ILP Techniques for Free-text Input Processing

Svetla Boytcheva

Department of Information Technology,
Faculty of Mathematics and Informatics,
Sofia University "St. Kliment Ohridski",
5 J. Bauchier Blvd., 1164 Sofia, Bulgaria,
svetla@fmi.uni-sofia.bg

Abstract. This paper presents an application of some Inductive logic programming (ILP) techniques for checking user's answer correctness in a Computer-Aided Language Learning (CALL) system STyLE (Scientific Terminology Learning Environment). STyLE supports adaptive learning of English terminology with a target user group of non-native English speakers. In STyLE are implemented many original features that make this system intelligent and adaptive, but we will focus only on one of them: supporting learner-system communication in Natural Language (NL). The proposed ILP system RICH is used for generation of least generalization(LG) and greatest specialization(GS) of the set of possible correct answers of a given question to the user from the system. The user's answer is correct if it is between LG and GS of the correct answers' set.

Keywords: *Inductive Logic Programming, Natural Language Processing, Free-text input*

1 Introduction

Supporting free NL input requires integration of complex NLP techniques, esp. parsing and checking the correctness of the learner's NL answer. A number of prototypes try to cope with the (almost free) NL input but according to [4] "so few of these systems have passed the concept demonstration phase". The prototypes in [4] contain mostly modules for checking students' competence in vocabulary, morphology, and correct syntax usage (parsers). The most sophisticated semantic analysis is embedded in `BRIDGE/MILT` which matches the learner's utterance (a lexical conceptual structure) against the prestored expected lexical conceptual structures. More recent systems (`CASTLE` in `RECALL` [5] and `SLALOM` [6]) still focus on spelling, morphological, and syntactic errors. Another example is `CIRCSIM-Tutor` [7], which expects quite short answers, permissively extracts whatever is needed and ignores the rest. To conclude, every CALL system pretending for some intelligence has to decide how to analyse learners' NL inputs and check their correctness but the present solutions especially for semantic analysis are far from being perfect.

This paper presents an application of some ILP techniques for checking user's answer correctness in a CALL system `STyLE` [3]. `STyLE` supports adaptive learn-

ing of English terminology with a target user group of non-native English speakers. In STyLE are implemented many original features that make this system intelligent and adaptive, but we will focus only on one of them: supporting learner-system communication in NL. This type of communication in STyLE is supported by its module STyLE-Parasite, which provides a mechanism for checking the correctness of learner's NL utterances. On the other hand STyLE-Parasite uses the system Parasite as an NLU machine.

Section 2 describes the system Parasite. Section 3 deals with the mechanism for checking the correctness of the learner's NL utterances. Section 4 describes application of ILP techniques. In Section 5 are given some examples. Section 6 gives the conclusion.

2 System Parasite

The system Parasite, developed at UMIST by Allan Ramsay, see e.g. [8] and [9], is already integrated in STyLE as an NLU machine for analysing learners' free utterances.

Parasite works using a lexicon, syntax grammar rules and a knowledge base of type (word) hierarchy and meaning postulates. The lexicon contains the morphological description of the words recognised in the input text. The grammar currently covers most of the English syntax, including complex embedded sentences. The hierarchy is a DAG (directed acyclic graph). The meaning postulates define in logical format the word semantics. It is not obligatory to define in advance the semantics of each word to be processed; the designer only has to keep in mind that the prover of the semantic correctness works with the available postulates. Parasite is an open system and allows for the insertion of new words, grammar rules and meaning postulates. When started Parasite checks the KB consistency (contradictions, loop definitions).

As a typical NLU artefact (in contrast to some prototypes for automatic KA),Parasite analyses every input string. It processes separate sentences as well as extended discourse of several sentences. Given a text paragraph, the user might choose analysis type: either independent analysis sentence by sentence, or analysis of all sentences as coherent discourse.

The analysis is performed step by step, starting by morphological and syntactic analysis. Diagnostics is available in cases of unknown or non-correctly derived words, as well as for wrong or ambiguous sentence structure. Soft parsing techniques provide correct analysis of sentences with "small" syntax errors (e.g. wrong subject-verb agreement). Some ambiguity types are resolved by heuristically predefined preference scores; currently the PP-attachment problems are tackled. Syntax analysis fails in case of unknown input words and unresolvable ambiguities.

After correct syntax analysis Parasite performs semantic analysis (see [1, 11, 12]). Meaning postulates are encoded in a language which is a dynamic, constructive version of Ray Turner's 'property theory' [11].

