Named Entity Extraction
-
Maximum Entropy Modeling & Bootstrapping NE-Lists

PD Dr. Günter Neumann
DFKI and Saarland University

Example NE Approach - MENE [Borthwick et al 98]

- Combining rule-based and ML NE to achieve better performance
- Tokens tagged as: XXX_start, XXX_continue, XXX_end, XXX_unique, other (non-NE), where XXX is an NE category
- Uses Maximum Entropy Modeling (MEM)
  - One only needs to find the best features for the problem
  - MEM estimation routine finds the best relative weights for the features

From Cunningham & Bontcheva, 2003

Core idea of Maximum Entropy Modeling

- Probability for a class Y (e.g., PERSON) and an object X (e.g., „Peter Müller“) depends solely on the *features* that are **active** for the pair (X,Y)
- Features are the means through which an experimenter feeds problem-specific information (e.g., Recognition of NE)
- The *importance* of each feature is determined automatically by running a parameter estimation algorithm over a pre-classified set of examples („training-set“)
- Advantage: experimenter need only tell the model *what* information to use, since the model will automatically determine *how* to use it.

Maximum Entropy Modeling

- Random process
  - produces an output value y, a member from a finite set Y
  - Might be influenced by some contextual information x, a member from a finite set X
- Construct a stochastic model that accurately describes the random process
  - Estimate the conditional probability P(Y|X)
- Training data: $(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)$

$$r(x, y) \equiv \frac{c(x, y)}{N}$$

Simple example

- Task: estimate a joint probability distribution p defined over $\{x,y\}\times\{0,1\}$
- Known facts (constraints) about p
  - $p(x,0)+p(y,0)=0.6$
  - $p(x,0)+p(y,0)+p(x,1)+p(y,1)=1$

P(a,b)	0	1	
X	?	?	
Y	?	?	
Total	.6		1

One way to satisfy constraints

P(a,b)	0	1	
X	.5	.1	
Y	.1	.3	
Total	.6		1

Is this also the most accurate one?

Simple Example

- Observed facts are constraints for the desired model p
- Observed fact p(x,0)+p(y,0)=0.6 is implemented as a constraint of feature f_1 of model p, $E_p f_1$, where

$$E_p f_1 = \sum_{a \in \{x,y\}, b \in \{0,1\}} p(a,b) f_1(a,b) \qquad f_1(a,b) = \begin{cases} 1 & if \;\; b = 0 \\ 0 & otherwise \end{cases}$$

Most uncertain way to satisfy constraints:

P(a,b)	0	1	
X	.3	.2	
Y	.3	.2	
Total	.6	.4	1

Histories, binary features & futures

- History b: information derivable from the corpus relative to a token:
  - text window around token w_i, e.g. $w_{i-2},\ldots,w_{i+2}$
  - word features of these tokens
  - POS, other complex features
- Features:
  - yes/no-questions on history used by models to determine probabilities of
- Futures: what we are predicting (e.g., POS, name classes)

Features represent evidence

- a = what we are predicting (e.g., tags)
- b = what we observe (e.g., words)
- A feature f has the form

  $f_{y,q}(a,b) = 1$ if $a=y$ & $q(b)$ = true

  0 otherwise
- E.g.,

  $f_{NNP,q1}(a,b) = 1$ if $a=NNP$ & $q1(b)$ = true

  $f_{VBG,q2}(a,b) = 1$ if $a=VBG$ & $q2(b)$ = true

Weight features with conditional probability model

$$P(a \mid b) = \frac{\prod_{j=1}^{k} \alpha_j^{f_j(a,b)}}{Z(b)} = \frac{\prod_{j=1}^{k} \alpha_j^{f_j(a,b)}}{\sum_a \prod_{j=1}^{k} \alpha_j^{f_j(a,b)}}$$

- Z(b) = normalization factor
- $\alpha_j > 0$: weights for feature f_j
- P(a|b): (normalized) product of weights of active feature on the (a,b) pair, i.e., those features f_j such that $f_j(a,b)=1$

MENE (2)

- Features
  - Binary features – "token begins with capitalised letter", "token is a four-digit number"
  - Lexical features – dependencies on the surrounding tokens (window ±2) e.g., "Mr" for people, "to" for locations
  - Dictionary features – equivalent to gazetteers (first names, company names, dates, abbreviations)
  - External systems – whether the current token is recognised as an NE by a rule-based system

From Cunningham & Bontcheva, 2003

MENE (3)

- MUC-7 formal run corpus
  - MENE – 84.2% f-measure
  - Rule-based systems it uses – 86% - 91 %
  - MENE + rule-based systems – 92%
- Learning curve
  - 20 docs – 80.97%
  - 40 docs – 84.14%
  - 100 docs – 89.17%
  - 425 docs – 92.94%

From Cunningham & Bontcheva, 2003

Information Extraction
-
Bootstrapping NE lists

PD Dr. Günter Neumann
DFKI and Saarland University

Details of Bootstrapping approaches

- Bootstrapping classical NE types
  - Michael Collins and Yoran Singer, 1999

- Bootstrapping generalized names
  - Yangarber, Lin, Grishman, 2002
  - Lin, Yangarber, Grishman, 2003

- Context Pattern Induction method
  - Talukdar, Brants, Liberman, Pereira, 2006

Bootstrapping NE: idea

- Define manually only a small set of trusted seeds
- Training then only uses un-labeled data
- Initialize system by labeling the corpus with the seeds
- Extract and generalize patterns from the context of the seeds
- Use the patterns to further label the corpus and to extend the seed set (bootstrapping)
- Repeat the process until no new terms can be identified

Bootstrapping NE-learning: idea