Knowledge Extraction from Text: Machine Learning for Text-to-rule Translation

J. F. Delannoy, C. Feng, S. Matwin, and S. Szpakowicz
Department of Computer Science
University of Ottawa
Ottawa, Ontario, Canada K1N 6N5
{delannoy,cfeng,stan,szpak}@csi.uottawa.ca

Abstract

Learning from texts is a noble but distant research goal. We investigate the practicality of a more modest enterprise in which machine learning (ML) and natural language processing (NLP) would mutually reinforce each other. We are designing a system that will process technical expository texts, in which narratives (embodying general knowledge) and examples (specific knowledge) are interleaved to best instruct the reader. A broad-coverage parser of English technical texts and an interactive *case analyzer* provide the front end. Their output will be translated into *Horn clauses*; some user participation will be required throughout the process. We discuss how the application of *absorption* will provide a useful hierarchization of the domain theory. We then apply the EBL approach, using translation of the narratives as the domain theory, and translation of the examples as the training examples. We illustrate with examples that such learning may result in a reformulation of knowledge that would not be attainable by NLP techniques alone. We present the overall design of the system, with the emphasis on the ML contribution to the synergistic effect we expect. We also briefly consider the applicability of ML techniques to the generalization of the user-directed translation itself.

1 Motivation and background

At the 1990 ICML meeting, Tom Dietterich suggested learning from texts as an important challenge for Machine Learning in the 1990's. The ability to extract knowledge from texts will no doubt prove rewarding, as much human knowledge is written down in textual format, and texts are one of the basic media of human communication. So far, there have been only a few answers to this challenge, like the work of [Cohen, 1990] and [Moulin and Rousseau, 1992]. Cohen, working on the card game of bridge, *manually* converted to Horn clauses a text containing almost complete knowledge about the rules of the game. This knowledge was then used to perform explanation-based generalization (EBG) on examples, and the main focus of his work was to determine the right level of operationality from the explanations of the examples. Moulin and Rousseau chose a more realistic domain (the National Building Code of Canada - a set of safety rules and constraints for building design) and used natural language processing techniques to generate rules from the text. Their system, however, performs no learning at either knowledge or symbol level [Dietterich, 1986]. Each of these two methods tackles only one part of knowledge extraction from texts, without addressing the other.

We propose to work on knowledge extraction from texts in its *entirety*, i.e., incorporating both text conversion and learning. We are developing a system called MaLTe (Machine Learning from Text), Figure 1, that will accept a broad class of real-life texts [Delisle and Szpakowicz, 1991], and return a representation of the knowledge contained in the text [Matwin and Szpakowicz, 1992], including the examples that belong to the text as long as they are expressed in natural language. We believe firmly that learning will be a necessary component of our system, i.e., that text analysis *without* learning will produce *less* knowledge, and of a *shallower* nature, than text analysis combined with learning techniques. In this sense we go far beyond the initial work exemplified by [Moulin and Rousseau, 1992] and [Cohen, 1990]. For example, our current domain - part of the personal income tax law as described in (Revenue Canada 1991) - contains *deep* knowledge which cannot be made explicit and used efficiently by a simple learning strategy such as in Cohen (1991).

This paper presents work in progress. We are currently involved in designing parts of the system, prototyping other parts, and testing yet other, already developed components. Consequently, ideas presented here are a product of discussions led within the MaLTe team for the last several months, but do not yet include results of experimentation with a full, running system. We would also like to mention that our objective is to apply learning *within* a system that extracts knowledge from texts, and not to address cognitively motivated problems of