For instance, the definition of "capital market":*"The capital market is an institutional mechanism which deals with capital goods."* can be translated as following Meaning postulate:

```
lexicalMP(
forall(X :: {capital_market(X)}, institutional_mechanism(X) &
         exists(Z::{deal(Z)}, theta(X,$agent,Z) &
          exists(Y::{capital_goods(Y)}, theta(Y,$object,Z))))).
```

3 Mechanism for checking the correctness of the learner's NL utterances

Answers in free English are linguistically analysed by the NL Understanding component `Parasite`. An especially implemented prover `STyLE-Parasite` checks whether the linguistically correct student's answer is correct as an answer to the particular exercise performed at the moment.

An especially performed user study [10] investigated how erroneous answers appear in terminology learning. Errors are usually caused by the following reasons:

- Language errors (spelling, morphological, syntax errors);
- Question misunderstanding, which causes wrong answer;
- Correct question understanding, but absent knowledge of the correct term, which implies usage of paraphrases and generalisation instead of the expected answer;
- Correct question understanding, but absent domain knowledge, which implies specialisation, partially correct answers, incomplete answers and wrong answers.

In principle `Parasite` covers errors due to the first two cases while the prover `STyLE-Parasite` discovers errors due to the two later cases. Parasite provides advanced NL understanding in cases when the learner is given the opportunity to type in freely. The expected answers are simple declarative sentences although `Parasite` handles complex sentences as well as simple discourse consisting of several sentences. Analysing the English input and its linguistic consistency, `Parasite` returns a model of the correct answers or indications of four kinds of errors: (i) unknown word, (ii) morpho, (iii) syntax and (iv) wrong. However, to know that an input utterance is linguistically correct is not enough in CALL, for instance "John loves Mary" is linguistically correct but does not answer the question "who does trade stocks on the primary market". Therefore a second step is necessary, to find out whether the given utterance is reasonable as an answer to the exercise being performed. `STyLE-Parasite` checks the answers' correctness against the available domain knowledge and the expected answer. Most generally, `STyLE-Parasite` takes the logical form built by `Parasite` , "compares" it to the logical forms of the predefined expected minimal and maximal

answers and makes the necessary inferences [2, 3]. Figure 1 presents the eight possible cases of intersections of the terms in the three logical forms and shows how `STyLE-Parasite` decides about the correctness of the input logical form (which strongly depends on the lexical choices and the syntactic structure of the concrete input). Since there might be many correct answers and their language expression varies considerably, it is not practical to compare the input to a single predefined correct logical form. Rather, `STyLE-Parasite` uses prestored maximal and minimal logical forms. Adding new terms to the maximal answer might be redundant or wrong. `STyLE-Parasite` inference is sound [2] but not complete, because the conclusion "(partially) correct learner utterances" is indicated after the first correct binding of variables. `STyLE-Parasite` returns the following indications of semantic mistakes: (i) correct, (ii) more general, (iii) more specific, (iv) paraphrase (usage of concept definition instead of the proper term), (v) incomplete, (vi) partially correct, (vii) wrong and (viii) combination of several mistakes.

