UNSUPERVISED RELATION EXTRACTION FROM WEB

CS671: NATURAL LANGUAGE PROCESSING PROJECT

Vempati Anurag Sai
Bhavishya Mittal

PROBLEM STATEMENT

Extracting relation tuples from an unstructured corpus that is effective at noise removal.

Query :

Input: A partially filled tuple,

System will search for possible entries for the missing fields

Rank the resulting tuples based on a probabilistic measure.

Example: Query… *? lower mortality*

A: ('use of non-steroidal anti-inflammatory drugs -LRB- NSAIDs -RRB-', 'decrease', 'mortality')

[1] Dan Klein and Christopher D. Manning. 2003. Accurate Unlexicalized Parsing. Proceedings of the 41st Meeting of the Association for Computational Linguistics, pp. 423-430.

[2] Marie-Catherine de Marneffe, Bill MacCartney and Christopher D. Manning. 2006. Generating Typed Dependency Parses from Phrase Structure Parses. In LREC 2006.

PAST WORK V/S OUR APPROACH

Past work:

- Previously decided set of relations.
- Supervised vs unsupervised.
  - Supervised: Manual annotations(tiresome) /wikipedia infobox(domain specific)
- Heavy linguistic machinery. Don't scale properly to web data.

Our approach:

- Single-Pass Extractor
  - Makes a single pass over the entire corpus to extract tuples.
  - Labeled tuples as trustworthy using heuristics.
  - Used noun phrases to make the tuples more informative
- Redundancy-Based Assessor
  - Used synsets to group similar tuples and get a frequency count.
  - Assigned a probability to each retained tuple.

SELF-SUPERVISED LEARNER

- Automatically labeling its own training data as positive or negative.
- Using this labeled data to train a classifier to decide trustworthiness.
- Extractions take the following form:
  - tuple ‘t’ = (ei , ri,j , ej)

  where ei and ej are entities, and ri,j is a relationship between them.

- Heuristics used to identify any tuple as trustworthy or not are:
  - The length of the dependency chain between ei , ej and ri,j. (Stanford Parser)
  - Neither ei nor ej consist solely of a pronoun. (Pronoun added to relation)
- Tuples mapped to feature vector. Some features used:
  - Presence of particular POS in entities/relation
  - Length of entities/relations relative to the sentence length
  - Distance between entities and relations
  - The ordering of entities and relations

SINGLE-PASS EXTRACTOR

- Noun chunker to extract noun phrases
- Text in-between considered for relations
- POS tags from NLTK
- Trustworthiness: Features extracted and sent to the classifier
- Group tuples similar to each other (Synsets)

Examples tuples:

Input: The American Civil War, also known as the War between the States or simply the Civil War, was a civil war fought from 1861 to 1865 in the United States after several Southern slave states declared their secession and formed the Confederate States of America.

('American Civil War','known as','War between the states')

('several Southern slave states', 'declared', 'secession')

Input: Tendulkar won the 2010 Sir Garfield Sobers Trophy for cricketer of the year at the ICC awards.

('Tendulkar', 'won', 'the 2010 Sir Garfield Sobers Trophy')

QUERY MODULE AND RESULTS

- Given a partially filled tuple, finds all matching tuples from the database
- Similar meaning tuples are also returned
- Ranks the tuples based on “extent of match”

Results:

Query: *? Move protons* → ('ATP hydrolysis', 'move', 'protons')

? lower infiltration → (Fc for 6 days', 'decreased', 'MNC infiltration‘)

? Treat ? → (‘Hep G2 cells’,’treated’,’2D3’)

[1] Banko, Michele, et al. “Open Information Extraction from the Web.” IJCAI. Vol. 7. 2007.
[2] Fader, Anthony, Stephen Soderland, and Oren Etzioni. “Identifying relations for open information extraction.” Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2011.