NER FOR NELL

EXPLOITING MORPHOLOGICAL PATTERNS IN CATEGORIES

Reza Bosagh Zadeh

October 29, 2009

OVERVIEW

- Task Description
- How to solve outside a NELL system
- Simple approach evaluated
- How to tackle in a NELL system: initial experiments

WHAT IS "NAMED ENTITY RECOGNITION"?

October 14, 2002, 4:00 a.m. PT

For years, Microsoft Corporation CEO Bill Gates railed against the economic philosophy of open-source software with Orwellian fervor, denouncing its communal licensing as a "cancer" that stifled technological innovation.

Today, Microsoft claims to "love" the open-source concept, by which software code is made public to encourage improvement and development by outside programmers. Gates himself says Microsoft will gladly disclose its crown jewels--the coveted code behind the Windows operating system--to select customers.

"We can be open source. We love the concept of shared source," said Bill Veghte, a Microsoft VP. "That's a super-important shift for us in terms of code access."

Richard Stallman, founder of the Free Software Foundation, countered saying…

Extract named-entities from text, label as "Person", "Organization", etc

Microsoft Corporation
CEO
Bill Gates
Microsoft
Gates
Microsoft
Bill Veghte
Microsoft
VP
Richard Stallman
founder
Free Software Foundation

Slide: William Cohen, Information Extraction

WHAT PATTERNS?

- Yarow-sky
- Min-ski
- Bosagh-Zadeh
- Milose-vitch

Current RTW system helps us find popular names using context frames.

Should be able to find patterns in popular names and use them to discover rarely used names.

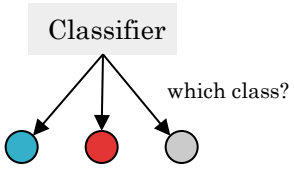
MODELS FOR NER

Lexicons (Gazetteers)

Abraham Lincoln was born in Kentucky.

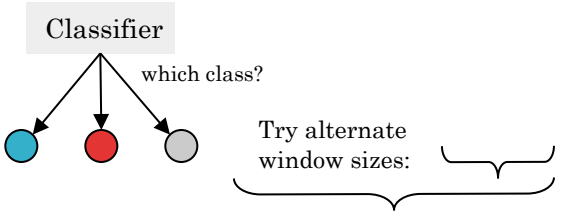
Classify Pre-segmented Candidates

Abraham Lincoln was born in Kentucky.

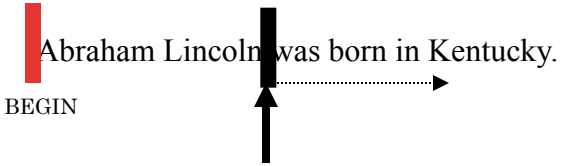


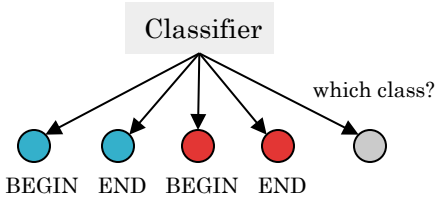
Sliding Window

Abraham Lincoln was born in Kentucky.



Boundary Models

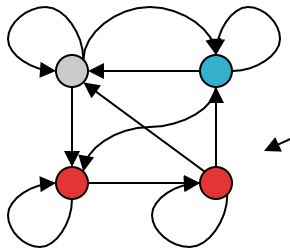




Token Tagging

Abraham Lincoln was born in Kentucky.

Most likely state sequence?



This is often treated as a structured prediction problem…classifying tokens *sequentially*

HMMs, CRFs, ….

Slide: William Cohen, Information Extraction

PAPER: MIKHEEV ET. AL.

- How well can we perform with only a lexicon (list/ gazeteer)?

- With lists:

category	learned lists		common lists		combined lists	
	recall	precision	recall	precision	recall	precision
organization	49	75	3	51	50	72
person	26	92	31	81	47	85
location	76	93	74	94	86	90

Recall: Number of correct tags in the answer file over total number of tags in the key file.

Precision: Number of correct tags in the answer file over total number of tags in the answer file.

NER FOR NELL

- Don't have easy access to supervised data: doesn't fit the never-ending-learner model
- Context isn't important anymore!
- Want to use Morphological patterns abundant in human names and surnames
- Need to be fast each iteration
- Initial experiment: focus on suffixes

COMMON SUFFIXES - TRIGRAMS

- Most common trigram endings of NPs in the list of person names currently obtained from RTW:

-SON, -MAN, -TON, -ELL, -LER, -LEY, -ING, -TER,-ERS, -ARD

- Not very useful: would have us believe “Rowing” is a person name.

COMMON SUFFIXES - NGRAMS

- Most common fourgram endings of NPs in the list of person names currently obtained from RTW:

-NSON, -LSON, -LLER, -NTON, -BUSH, -RSON, -ROWN, -MITH, -TEIN, -RMAN

- Not very useful: would have us believe "Protein" is a person name.

- Same problem for ngrams of length 3 to 6

-NSON, -LSON, -LLER, -NTON, -OHNSON, -HNSON, -BUSH, -LINTON, -INTON, -RSON

PROBLEM: HOW TO FIND DISCRIMINATIVE NGRAMS?

- Not only identify the most common suffixes in the list of names, but those name suffixes which also appear *rarely in all NPs*.

- Two competing requirements

- Borrow ideas from TF-IDF and define score for ngram i:

$$\text{score}(i) = \frac{a_i}{b_i}$$

a_i: frequency of ngram i in names list
b_i: frequency of ngram i in entire NP list

MUCH NICER

- Take all ngrams and sort by score function
- Use top 100-scoring ngrams
- Length freely varying from 3 to 5
- Picks up…

MUCH NICER

New names, *not picked up before*

TAKEI, RICHARD_M._NIXON, BISMARK, BASSANO, PARRISH, BUSBY, CANOGA, **lubavitch**, LUSTIG, MOHR, ROBB, JAMES_BALDWIN, ROHRER, BIZNIK, FINNERAN, MOREA, KATZ, WAHID, SOLOW, SELES, POLOS, SCHAPIRO, CHAUNCEY, KAHN, OLIVIER, DEVANEY, LEGUIZAMO, MUSEE, BALDWIN, SHULA, MORAVEC, SPADER, ZHANG, MUSIAL, YODER, CUSACK, SMYTH, SMOLIN, WANNSTEDT, STAGG, MOHER, PITTS, NIVEN, LOWRY, METZLER, WHYTE, ANJOU, FUKUDOME, VOGEL, CULKIN, EMMITT, ZOOK, CURRIE, STAAL, PEDROIA, MCCORMICK

List not filtered or altered in any way: all seem to be names

Some very familiar-but-rare suffixes, such as **-vitch**

NEXT STEPS

- Use prefixes as well as suffixes:

  McDowell
  McCartney
  O'Connor
  O'Dowel

- Try other categories

  Aghani-**stan**
  Paki-**stan**

  Can potentially work for locations:

  Fin-**land**
  Green-**land**
  Eng-**land**

NEXT STEPS

- Put this into main pipeline for RTW
- Insert new names during bootstrapping process
  - Should be interesting to see the interaction between morphologically identified names and names found using contexts
- Use confidence scores

Thanks!