Inducing Constraint-based Grammars using a Domain Ontology

Smaranda Muresan
Department of Computer Science
Columbia University
New York, New York, 10027
smara@cs.columbia.edu

In many knowledge intensive applications, there is a critical need to populate knowledge bases rapidly and to keep them up to date. Since the World Wide Web is a large source of information that continuously is being updated, a solution is to automatically acquire knowledge from text, which requires language understanding in a smaller or grater degree. The need for "rapid" text-to-knowledge acquisition imposes some critical conditions on the methods used: scalability and adaptability. Thus, there is a need to move from hand-crafted grammars and hand-build systems to learning methods. However, most statistical and learning techniques have been applied only to restricted domains (e.g., air travel domain) or tasks (e.g., information extraction where the knowledge is limited to *a priori* relations and entities), reducing the variety of the acquired knowledge.

This thesis presents a framework for domain specific text-to-knowledge acquisition, with focus on medical domain. The main challenge of this domain is the abundance of linguistic phenomena that require both syntactic and semantic information in order to "understand" the meaning of the text, and thus to acquire knowledge. Examples include prepositional phrases, coordinations, noun-noun compounds and nominalizations, phenomena which are not well covered by existing syntactic or semantic parsers.

In my thesis, I propose a relational learning framework for the induction of a constraint-based grammar able to capture both syntax and aspects of meaning in an interleaved manner from a small number of semantically annotated data. The novelty of this framework is the learning method based on an ordered set of examples. This approach to learning follows the argument that language acquisition is an incremental process, in which simpler rules are acquired prior to complex ones. Several new theoretical concepts need to be tied together in order to make the approach feasible and theoretically sound: 1) a type of constraint-based grammar, called *lexicalized well-founded grammar*, which is learnable and able to capture large fragments of natural language; 2) a semantic representation, which we call *semantic molecule* that can be linked to the grammar and is simple enough to allow the relational learning of the grammar; 3) a small ordered set of semantically annotated examples, called *representative examples*, which is used as our training data; and 4) an ontology-based semantic interpretation encoded as a constraint at the grammar rule level (Φ_{onto}), which refrains from full logical analysis of meaning, known to be intractable. On the application side, the grammar learning is used for rapid acquisition of medical terminological knowledge from text.

Semantic molecule. Given a natural language expression, w, we denote by $w\prime = h \bowtie b$ the semantic molecule of w, where h is the head acting as a valence for semantic composition, and b is the body acting as the semantic representation of w. The head is represented as a one level feature structure (i.e., feature values are atomic), while the body is a Canonical Logical Form given as a flat semantic representation, similar with Minimal Recursion Semantics (MRS). Unlike MRS, it uses as semantic primitives a set of frame-based atomic predicates of the form: *concept.attr=concept*, suitable for the interpretation on the ontology: *concept* corresponds to a frame in the ontology and *attr* is a slot of the frame, encoding either a property or a relation. For example, for the adjective "chronic" we have the following semantic molecule:

$$\begin{bmatrix} cat & a \\ head & X \\ mod & Y \end{bmatrix} \bowtie [X.isa = chronic, Y.Has\_prop = X]$$

The *cat* attribute (i.e., syntactic category), is mandatory for each molecule head, and is used as principle for grammar lexicalization. The composition of semantic molecules occurs at the grammar rule level via a constraint, Φ_{comp}.

Lexicalized Well-Founded Grammar and Representative Examples. A *lexicalized well-founded grammar (LWFG)* is a Context-Free Grammar enhanced with a set of partial ordering relations among the nonterminals and with a set of semantic molecules associated with the set of terminals. $LWFG$ has the following properties: 1) the set of nonterminals is well-founded; 2) every nonterminal symbol is a left-hand side in at least one ordered nonrecursive rule, i.e. it is greater than all nonterminals from the right-hand side; 3) the empty string cannot be derived from any nonterminal symbol; and 4) all substrings, w derived from a nonterminal have the same category (*cat*) of their semantic molecules, $w\prime$. Based on the first property the concept of representative examples can be defined, as given below, and the bottom up induction of the grammar is guaranteed. The second and the third properties ensure the termination condition for the induction process, while the last property guides the predicate invention during learning through the attribute *cat* of the semantic molecule's head.

For a *LWFG*, G, the *representative set*, E_R of the sublanguage $E \subseteq L(G)$ has been defined and computed, such that: 1) E_R is small (i.e., its size is smaller than the size of the set of grammar production rules); 2) E_R is ordered based on the partial ordering relation among the nonterminals; and 3) if G covers E_R, the grammar covers the sublanguage E. The *LWFG* Induction Theorem in (Muresan, Muresan, & Klavans 2004) showed that E_R can be used as a small semantic treebank for the bottom-up induction of a *LWFG* .