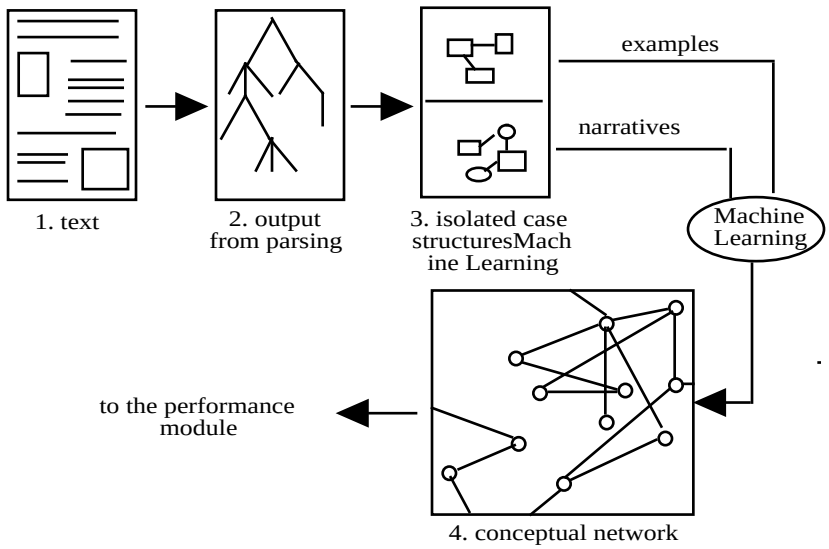

Figure 1: *MaLTe's architecture.*

full-fledged learning of natural language [Powers and Turk, 1991]. We start with the following tenets:

- Fully automatic knowledge extraction from text is practically impossible unless the text describes a toy domain, and unless a thorough representation of common-sense and domain knowledge is available. Instead of such a representation some user participation in the acquisition exercise is required.
- We want to rely as little as possible on pre-existing, coded knowledge. We believe that systems extracting knowledge from texts should operate without extensive knowledge bases; if such bases existed, knowledge extraction from text would not be interesting. Ideally, MaLTe would operate with no other knowledge than what is explicitly present in the text, and what is elicited from the user in order to enrich and reorganize the literal meaning of the text.
- The latter knowledge will be provided on an "as needed" basis, i.e., it will be entered by the user in order to resolve ambiguities, address inconsistencies, and define synonyms encountered in the extraction process, rather than engage in an open-ended knowledge acquisition exercise.
- Most existing systems performing knowledge extraction from texts have difficulty dealing with examples, and many ignore them [Gomez, 1991], [Reimer, 1990]. We believe, however, that examples often provide knowledge not represented elsewhere in the text, and are meant to be generalized by the user. We propose some specific machine learning techniques to provide this capability in MaLTe.

Our work so far has uncovered a number of questions that have to be addressed in order to extract knowledge from texts by means of a mixed, natural language processing and machine learning system:

- What are the linguistic characteristics of an interesting, but realistic class of texts for us to focus on? For example, how does one characterize legal text, regulations, specifications, and technical manuals? (See Section 2 for a list of difficulties stemming from the contents and organization of such texts).
- What knowledge representation adequate and conducive for learning can be produced by the NLP component of the system? For example, in natural language processing semantic networks or Schankian representation based on memory structures are commonly used, but logic-based representation seems more adequate for the kind of learning that we have in mind (see Section 3)
- Can an adequate representation be produced from text with little "handcrafting" and little user interaction? We plan to convert case analyses derived directly from text fragments into Horn clauses useful in EBL. This conversion may require a large amount of domain knowledge and common-sense knowledge; we will stick as much as possible to the principle of deriving it from textual input and from the user - possibly applying a case-based reasoning process to alleviate the burden of frequent interaction by drawing upon experience.
- What learning strategy will be able to generalize the knowledge from the text and yet produce efficient and explicit rules for possible use by an expert system? For example, we may need the generalization techniques used in inductive logic programming and explanation-based learning.

We address these questions in the subsequent sections of this paper.

2 Characteristics of the text

The tax guide we have chosen for the initial experiments (Revenue Canada 1991) can be characterized as a *technical expository* text. The definition of technical texts is beyond the scope of this paper. Informally, such texts pose relatively few linguistic problems. For example, lexical ambiguity is low, the domain is well delineated, reference resolution is straightforward, and so on. Expository texts are well examplified by user's guides: they consist of general narrative sections and specific examples.

An important property of the tax guide is its *advisory* nature, as opposed to a mere normative exposition of the law. The text is organized around practical questions that the reader might ask herself. In particular, it contains sections with headings such as "What are child care expenses?" and "Who is an eligible child?" (definitions), or "What payments can you deduct?" (details of the coverage of eligible payments), and "How much can you or the other person deduct?" (numeric details).

The text contains a number of examples, and they are typically 10 to 15 lines long. For example, the section entitled "What payments can you deduct?" is illustrated with two examples; one of them is shown at the end of Section 2.3.