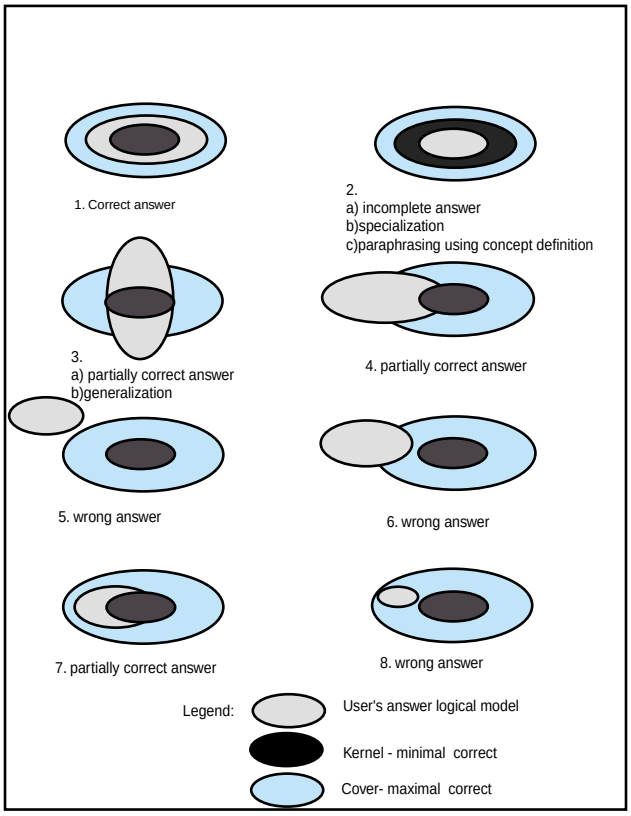

Fig. 1. Comparison and inference of logical forms

4 Application of ILP Techniques

Let create clauses from the logical models generated after `Parasite` analyses, where this model is used as a body of the clause and all clauses have one and same predicate symbol "answer" with arity 1.

To generate sets of minimal and maximal correct answers we will use some ILP Techniques. First we will give some preliminary definitions and results.

Definition 1: (subsumption) Let C and D be clauses. We sat that *C subsumes D*, denoted as $C \geq D$ if there exists a substitution θ such as $C\theta \subseteq D$.

In order subsumption we will say that the clause C is more general than the clause D (or dually D is more specific than C) if C subsumes D.

Definition 2: (implication) Let C and D be clauses. We sat that *C logically implies D*, denoted as $C \models D$ if every model of C is also a model of D.

In order implication we will say that the clause C is more general than the clause D (or equivalently D is more specific than C) if C logically implies D.

Corollary 1 If $C \geq D$ then $C \models D$. The converted does not hold.

Lemma (Gottlob) Let C and D be clauses, which are non-tautologous. If $C \models D$ than $C^+ \geq D^+$ and $C^- \geq D^-$, where by C^+ are denoted positive literals in C and C^- denotes negative literals in C.

Following arguments mentioned in section 3 we can formulate the following theorem:

Theorem Let C be the clause representing the minimal correct answer of a question, D be a clause representing the maximal correct answer of the same question and U be a clause representing user's answer on this question. Then U is a correct answer iff $K \models U$ and $U \models C$.

Proof: 1. Let U be a correct answer, hence there exists a substitution θ such as $\theta K \subseteq U$, because U as a correct answer contains the logical model of minimal correct answer. Form Definition 1 follows that $K \geq U$. From Corollary 1 follows that $K \models U$. Dually U as a correct answer includes in the logical model of the maximal correct answer, hence there exists a substitution σ such as $U\sigma \subseteq C$. Form Definition 1 follows that $U \geq C$. From Corollary 1 follows that $U \models C$.

2. Let $K \models U$ and $U \models C$. From Lemma (Gettlob) follows that $K^- \geq U^-$ and $U^- \geq C^-$. Hence there exists substitutions θ and σ such as $K^-\theta \subseteq U^-$ and $U^-\sigma \subseteq C^-$. But negative literals in U represents the logical model of the user's answer. Hence the logical model of U contains K and includes in C. Hence U is a correct answer.

Corollary 2 The minimal correct answer of a question is a least generalization under implication (LGI) of all correct answers of this question.

Corollary 3 The maximal correct answer of a question is a greatest specialization under implication(GSI) of all correct answers of this question.

Hence for generating the set of minimal correct answer we can use some ILP algorithms for generation of LGI and GSI. We will use a system RICH (Relative Implication of Clauses of Horn) [13]. In RICH are implemented algorithms for specialization and generalization under relative implication. Both the set of clauses and background knowledge (BK) sets processed by RICH are finite sets of function-free Horn clauses with some restrictions. RICH is an empirical non-interactive single predicate learning system. RICH can generate new predicates. RICH uses direct constructing of hypothese approach, instead of seraching the hypotheses space. The main methods for constructing hypothesis is covering approach, unification algorithm, anti-unification algorithm and resolution. The main idea of the algorithm for inducing least generalization under relative implication is sketched on Fig. 2:

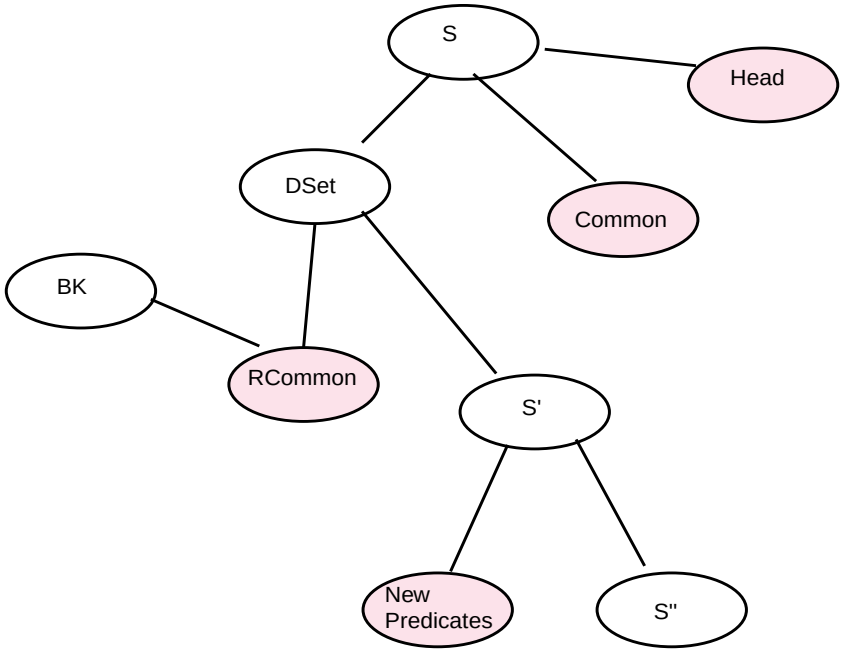

Fig. 2. LGRI algorithm of system RICH

Where S is the set that to be generalized, BK represents the background knowledge set. Head contains head of the hypothesis generated by anti-unification

algorithm. Common is a greatest subset of S for which exists most general unifier and DSet is its corresponding disagreement set resulted of unification algorithm. RCommon is a greatest subset of the set of all resolvents of BK and DSet for which exists most general unifier and S' is its corresponding disagreement set resulted of unification algorithm. NewPredicates is a set of autimatically generated new predicates from S'. S" contains literals from S' that to be dropped.

The final hypothesis is constructed from literals in sets Head, Common, RCommon and head literals from the set NewPRedicates. In more details algorithm is presented in [13].

In application of RICH in `STyLE-Parasite` we do not have background knowledge set.

5 Example

In these section we will show an example for generation of minimal and maximal correct answers' sets using ILP system RICH.

Question (1):
Each of the statements below describe a characteristic of one major type of market. Which? Supports the building of homes, factories, shopping centres.

Some possible correct answers:

(2.1) This situation describes capital market.
(2.2) This is capital market.
(2.3) This characteristics describe capital market

The corresponding clauses created from `Parasite`'s logical model for each of these answers are:

```
(3.1)
answer(N152):-theta(N144152,Object3,N143152),
              capital_goods(N144152),
              deal(N143152),situation(N153),
              theta(N152,Agent3,N143152),
              institutional_mechanism(N152),
              capital_market(N152),market(N152),
              associated_capital(N152),
              theta(N150,Object4,N152),
              theta(N150,Agent4,N153),
              describe(N150).
(3.2)
answer(N149):-theta(N144149,Object5,N143149),
              capital_goods(N144149),deal(N143149),
              theta(N149,Agent5,N143149),
              institutional_mechanism(N149),
              capital_market(N149),market(N149),
```

```
              associated_capital(N149),
              theta(N147,Pred1,N149),
              theta(N147,Topic1,N1475072),
              predication(N147).
(3.3)
answer(N159):-theta(N144159,Object1,N143159),
              capital_goods(N144159),deal(N143159),
              characteristic(N160),
              theta(N159,Agent1,N143159),
              institutional_mechanism(N159),
              capital_market(N159),market(N159),
              associated_capital(N159),
              theta(N157,Object2,N159),
              theta(N157,Agent2,N160),
              describe(N157).
```

The genreated LGI (minimal set) of these correct answers' set from RICH system will be the following hypothesis:

```
(4)
answer(N5569):-institutional_mechanism(N5569),
               capital_market(N5569),market(N5569),
               associated_capital(N5569),
               capital_goods(N5601), deal(N5633),
               theta(N5569,Agent9,N5633),
               theta(N5601,Object9,N5633).
```

The generated GSI (maximal set) of these correct answers' set from RICH system will be the following hypothesis:

```
(5)
answer(N152):-theta(N144152,Object3,N143152),
              capital_goods(N144152),
              deal(N143152),situation(N153),
              theta(N152,Agent3,N143152),
              institutional_mechanism(N152),
              capital_market(N152), market(N152),
              associated_capital(N152),
              theta(N150,Object4,N152),
              theta(N150,Agent4,N153),
              describe(N150),
              theta(N147,Pred1,N152),
              theta(N147,Topic1,N1475072),
              predication(N147),
              characteristic(N153).
```

