Project Proposal

CSCI-GA.2590-001 Natural Language Processing, Spring 2012

Maor Leger, N12970347

ABSTRACT

A proposal for the final project requirement, providing a general description, preliminary list of references, a description of a baseline system, and one experiment performed so far

Table of Contents

General Description 3

Possible techniques 3

References (preliminary) 4

Baseline 5

One experiment performed so far 7

General Description

In my project, I intend to extend Homework #7 – Relation Extraction.

I intend to use the partitive task corpus from BlackBoard provided by Professor Meyers and the MaxEnt wrapper package written by the TA, Ang Sun.

Currently, I intend to keep the corpus unchanged, and avoid simplifying my task but this may change given the success rate of my experiments and time constraints. At any rate, I'll be sure to note any simplifications I've made along the way and modify the scorer program accordingly.

Possible techniques

For the purpose of this project, I intend to experiment with the following techniques:

1. Statistical approach
    a. Maximum Entropy model
    b. Hidden Markov Model
    c. Maximum Entropy Markov Model
2. Rule based system
    a. Regular expressions
    b. Some deterministic FSA
    c. Parsing techniques based on CFGs

Note that I may not actually use all of these techniques in my final project, but rather I'll try to combine elements from each technique in a way that increases my final measures for precision, recall, and f-score

References (preliminary)

1. ZHOU, SU, ZHANG, ZHANG – Exploring Various Knowledge in Relation Extraction, 2005. http://www.aclweb.org/anthology/P/P05/P05-1053.pdf
2. ACE English Relations Annotation Guidelines. http://projects.ldc.upenn.edu/ace/docs/English-Relations-Guidelines_v5.8.1.doc
3. Jurafsky, Martin – Speech and Language Processing, Second Edition. ISBN-10: 0131873210. http://www.cs.colorado.edu/~martin/slp.html
4. Sun, Ang – Extracting Arguments for %. http://cs.nyu.edu/courses/spring12/CSCI-GA.2590-001/NLP_HW7_specs.pdf

Note that I may not actually use all references, and in fact I may choose different references as the project work continues, but these references look complete in the sense that they contain one approach to a solution, one guideline document, and of course the textbook which highlights many approaches to this task.

Baseline

My baseline system uses strictly MaxEnt to predict the ARG0, ARG1, ARG2, and ARG3 of a sentence given the PRED. The baseline system is actually a slightly modified version of the system used for the % task in homework 7.

The approach:
Using the training file, train the MaxEnt model using the following features:

1. relationClass = the name of the relation class that the PRED and ARGs belong to.
2. candToken = The actual token examined
3. candTokenPOS = The POS tag of candToken
4. tokenBeforeCand = The token immediately preceding candToken
5. posBeforeCand = The POS tag of tokenBeforeCand
6. tokenAfterCand = The token immediately following candToken
7. posAfterCand = The POS tag of tokenAfterCand
8. tokensBetweenCandPred = The _ joined list of tokens between but not including candToken and predicateToken
9. numberOfTokensBetween = The number of tokens between but not including candToken and predicateToken
10. possBetweenCandPred = The _ joined list of POS tags of the tokensBetweenCandPred
11. existVerbBetweenCandPred = Whether a VB_ POS tag exists between candToken and predicate
12. BIOChunkChain = The _ joined list of the BIO tags of the tokens between and including candToken and predicate
13. ChunkChain = The _ joined list of different phrase chunks between and including candToken and predicate
14. candPredInSameNP = whether the candToken and predicate are in the same NP or not
15. candPredInSamePP = whether the candToken and predicate are in the same PP or not
16. candPredInSameVP = whether the candToken and predicate are in the same VP or not
17. existSupportBetweenCandPred = whether a SUPPORT relation tag is found between candToken and predicate

In the next step, I tag each sentence individually as follows:

1. Extract the features above for each token
2. Call MaxEntPredict.jar to get a probability distribution among the different ARGS = {ARG0, ARG1, ARG2, ARG3, None}
3. Simply assign each ARGi to the token with the highest probability for the corresponding ARGi.

This system thus makes many simplifying assumptions but the first one that needs to be addressed in my project is the assumption that each sentence that contains a PRED will contain exactly one of each ARGi. Of course, we know that the relation class of the sentence can significantly constrain the types of ARGs the PRED token can have, and thus my first experiment would be to limit the type of ARGs that can be assigned for tokens in a sentence.

Results from the baseline system are not impressive, but they at least show me the task CAN be done:

Experiment	**Precision**	**Recall**	**F-Score**
Baseline MaxEnt	0.22	0.46	0.3

One experiment performed so far

The first experiment involved adding selectional restrictions on the type of ARGs a sentence can have.

Basically, following the guidelines set from the final_project_possibilities.pdf file – slide 8.

According to these guidelines:

- ARG0 can only appear in the SHARE class
- ARG1 can appear in the following classes:
    - PARTITIVE-QUANT
    - GROUP
    - SHARE
    - PARTITIVE-PART
- ARG2 can appear in the following classes:
    - SHARE
    - GROUP
- ARG3
    - Ignored for now, but a future experiment may limit ARG3 or I may choose to simplify my task by scoring ARG1 and ARG3 as one (i.e. if the key states ARG3 and the system outputs ARG1 I may choose to simplify the task by scoring this pairing as correct)

Therefore, my experiment involved ignoring the probabilities of ARG0 and ARG1 if the relation class does not support it. For example, if the class is GROUP I will ignore any probabilities of ARG0 and not mark any ARG0 in this sentence.

Future improvement: I do not take this into account when training and when actually running MaxEntPredict on a list of tokens. What this means is that my probabilities really do not add up to 1 in all cases (see the example above, and this is due to the fact that the above rules are not "set in stone" – there are exceptions to these rules in the training file). A future experiment may create separate models for each relation class and run the appropriate model when tagging such that each model only contains the ARGs that may belong there (again, in the above example the model for GROUP will include the following outcomes: {None, ARG1, ARG2, ARG3} whereas the model for SHARE will include all ARGs).

Please see below for a comparison of the baseline system with this first experiment:

Experiment	Precision	Recall	F-Score
Baseline MaxEnt	0.22	0.46	0.3
Experiment 1 – selectional restrictions	0.38	0.59	0.46