2.1 Linguistic difficulties with the tax guide as a source of knowledge

Concepts in the tax guid are presented informally. One form of presentation is a list of instances, as in:

``Child care expenses are amounts you pay ... This would include amounts you pay to: friends or neighbours, relatives, nursery schools, ... boarding schools.''

It is unclear whether the list is meant to be exhaustive, and common sense dictates it is not. In our method, however, the text is the authoritative source of knowledge, so that the literal translation must be treated with due suspicion by the text-to-rule translator in MaLTe.

Handling cross-references within the text will be facilitated by the hierarchical representation of the text in MaLTe's text-base, using a set of text organization and management tools developed for the TANKA case analyzer [Copeck *et al.*, 1992]. References pointing to a different text can be considered part of background knowledge.

Some examples contain rather complicated conditional rules. Sometimes examples are incompletely instantiated, and have alternatives, as in:

``Bob and Cathy are living together and have three children... If Bob is the father of only one child, only he may claim the child care expenses deduction for that child... If Bob is the father of all the children ... If Bob is not the father of any of the children ...''

In this example, we might generalize the number of children after checking that the number 3 does not have a special meaning (as from being a threshold in the regulation).[1]

The text contains procedural passages, such as:

``Since Rosa is the supporting person... she would claim the child care expenses. ... she would complete Form T778 ... [and] would enter Miguel's name in section 1(d) as the other supporting person.'',

which can be represented as rules with condition-action semantics.

2.2 Problems concerning the logical structure

The presentation of general rules in the tax guide is not always precise enough. For the rules which have exceptions, the text starts with the general case and only mentions exceptions or restrictions later on. This amounts to non-monotonicity if we process the text sentence by sentence (the paragraph mode will be implemented later). Consider this fragment:

``Child care expenses are amounts you pay ... to relatives 21 years of age or older ... Under proposed legislation, for 1990 and later taxation years, you will be able to deduct amounts paid to a relative 18 years of age or older.''

``Generally, whoever has the 'lower net income' before deducting child care expenses, and social benefit ... deducts child care expenses. However, in some cases, the person with the ''higher net income' can claim the child care expenses.''

The first passage is far from obvious to interpret, but since it does not apply to current legislation it could be eliminated by the user. To deal with the second example, the translation process may have to change the order of the rules so that the more specific ones are examined first.

Not only there is the well-known phenomenon of ambiguous logical connectives (the comma may express disjunction ("or") or conjunction ("and") in a list), but also itemized lists can be interpreted in more than one way:

``You are considered to have been married throughout the entire year if:
- you separated from your spouse during 1990, and you reconciled etc.
- you separated from your spouse during 1990, the

[1] The principle, used in EBG, of generalizing constants not mentioned in the available domain knowledge may be awkward to apply here, because such knowledge might well be lacking if not much text has been processed so far. Alternatively, there are methods for learning with incomplete knowledge.

```
period you were separated was less than 90 days
etc.''.
```

A clue may appear before the last item, but the ambiguity remains until then. In such situations, the parser [Delisle and Szpakowicz, 1991] will have to resort to the user for resolving the problem.

Explanations (elements of proof trace referring to some nodes, but not to rules) are given in some examples: it could be interesting to use them to help EBG and validate its results.

Finally, the guide, as stated in its introduction, "contains information for individuals who incurred child care expenses in 1990" and "explains the expenses [they] may deduct and who is eligible to deduct them", so that the various questions underlying the text, explicitly or implicitly, can be considered to point to a *top-level logical goal* of "claiming child care expense deductions".

2.3 An example

The following is the first example appearing in the child care expenses guide.

```
  Jim, a member of the Canadian Armed Forces,
was posted to Lahr, West Germany in 1989. His
wife Louise and their two preschool children
moved with him to Lahr. They broke all reside-
ntial ties with Canada. Because he is serving
abroad in the armed forces, Jim and his family
are deemed residents of Canada for income tax
purposes.
  While in Lahr, Louise worked part time at a
store on the army base. During 1990, Jim and
Louise paid $1,500 to a German neighbour to
take care of their children while they were at
work. These payments qualify as eligible child
care expenses.
```

Some of the translation from the text into logic format is straightforward, e.g.:

```
isa_member(jim,armed_forces)
```
[2]

