curran	75.61%	62.46%	68.41±1.4
McCallum	75.97%	61.72%	68.11±1.4
Munro	69.37%	66.21%	67.75±1.4
Carreras (a)	77.83%	58.02%	66.48±1.5
Wu	75.20%	59.35%	66.34±1.3
Chieu	76.83%	57.34%	65.67±1.4
Hendrickx	71.15%	56.55%	63.02±1.4
De Meulder	63.93%	51.86%	57.27±1.6
Whitelaw	71.05%	44.11%	54.43±1.4
Hammerton	63.49%	38.25%	47.74±1.5
Baseline	31.86%	28.89%	30.30±1.3

* Not significantly different

Source: Marti Hearst, i256, at UC Berkeley

10

State of the Art Performance

- Named entity recognition
  - Person, Location, Organization, …
  - F1 in high 80's or low- to mid-90's
- However, performance depends on the entity types

  [Wikipedia] At least two hierarchies of named entity types have been proposed in the literature. BBN categories [1], proposed in 2002, is used for Question Answering and consists of 29 types and 64 subtypes. Sekine's extended hierarchy [2], proposed in 2002, is made of 200 subtypes.
- Also, various domains use different entity types (e.g. concepts in biomedical texts)

11

Sequence Labeling

- Inputs: x = (x_1, …, x_n)
- Labels: y = (y_1, …, y_n)
- Typical goal: Given x, predict y
- Example sequence labeling tasks
  - Part-of-speech tagging
  - Named Entity Recognition (NER)
    - Target class/label is the entity type, in IOB notation

Word	POS	Chunk	EntityType
U.N.	NNP	I-NP	I-ORG
official	NN	I-NP	O
Ekeus	NNP	I-NP	I-PER
heads	VBZ	I-VP	O
for	IN	I-PP	O
Baghdad	NNP	I-NP	I-LOC
.	.	O	O

Source: Andrew McCallum, UMass Amherst

12

Methods for Sequence Labeling

- Typically the following methods are used for NER:
  - **a) Hidden Markov Model** (HMM)
  - **b) Maximum Entropy Classifier** (MaxEnt)
  - c) Maximum Entropy Markov Model (MEMM)
  - **d) Conditional Random Fields** (CRF)
- These are all classifiers (i.e., supervised learning) which model sequences (rather than individual random variables)

Instance/word Features for NER

- Characteristics of the token & text in a surrounding window.

Feature	Explanation
Lexical items	The token to be labeled
Stemmed lexical items	Stemmed version of the target token
Shape	The orthographic pattern of the target word
Character affixes	Character-level affixes of the target and surrounding words
Part of speech	Part of speech of the word
Syntactic chunk labels	Base-phrase chunk label
Gazetteer or name list	Presence of the word in one or more named entity lists
Predictive token(s)	Presence of predictive words in surrounding text
Bag of words/Bag of N-grams	Words and/or *N*-grams occurring in the surrounding context

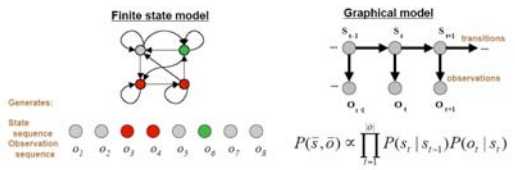
- Shape/orthographic features

Shape	Example
Lower	cummings
Capitalized	Washington
All caps	IRA
Mixed case	eBay
Capitalized character with period	H.
Ends in digit	A9
Contains hyphen	H-P

Source: Jurafsky & Martin "Speech and Language Processing"

a) Hidden Markov Models (HMMs)

HMMs are the standard sequence modeling tool in genomics, music, speech, NLP, …



$$P(\bar{s},\bar{o}) \propto \prod_{t=1}^{|\bar{o}|} P(s_t \mid s_{t-1}) P(o_t \mid s_t)$$