We encode the *lexicalized well-founded grammar* in the Definite Clause Grammar formalism, where each nonterminal is augmented with a semantic molecule, and each rule is augmented with two constraints: one for semantic composition, Φ_{comp}, and one for ontological validation, Φ_{onto}.

$$\begin{aligned} A(w,h \bowtie b) \Rightarrow\ & B_1(w_1,h_1 \bowtie b_1),...,B_n(w_n,h_n \bowtie b_n): \\ & w = w_1...w_n,\ b = b_1,...,b_n, \\ & \Phi_{comp}(h,h_1,...,h_n),\ \Phi_{onto}(b) \end{aligned}$$

$\Phi_{comp}(h,h_1,...,h_n)$, which is rule specific, is learned together with the grammar rule, based on the information stored both in the head and the body of the semantic molecules (but applied only to the heads). $\Phi_{onto}(b)$ is built based on a meta-interpreter with *freeze* (Muresan, Potolea, & Muresan 1998), asserting/validating the information from the body of the semantic molecules into/on the ontology.

Learning Framework. The induction of the constraint-based grammar is done incrementally, using a relational learning framework based on Inverse Entailment. Unlike other relational learning methods that use randomly-selected examples and for which the class of efficiently learnable rules is limited, our algorithm learns from an ordered set of representative examples, allowing a polynomial efficiency for more complex rules. Moreover, the size of this set is small and thus our algorithm is able to learn when no large annotated treebanks can be easily built. For each representative example a cover set algorithm performs two steps: 1) the most specific constraint rule generation using a robust bottom-up active chart parser and 2) the generation of the final hypothesis based on the most specific rule generalization using a set of heuristics. The process continues iteratively until all the representative examples are covered. The algorithm is linear on the length of the learned hypothesis.

Input: representative example (w, w'):
(chronic # disease, [cat=n,head=Y] $\bowtie$
[X.isa=chronic,Y.Pn=X,Y.isa=disease])
Most specific constraint rule:
N1(h $\bowtie$ b) $\Rightarrow$ Adj(h1 $\bowtie$ b1), Noun(h2$\bowtie$ b2):
$\Phi_{comp}(h,h1,h2),\Phi_{onto}(b)$
Output: final grammar rule
N1(h $\bowtie$ b) $\Rightarrow$ Noun(h1 $\bowtie$ b1) : $\Phi_{comp}(h,h1),\Phi_{onto}(b)$
N1(h $\bowtie$ b) $\Rightarrow$ Adj(h1 $\bowtie$ b1), N1(h2 $\bowtie$ b2):
$\Phi_{comp}(h,h1,h2),\Phi_{onto}(b)$

The learning engine uses both annotated and unannotated sets of examples. First, the cover set algorithm is based only on the representative set that is semantically annotated (pairs of strings and their semantic molecules). During the generation of the final hypothesis, weakly annotated (only chunked), and optionally unannotated examples are used for the performance criteria in choosing the best rule. Also negative examples are used, if needed, given also as weakly annotated data. During learning background knowledge is used, containing: 1) the ontology, 2) the lexicon that specifies for each word its concept in the ontology and its semantic molecule, 3) the previously learned grammar and 4) the previously learned compositional semantic constraints.

Terminological Knowledge Base. The constraint-based grammar induction framework is applied to build a medical terminological knowledge base. I focus on definitions as my corpus since they are a rich source of conceptual information. This corpus is automatically extracted from on-line articles using our system DEFINDER (Muresan & Klavans 2002). Using the learned grammar and a bottom-up active chart parser, the definitions are semantically parsed. Ideally, a definition should contain the necessary and sufficient conditions to place a concept within a conceptual system. In this ideal setting after semantic parsing, a definition can be asserted to the knowledge base through the use of our meta-interpreter. However, a challenge of our corpus is that it is heterogeneous, containing many definitions for the same medical term. My solution is to merge these definitions, process in which I identify similarities and differences, and then to add the fused definition to the knowledge base.

Current Status and Plan for Completion

To date, the constraint-based grammar induction framework is implemented and has been applied to a fragment of the definitional corpus (e.g., complex noun phrases with prepositional phrases, active/pasive forms of verbs). Moreover, the formal specification of the new theoretical concepts together with the proof of the *LWFG* Induction Theorem was published in (Muresan, Muresan, & Klavans 2004).

Currently I am extending the grammar coverage to noun-noun compounds and nominalizations. I plan to implement a statistical refinement algorithm that will be applied after the grammar learning. I will evaluate the grammar induction given different settings: 1) with and without ontology constraints; 2) with and without statistical refinement. The merging algorithm needs to be implemented. I plan to evaluate it using human judgments (for redundant and missing information). I will also perform an evaluation for identification of paraphrases by comparison to an existing state-of-the art paraphrase identification system.

Acknowledgements. I would like to thank my advisor Judith Klavans, Regina Barzilay, Noemie Elhadad and Tudor Muresan for useful discussions and feedback. This work is supported by NSF DLI-II Grant No. IIS 98-17434.

References

Muresan, S.; Muresan, T.; and Klavans, J. 2004. Inducing Constraint-based Grammars from a Small Semantic Trebank. In *Proceedings of AAAI Spring Symposium on Language Learning:An Interdisciplinary Perspective*. Stanford Univ., CA.

Muresan, T.; Potolea, R.; and Muresan, S. 1998. Amalgamating CCP with Prolog. *Scientific Journal of Politechnics University, Timisoara* 43(4).

Muresan, S., and Klavans, J. L. 2002. A Method for Automatically Building and Evaluating Dictionary Resources. In *Proceedings of LREC 2002*.