However, when there are attachments, as complement expressing location and time, the translation is more complicated, e.g.:

```
post_agt_obj_lto_tat(nil,jim,lahr,year_1989.)
```

The predicate is constructed from the words themselves and from the names of the cases in the output of the analyzer: `lto` for *location to*, `tat` for *time at*, and `manr` for *mannner*; `nil` means undefined. The linguistic justification of TANKA's case system is given in [Copeck *et al.*, 1992]. The clause above translates the whole sentence; however, we cannot be certain at this stage whether the location and time information will be relevant for producing rules, so we must bear in mind that the following are also possible candidates for the sentence:

```
post_agt_obj(nil,jim).
post_agt_obj_lto(nil,jim,lahr).
post_agt_obj_tat(nil,jim,year_1989).
```

Because normally these candidates are few and they can always be distinguished syntactically, this does not result in too much inefficiency.

Sentences about Jim's family are translated into the following:

```
child(child1,jim).
child(child2,jim).
age(child1,preschool).
age(child2,preschool).
```

These, though repetitive, are necessary to help us to capture the information contained in the text. The following are part of the sentence-to-sentence translations of the facts available about Jim's family.

```
deemed_resident(jim).
deemed_resident(louise).
work_agt_manr_lat(louise,part_time,a_store).
pay_agt_obj_recp(jim,dollar_1500,a_neighbour).
take_care_agt_obj(a_neighbour,child1).
take_care_agt_obj(a_neighbour,child2).
child_care_expenses(dollar_1500).
```

3 Learning strategies for text-to-rule translation

Learning can contribute positively to elicit semantic relationships [DeJong and Mooney, 1986] e.g., to complete, correct, and integrate the text-to-rule translations. We are interested in two roles that learning can play; they are *knowledge abstraction* and *explanation-based generalization* (EBG) [Mitchell *et al.*, 1986]. A significant characteristic of texts is the need to abstract implicit knowledge (Section 2). For example, in our task domain the top-level (most abstract) rule for claiming child care expenses is never explicitly stated, though it has been indicated numerous times elsewhere (e.g., in the examples). As a consequence, abstraction is needed to make such information explicit, and to generalize from the examples provided in the text (Figure 2); it brings the *implicit meaning* of the text to the surface (Section 2). We want to abstract a set of facts acquired from examples in the text into higher-level concepts present in the narrative. To achieve this we propose to use *absorption* - an operator used in inductive logic programming (ILP) for knowledge abstraction ([Muggleton, 1992], [Muggleton and Feng, 1992], [Sammut and Banerji, 1986]).

Texts can also be ill-organized when they contain multiple levels of concepts, each level supporting its immediate upper level (Section 2). For example, reasoning

[2] In fact, the constants and variables are typed [Matwin and Szpakowicz, 1992], but for the purpose of this paper we decided to ignore them.

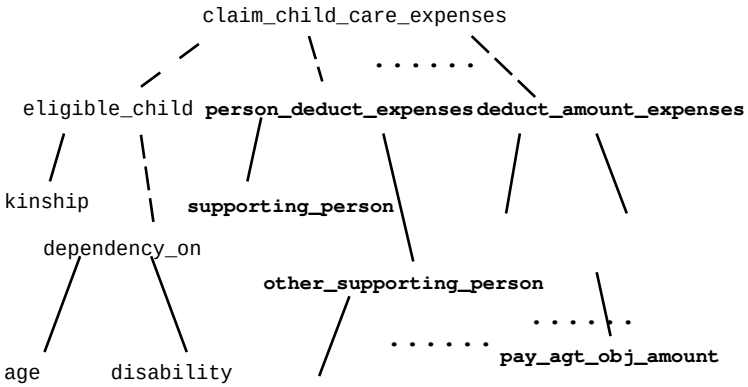

Figure 2: *Abstraction levels of facts to form higher level rules. Note that the dashed lines indicate that the relationship does not exist in the text explicitly.*

through multiple levels is required to establish the validity of "a person has an eligible child for deducting child care expenses". Explanation-based generalization is a well-established technique in machine learning [Mitchell *et al.*, 1986] to compile pieces of knowledge into useful units. EBG can integrate the translated rules so that the most useful features (e.g., "a mentally disabled child") are explicit. This will also *operationalize* a rule, e.g., to make it specifically applicable to a group of persons who are "mentally disabled children". Indeed, this is also recognized by the text writers, e.g., they make special provisions for different subgroups of people – single parents, divorced persons etc.

