Introduction to NER project

Omid Moradiannasab
Michal Petko

Agenda

- NER task
  - Introduction
  - Applications
  - Ambiguity
  - Evaluation
- Solutions
  - Rule based approaches
  - Statistical approaches
- Available resources
  - Tools
  - Corpora
- Our goals in order

What is tagging?

- Tagging as one of the fundamental applications in NLP
  - POS tagging
  - Named-Entity tagging

- one of the very earliest problems considered in statistical or machine learning approaches to NLP
- Goes back to the late 1980s.

Part-of-Speech Tagging

INPUT:

Profits soared at Boeing Co., easily topping forecasts on Wall Street, as their CEO Alan Mulally announced first quarter results.

OUTPUT:

Profits/N soared/V at/P Boeing/N Co./N ,/, easily/ADV topping/V forecasts/N on/P Wall/N Street/N ,/, as/P their/POSS CEO/N Alan/N Mulally/N announced/V first/ADJ quarter/N results/N ./.

N = Noun

V = Verb

P = Preposition

Adv = Adverb

Adj = Adjective

…

Named Entity Recognition

INPUT:

Profits soared at Boeing Co., easily topping forecasts on Wall Street, as their CEO Alan Mulally announced first quarter results.

OUTPUT:

Profits soared at [Company Boeing Co.], easily topping forecasts on [Location Wall Street], as their CEO [Person Alan Mulally] announced first quarter results.

- locate and classify atomic elements in text consisting of blocks of one or more words into predefined categories
- persons, organizations, locations, date, time, monetary values, percentages, etc
- at first glance does not look like a tagging problem

Named Entity Recognition as Tagging

INPUT:

Profits soared at Boeing Co., easily topping forecasts on Wall Street, as their CEO Alan Mulally announced first quarter results.

OUTPUT:

- Profits/NA soared/NA at/NA Boeing/SC Co./CC ,/NA easily/NA topping/NA forecasts/NA on/NA Wall/SL Street/CL ,/NA as/NA their/NA CEO/NA Alan/SP Mulally/CP announced/NA first/NA quarter/NA results/NA ./NA
- NA = No entity
- SC = Start Company
- CC = Continue Company
- SL = Start Location
- CL = Continue Location
- ….

NER Applications

- Is now available- and I think popular – in applications like Apple or Google mail, and web indexing

NER Applications

- Some other uses:
  - Sentiment can be attributed to companies or products
  - A lot of IE relations are associations between named entities
  - For question answering, answers are often named entities.

Evaluation

- Recall and precision are straightforward for tasks like IR and text categorization, where there is only one grain size (documents)
- The measures don't behave the same when there are **boundary errors** (which are common)
  - First Bank of Chicago announced earnings …
  - First Microsoft announced … then Apple …
  - This counts as 2 errors: both a fp and a fn
  - Selecting NOTHING would have been better
  - There are some other metrics (e.g. MUC scorer) which give some partial credit in such situations

Ambiguity?

- Ambiguity as in many other problems in NLP
  - **marathon** is a village in Marathon County, Wisconsin, United States and a sporting event
  - “boston marathon” is a specific sporting event
- model of the words in and around an entity (Local & contextual)

Rule-based techniques

- Uses *gazetteers* (lists of words and phrases) that categorize names
  - E.g. cities, countries, …
  - Doesn't have enough contextual information to handle ambiguity
- Rules also used to verify or find new entity names
  - Local pattern:
    - 14th March 2011
    - 14/03/2011
  - Contextual patterns:
    - "<number> <word> street" for addresses
    - "<street address>, <city>" or "in <city>" to **verify** city names
    - "<street address>, <city>, <state>" to **find** new cities
    - "<title> <name>" to find new names
- better precision, but at the cost of lower recall and months of work

Statistical methods

- From the **training set** (manually annotated text), induce a function that maps new sentences (X) to their tag sequences (Y)
  - Trigram Hidden Markov Model
    - representing the dependencies of the variables x and y as a joint probability distribution P(X,Y)
    - defines distributions over the "next word" given a finite history.
    - A sequence of decisions given a brief history
    - The formula for a trigram HMM:

$$p(x_1 \cdots x_n, y_1 \cdots y_n) = \prod_{i=3}^{n} q(y_i|y_{i-2}, y_{i-1}) \cdot \prod_{i=1}^{n} e(x_i|y_i)$$

  - Global linear model
    - Conditional probability P(Y|X) instead of the joint probability
    - move away from history-based models No idea of attaching probabilities to "decisions"
    - model feature vectors over the whole sequence
    - Any features you want!
      - If current word is *base* and the tag is *VB*
      - If current word ends in *ing* and tag is *VBG*
      - if <t-2; t-1; t> = <DT, JJ, VB>
    - feature selection can be a difficult problem

$$F(x) = \arg\max_{y \in \mathbf{GEN}(x)} \mathbf{f}(x, y) \cdot \mathbf{v}$$

- requires a large amount of manually annotated training data

Statistical methods

- Accurate recognition requires millions of words as training data
  - may be more expensive than developing rules for some applications
- Both rule-based and statistical can achieve about 90% effectiveness for categories Such as names, locations, organizations
  - others, such as product name, can be much worse

Available Tools

- Stanford Named Entity Recognizer : “a Java implementation of a (arbitrary order) linear chain Conditional Random Field (CRF) sequence models.”
- NLTK ”provides a classifier that has already been trained to recognize named entities, accessed with the function nltk.ne_chunk()”
- GATE (University of Sheffield) “is an NLP toolkit written in Java. It includes ANNIE, a ready-to-run information extraction system made from statistical NLP components. ANNIE includes a sentence splitter, a tokenizer, a part-of-speech tagger, and a named entity recognizer.”

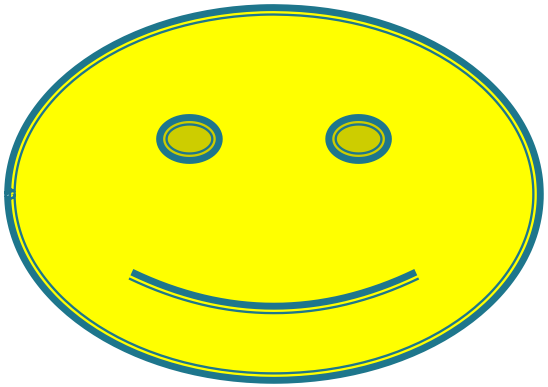
Corpora

- Well-known POS-tagged corpus for Persian
  - gathered form daily news and common texts
  - contains about 2.6 millions manually tagged words with a tag set that contains 40 Persian POS tags
  - Useful tags for our task:
    - N_SING_PR -> Bryan, Fox, News, …
    - N_SING_LOC -> England, Shop, …
    - N_SING_TIME -> year, today, night, earlier, …
    - …
- Persian Treebank
  - currently contains 1000 sentences
  - Useful tags for our task:
    - “pers” -> Adolf Born
    - “loc” -> New York City
    - “time” -> 1960
- Slovak corpus

Goals in order

1. Manually tagging a training, development, and test set for both languages (quite laborious task!)
2. Making gazetteers and regular expressions
3. Implementing one or more statistical approaches
4. Adapting and training one or more of the already available tools (optional)
5. Evaluating the taggers over both Persian and Slovak and doing a comparative study on the results of every approach
6. Extracting some contextual information like patterns of the context for every entity type (as a probable extension)

Any Question?



References

- “Tagging Problems, and Hidden Markov Models”, Michael Collins, Columbia University
- “Information Extraction and Named Entity Recognition”, Christopher Manning, Stanford University