Parameters: for all states $S=\{s_1, s_2, \ldots\}$
- Start state probabilities: $P(s_1)$
- Transition probabilities: $P(s_t|s_{t-1})$
- Observation (emission) probabilities: $P(o_t|s_t)$ Usually a multinomial over atomic, fixed alphabet

Training:
- Maximize probability of training observations (w/ prior)

Source: Andrew McCallum, UMass Amherst

NER with HMMs

Given a sequence of observations:

Yesterday Pedro Domingos spoke this example sentence.

and a trained HMM:

Find the most likely state sequence: (Viterbi)

Any words said to be generated by the designated "person name" state extract as a person name:

Person name: Pedro Domingos

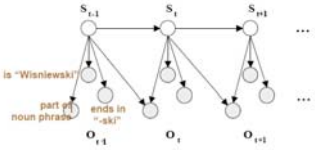
Source: Andrew McCallum, UMass Amherst

16

We want More than an Atomic View of Words

- Would like richer representation of text: many arbitrary, overlapping features of the words.

identity of word
ends in "-ski"
is capitalized
is part of a noun phrase
is in a list of city names
is under node X in WordNet
is in bold font
is indented
is in hyperlink anchor
last person name was female
next two words are "and Associates"



Source: Andrew McCallum, UMass Amherst

17

However…

- These arbitrary features are not independent.
  - Multiple levels of granularity (chars, words, phrases)
  - Multiple dependent modalities (words, formatting, layout)
  - Past & future
- Possible solutions:
  - ✖ Model dependencies
  - ✖ Ignore dependencies
  - ✓ Conditional Sequence Models

Source: Andrew McCallum, UMass Amherst

18

b) Maximum Entropy Classifier

- Conditional model p(y|x).
  - Do not waste effort modeling p(x), since x is given at test time anyway.
  - Allows more complicated input features, since we do not need to model dependencies between them.
- Feature functions f(x,y):
  - f1(x,y) = { word is Boston & y=Location }
  - f2(x,y) = { first letter capitalized & y=Name }
  - f3(x,y) = { x is an HTML link & y=Location}

Source: Andrew McCallum, UMass Amherst

Maximum Entropy Models

- Same as **multinomial logistic regression** (thus an exponential model for n-way classification)
- Features are constraints on the model.
- Of all possible models, we choose that which maximizes the entropy of all models that satisfy these constraints.
- Choosing one with less entropy means we are adding information that is not justified by the empirical evidence.

Source: Jason Balbridge, U of Texas at Austin

MaxEnt: A Simple Example

- Say we have an unknown probability distribution $p(x,y)$, where $x \in \{a,b\}$ and $y \in \{0,1\}$.
- We only know that $p(a,0)+p(b,0)=.6$.
- Well, actually we also know that $\Sigma_{x,y}\, p(x,y)=1$ since p is a probability distribution.

$p(x,y)$	0	1	
a	?	?	
b	?	?	
total	.6		1.0

- There are infinitely many consistent distributions conforming to the constraint.

Source: Jason Balbridge, U of Texas at Austin

MaxEnt: A Simple Example (cont.)

- Principle of Maximum Entropy recommends non-committal assignment of probabilities, subject to the constraints.

A consistent assignment

$p(x,y)$	0	1	
a	.5	.1	
b	.1	.3	
total	.6		1.0

Maximum entropy assignment

$p(x,y)$	0	1	
a	.3	.2	
b	.3	.2	
total	.6		1.0

- What if we also knew that $p(a,0)+p(a,1)=.6$?

Source: Jason Balbridge, U of Texas at Austin 22

MaxEnt: A Simple Example (cont.)