3.1 Knowledge abstraction

Text writers make many assumptions about the ability of readers. As a result, there exist many omissions, especially if a piece of information seems "obvious" or is "commonsense". For example, the top-level rule for claiming child care expenses is obvious to a human reader, who can infer from the context (e.g., section titles) that the writer of the guide intentionally abstracted the process for claiming child care expenses into a few conditions. We use the example in Section 2.3 to illustrate this.

This is a positive example of "Jim can claim the $1500 in child care expenses". Suppose the background knowledge (extracted from the normative texts) contains the following rules:

```
eligible_child(Child,Person):-
    child(Child,Person),
    age(Child,preschool).
```

The absorption operator, applied on the facts about Jim's family acquired from the text as shown in Section 2.3, will abstract `eligible_child(child1,jim)` from `child(child1,jim)` and `age(child1,preschool)` (i.e., a child must be an eligible child for the parent to claim child care expenses.)

Another abstraction uses the rule:

```
child_care_expenses(Exp):-
    deemed_resident(Person),
    amount_of_money(Exp),
    pay_agt_obj_recp(Person,Exp,SomeOne),
    take_care_agt_obj(SomeOne,Child),
    child(Child,Person).
```

which produces `child_care_expenses(dollar_1500)`. Note that `deemed_resident(Person)` is generalized, using a rule, from the facts:

```
isa_member(jim,armed_forces),
serve_agt_obj_lat(jim,armed_forces,abroad),
break_agt_obj_acmp(jim,residential_ties,canada).
```

After the repeated application of the rules in the background knowledge, we can obtain the following:

```
eligible_child(child2,jim).
eligible_child(child2,jim).
child_care_expenses(jim,dollar_1500).
deduct_child_care_expenses(jim).
deduct_agt_obj_caus(jim,dollar_1500,child1).
deduct_agt_obj_caus(jim,dollar_1500,child2).
```

Note that the facts that are "subsumed" by these facts have been removed. When the constants in the facts are generalized to variables, we can obtain the rule [3]:

```
claim_agt_obj(Person,Exp):-
    eligible_child(FirstChild,Person),
    eligible_child(SecondChild,Person),
    child_care_expenses(Person,Exp),
    deduct_child_care_expenses(Person),
    deduct_agt_obj_caus(Person,Exp,FirstChild),
    deduct_agt_obj_caus(Person,Exp,SecondChild).
```

which is a specialization of the top-level rule shown in Section 2. This rule should only cover examples which involve two children for claiming child care expenses, but it also covers many other cases[4].

[3] We have ignored issues related to clause reduction and over-generalization.