For instance, let we have the following users' answers of this question: (6a) *This is an institutional mechanism which deals with capital goods.* (6b) *This is a financial market that operates with debt instruments.*

The logical model created from `Parasite` of (6a) is:

```
answer(N140):- theta(N144140,Object6,N143140),
               capital_goods(N144140),deal(N143140),
               theta(N140,Agent6,N143140),
               institutional_mechanism(N140),
               capital_market(N140),market(N140),
               associated_capital(N140),
               theta(N141,Pred1,N140),
               theta(N141,Topic1,N1475072),
               predication(N141).
```

`STyLE-Parasite` will classifies (6a) as "paraphrase of the correct answer", because LGI (4) is not a logical implication from (6a), but GSI(5) logically implies (6a).

`STyLE-Parasite` will classifies (6b) as "wrong answer", because neither LGI (4) is a logical implication from (6b), nor GSI(5) logically implies (6b).

6 Conclusion

One of the crucial points in implementation of complex learning systems is knowledge base building. Non-automatic generation of minimal and maximal correct answers' sets is rather hard and requires extended knowledge about the system. Presented approach allows automatic generation of minimal and maximal correct answers' sets and avoids necessity knowledge expert to be familiar with rather complex internal data representation. This approach is independent of knowledge domain of the learning system as far as `Parasite`'s lexicon and meaning postulates base have to include this domain concepts. Generation of minimal and maximal correct answers' sets is a pre-process and `STyLE-Parasite`'s efficiency does not depends of it during the learning sessions. Presented approach is a step toward in free-text input processing.

References

1. Cryan, M and A M Ramsay, A Normal Form for Property Theory, 14th Conference on Automated Deduction, 1997, Springer Lecture Notes in Computer Science 1249, pp. 237-251
2. Boytcheva, Sv., O. Kalaydjiev, A. Strupchanska, G. Angelova, Between Language Correctness and Domain Knowledge in CALL, Proc. of European Conference Recent Advances in Natural Language Processing (RANLP-2001), Bulgaria, pp. 40-46, 2001.

3. Angelova G., S. Boytcheva, O. Kalaydjiev, S. Trausan-Matu, P. Nakov and A. Strupchanska, Adaptivity in Web-Based CALL, Proc. of the 15th European conference on Artificial Intelligence (ECAI - 2002), 21-26 July 2002, Lyon, France, 2002 (to appear).
4. Holland, V. M., Kaplan, J. and M. Sams (eds.) Intelligent Language Tutors: Theory Shaping Technology. Lawrence Erlbaum Associates, UK, 1995.
5. RECALL, a Telematics Language Engineering project (1997), http://iserve1.infj.ulst.ac.uk/ recall.
6. McCoy, K., Pennington, C.A., and Suri, L.Z. English error correction: A syntactic user model based on principled "mal-rule" scoring. In Proc. 5th User Modelling, 1996, pp. 59-66.
7. Glass, M. Processing Language Input in the CIRCSIM-Tutor Intelligent System. AAAI 2000 Fall Symposium on Building Dialogue Systems for Tutorial Applications. http://www.csam.iit.edu/ circsim/index.html
8. Ramsay, A. Meaning as Constraints on Information States, in Rupp, Rosner, Johnson (eds.) Constraints, Language and Computation, 1994, Academic Press, London: 249-276.
9. Ramsay, A. and H. Seville. What did he mean by that? Proc. AIMSA-2000, Bulgaria, Sept. 2000, LNAI 1904, Springer, pp. 199-209.
10. Vitanova, I. Learning Foreign Language Terminology: the User Perspective. Larflast report 8.1, August 1999, submitted to the European Commission.
11. Ramsay, A. , Theorem Proving for Intensional Logic, Journal of Automated Reasoning 14, 1995, pp. 237-255
12. Ramsay A., Does It Make Any Sense? Update Semantics as Epistemic Reasoning, see http://www.co.umist.ac.uk/staff/ramsay.htm
13. Boytcheva, S., Z. Markov, An Algorithm for inducing least generalization under relative implication, In Proc. of the 15th International conference of Florida Artificial Intelligence Research Society (FLAIRS-2002), AAAI Press, 13-16 May 2002, Pensacola, Florida, USA, pp. 322-326, 2002.