- An observation such as $p(a,0)+p(b,0)=.6$ is implemented as a constraint on the model p's expectation of a feature f :

$$E_p f = .6$$

- In general,

$$E_p f = \sum_{x \in \{a,b\}, y \in \{0,1\}} p(x,y) f(x,y)$$

where f(x,y) is an indicator function for the feature f (and there is one function for each feature; below is one example):

$$f(x,y) = \begin{cases} 1 & \text{if } b = 0 \text{ } y = 0 \\ 0 & \text{otherwise} \end{cases}$$

Source: Jason Balbridge, U of Texas at Austin 23

Maximizing Entropy

- To find the maximum entropy model, our objective is to maximize the log likelihood:

$$H(p) = - \sum_{x \in \{a,b\}, y \in \{0,1\}} p(x,y) \log p(x,y)$$

subject to the known constraints.

Note: - p(x,y) log(p(x,y)) is the entropy of p(x,y), which is the given model p.

In summary, *"the intuition of maximum entropy is to build a distribution by continuously adding features. Each feature is an indicator function, which picks out a subset of the training observations. For each feature, we add a constraint on our total distribution, specifying that our distribution for this subset should matcc with the empirical distribution that we saw in our training data. We then choose the maximum entropy distribution that otherwise accords with these constraints."*

Source: Jason Balbridge, U of Texas at Austin; Jurafsky & Martin "Speech and Language Processing" 24

MaxEnt Classification

- To identify the most probable label for a particular context *x*, we calculate:

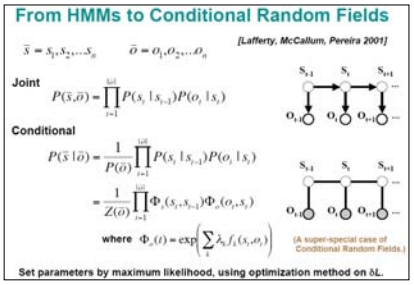
$$\begin{aligned}\hat{y} &= \text{argmax}_y\ p(y|x)\\ &= \text{argmax}_y\ log\ p(y|x)\\ &= \text{argmax}_y\ log(\frac{1}{Z(x)} exp\left(\sum_{i=1}^{n}\lambda_i f_i(x,y)\right))\\ &= \text{argmax}_y\ log(\frac{1}{Z(x)}) + log(exp\left(\sum_{i=1}^{n}\lambda_i f_i(x,y)\right))\\ &= \text{argmax}_y\ \sum_{i=1}^{n}\lambda_i f_i(x,y)\end{aligned}$$

Source: Jason Balbridge, U of Texas at Austin

25

d) Conditional Random Fields (CRF)

- Conditionally-trained, undirected graphical model.

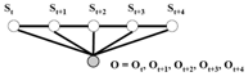


Source: Andrew McCallum, UMass Amherst

26

Characteristics of CRFs

- In CRF, the graph structure is often a linear chain, where the state sequence (S) connects S_{t-1} and S_t, and each state S_i is dependent on the observation sequence (O).



Markov on *s*, conditional dependency on *o*.

$$P(\vec{s}\,|\vec{o}) \propto \frac{1}{Z_{\vec{o}}}\prod_{t=1}^{|\vec{s}|}\exp\left(\sum_j \lambda_j f_j(s_t, s_{t-1}, \vec{o}, t)\right)$$

- CRF is a more general and expressive modeling technique than HMM, since it can model the dependency of each state on maximally the whole/entire observation sequence (not just its corresponding output unit).

Source: Andrew McCallum, UMass Amherst

27

CRFs

- CRFs are widely used and applied in NLP research. CRFs give state-the-art results in many domains and tasks.
  - Part-of-speech tagging
  - Named Entity Recognition
  - Gene prediction
  - Chinese word segmentation
  - Extracting information from research papers.
  - etc.

Source: Andrew McCallum, UMass Amherst 28

- For a standard linear-chain structure:

$$P(\mathbf{y} \mid \mathbf{x}) = \prod_t \Psi_k(y_t, y_{t-1}, \mathbf{x})$$

$$\Psi_k(y_t, y_{t-1}, \mathbf{x}) = \exp\left(\sum_k \lambda_k f(y_t, y_{t-1}, \mathbf{x})\right)$$

Source: Andrew McCallum, UMass Amherst 29