[4] This may be a defect in our representation for dealing

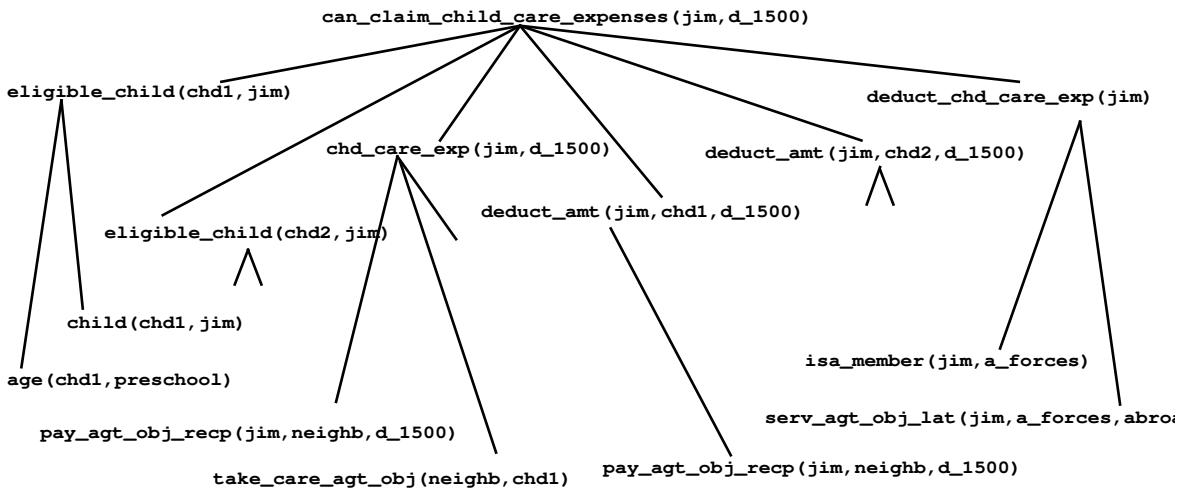

Figure 3: *Part of the explanation tree for the case about Jim's family.*

3.2 Explanation-based generalization

Abstraction can help to organize the knowledge contained in text better, to complete parts that are not explicitly stated, and even to transfer the knowledge to human readers. However, we also need to integrate the knowledge, using EBG, to a level which is operational, to make it efficiently applicable to individual cases. For the case about Jim's family, we have the explanation tree in Figure 3.[5] Given such an explanation, there are existing techniques in EBG that can generalize it according to a given *operationality* criterion.

Operationality represents the knowledge of what features are relevant to the problem, i.e. child care expenses claims. However, this knowledge is usually not given in the text; this has become a crucial problem for us. This issue is further complicated by the fact that there may be variant operationalities for different rules and to different readers. Ideally, this can be solved by including many examples of past claims and generalizing from them (e.g. using the approach in [Cohen, 1990]). To control generalization to obtain *good* operationality, we need to trade-off the coverage of rules against efficiency when executed. In the tax guide the examples are not numerous, so these examples have to come from additional sources (e.g. past claim records).

At present, we assume that the operationality is given. The text presents a good starting point because some of the operationality can be reasonably inferred. The text-drafters, perhaps, have made conscientious effort to include examples that are representative of the knowledge contained in the text. For example, we consider that the following predicates in Jim's case are operational:

```
isa_member/2,
serve_agt_obj_lat/3,
child/2,
age/2,
pay_agt_obj_recp/3,
take_care_agt_obj/2,
```

which gives us the rule:

```
claim_agt_obj(Person,Exp):-
    isa_member(Person,armed_forces),
    serve_agt_obj_lat(Person,armed_forces,abroad),
    child(FirstChild,Person),
    child(SecondChild,Person),
    age(FirstChild,preschool),
    age(SecondChild,preschool),
    pay_agt_obj_recp(Person,Exp,SomeOne),
    take_care_agt_obj(SomeOne,FirstChild),
    take_care_agt_obj(SomeOne,SecondChild).
```

obtained by EBG from the top-level rule shown at the end of Section 3.1.

4 Conclusion and future work

In this paper, we described the ongoing MaLTe project that involves knowledge extraction from expository texts. We have shown the various characteristics of such texts and illustrated the basic ideas that are used to translate the text into logic format through users' guidance. We intend to follow the principle of staying as close to the text as possible. Text structure poses some linguistic problems, but the difficulties lie more with semantic ambiguities and lack of information. This is addressed through the incremental building of a knowledge base in interaction with the user, rather than by separate handcrafting. Linguistic analysis will be performed by the DIPETT parser and the TANKA case analyzer [Copeck *et al.*, 1992].

with sets of individuals. We are working on more robust representations which may eliminate such problems.

[5]Of course, in generating such explanations we need to consider the incompleteness of the knowledge derived from the text. Because of limited space, we will not discuss this issue in this paper.

Knowledge converted in this way can be implicit and ill-organized. To overcome this, we exploit various abstraction and EBG techniques developed in machine learning. In particular, we have shown how absorption and explanation-based generalization can be used to streamline the translated text and to care for individual cases. However, at this early stage of the project, we have not investigated a number of problems that are crucial to the success of our effort:

Incomplete knowledge. The prime requirement of EBG is the completeness of the domain knowledge. Our domain knowledge comes from the text, and thus it may not be complete. When this happens we need to devise a method to complete the domain knowledge using only the existing rules and the examples provided in the text.

Over-generalization. When absorption is applied, the clause produced may be over-general. This problem can be alleviated by including more examples. However, our text does not contain many such examples.

Rule reduction. Some facts may not be "absorbed" by any rules from the domain knowledge. So they will remain after repeated applications of absorption. The question arise: whether they should be part of the conditions of the final rule? Should they be ignored, and what is the justification?

Operationality. The text does not provide an operationality criterion. This needs either to be inferred from the text or to come from outside the text. Ideally, we would prefer the former because it does not involve knowledge outside of the text. We have not solved this problem.

We also intend to explore the application of learning methods to text-to-logic conversion. We believe that a person asked to convert a text systematically into Horn clauses would apply some meta-rules codifying her understanding of the text. For example, the verb "go" in the sentence "Joe goes to university" is likely to be ultimately represented by the predicate "attend". Our system would remember this substitution and propose to apply it if it later encountered the verb "go" in the same context. Other possible conversions of "go" may be learned, depending on the semantics of the arguments of the verb "go". The meta-rules acquired in this fashion may even be generalized syntactically (e.g. if they mention the same case markers but different verbs) and semantically (e.g. if they mention nouns closely related in some ontology).

We expect that, for every new domain, such a system would initially require a large amount of user interaction to produce adequate Horn clauses. Gradually, as more conversion rules are acquired and reapplied, the amount of interaction should decrease according to a standard learning curve. It is one of the objectives of our future work to investigate this hypothesis. We will build such a linguistic learner and experiment with it while testing other components of the MaLTe system.

Acknowledgements

This research is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) under grant STR0117765.

Thanks to Denys Duchier for state-of-the-art LaTeX wizardry.

References

[Cohen, 1990] W. Cohen. Learning from textbook knowledge: a case study. In *AAAI-90 : proceedings of the Eighth National Conference on Artificial Intelligence.*, pages 743 – 748, Boston,MA, 1990.

[Copeck *et al.*, 1992] T. Copeck, S. Delisle, and S. Szpakowicz. Parsing and case analysis in TANKA. In *Proceedings of the 15th Intl. Conf. on Computational Linguistics COLING-92*, pages 1008–1012, Nantes, 1992.

[DeJong and Mooney, 1986] G. DeJong and R. Mooney. Explanation-Based Learning: An Alternative View. *Machine Learning*, 1(2):145–176, 1986.

[Delisle and Szpakowicz, 1991] S. Delisle and S. Szpakowicz. A broad-coverage parser for knowledge acquisition from technical texts. In E. Johnson, ed., *Proc. Fifth Intl. Conf. on Symbolic and Logic Computing*, pages 169–183, 1991.

[Dietterich, 1986] T. Dietterich. Learning at the Knowledge Level. *Machine Learning*, 1(3):287–316, 1986.

[Gomez, 1991] F. Gomez. The Acquisition of Word Syntactic Knowledge from Sample Sentences *6th Knowledge Acquisition from Knowledge-Based Systems Workshop*, Banff, Alberta (Canada), pages 9-1 – 9-17, 1991.

[Matwin and Szpakowicz, 1992] S. Matwin and S. Szpakowicz. Machine learning techniques in knowledge acquisition from text. *THINK*, 1 (2):37–50, 1992.

[Mitchell *et al.*, 1986] T.M. Mitchell, R.M. Keller, and S.T. Kedar-Cabelli. Explanation-Based Generalization: A Unifying View. *Machine Learning*, 1(1):47–80, 1986.

[Moulin and Rousseau, 1992] B. Moulin and D. Rousseau. Automated knowledge acquisition from regulatory texts. *IEEE Expert*, Oct.:27-35, 1992.

[Muggleton and Feng, 1992] S. Muggleton and C. Feng. Efficient Induction of Logic Programs. In *Inductive Logic Programming.* Academic Press, 1992.

[Muggleton, 1992] S.H. Muggleton. *Inductive Logic Programming.* Academic Press, 1992.

[Powers and Turk, 1991] D. Powers and C. Turk. *Machine Learning of Natural Language.* Springer-Verlag, Berlin, 1991.

[Reimer, 1990] U. Reimer. Automatic Knowledge Acquisition from Texts: Learning Terminological Knowledge via Text Understanding and Inductive Generalization *6th Knowledge Acquisition from Knowledge-Based Systems Workshop*, Banff, Alberta (Canada), pp 17-1 - 17-16, 1990.

[Sammut and Banerji, 1986] C. Sammut and R. Banerji. Learning Concepts by Asking Questions. In R. Michalski, J. Carbonell, and T. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach. Vol. 2*, pages 167–192. Morgan Kaufmann, Los Altos, CA, 1986.