

Acquisition of a Language Computational Model for NLP

Svetlana SHEREMETYEVA
Computing Research Laboratory
New Mexico State University
Las Cruces, NM, USA 88003
lana@crl.nmsu.edu

Sergei NIRENBURG
Computing Research Laboratory
New Mexico State University
Las Cruces, NM, USA 88003
sergei@crl.nmsu.edu

Abstract

This paper describes an approach to actively acquire a language computational model. The purpose of this acquisition is rapid development of NLP systems. The model is created with the syntax module of the Boas knowledge elicitation system for a quick ramp up of a standard transfer-based machine translation system from L into English.

Introduction

Resource acquisition for NLP systems is a well-known bottleneck in language engineering. It would be a clear advantage to have a methodology that could provide a much cheaper way of NLP resources acquisition. The methodology should be universal in the sense that it could be applied to any language and require no skilled labour of professionals. Our approach attempts just that.

We describe it on the example of the syntax module of the Boas knowledge elicitation system for a quick ramp up of a standard transfer-based machine translation system from any language into English (Nirenburg 1998). This work is a part of an ongoing project devoted to the creation of resources for NLP by eliciting knowledge from informants.

1 Other Work on Syntax Acquisition

Experiments in “single-step” automatic acquisition of knowledge have been among the most fashionable topics in NLP over the past decade. One can mention work on automatic acquisition of phrase structure using distribution analysis (Brill et al 1990). The problems with

the current fully automatic corpus-based approaches include difficulties of maintaining any system based on them, due to the opaqueness of the method and the data to the language engineer. At the present time, the most promising NLP systems include elements of both corpus-based and human knowledge-based methods. One example is acquisition of Twisted Pair Grammar (Jones and Havrilla 1998) for a pair of English and a source language (SL). Another example of a mixture of corpus-based and human knowledge-based methods is a system to generate a Lexicalized Tree-Adjoining Grammar (F. Xia et al. 1999) automatically from an abstract specification of a language. Grossly simplifying and generalizing due to lack of space, one can state that these experiments are seldom comprehensive in coverage and their results are not yet directly useful in comprehensive applications, such as MT.

2 Acquisition of Syntax in Boas

2.1 Methodologies for Selection of Syntax Parameters

In general, the issue of the selection of parameters for grammar acquisition is one of the main problems for which there is no single answer. Parameters applicable to more than one language are studied in the field of language universals as well as the principles-and-parameters approach (Chomsky 1981) and its successors (Chomsky 1995). Widely devised as the basis of universal grammar, the principles-and-parameters approach has focused on the universality of certain formal grammatical rules within that particular approach rather on the substantive and exhaustive list of universal parameters, a subset of which is applicable to each natural language, along with their

corresponding sets of values, such as a parameter set of nominal cases. In some other approaches, parameters and parameter values are either not sought out or are expected to be obtained automatically (e.g. Brown et al. 1990; Goldstein 1998), and, while holding promise for the future as a potential component of an elicitation system, cannot, at this time, form the basis of an entire system of this kind.

In order to ensure uniformity and systematicity of operation of a language knowledge elicitation system, such as Boas, it is desirable to come up with a comprehensive list of all possible parameters in natural languages and, for each such parameter, to create a cumulative list of its possible values in all the languages that Boas can expect as SLs. Three basic methodological approaches are used in Boas.

Expectation-driven methodology: covering the material by collecting cross-linguistic information on lexical and grammatical parameters, including their possible values and realizations, and asking the user to choose what holds in SL; while it is beyond the means of the current project to check all extant languages for possible new parameters, we have included information from 25 languages.

Goal-driven methodology: in the spirit of the “demand-side” approach to NLP (Nirenburg 1996) Boas was tailored for elicitation of MT relevant parameters rather than any syntactic parameters that can be postulated. A parameter was considered to be relevant if it was necessary for the parser and the generator used in MT in the Expedition project (<http://crl.NMSU.Edu/expedition/>).

The parser used is a heuristic clause chunker developed at NMSU CRL which replaces the complex system of phrase structure rules in a traditional grammar and uses language specific information, among them word order (SVO vs. SOV), clause element (subject, object, etc.) marking, agreement marking, noun phrase structure pattern, position of a head.

Data-driven methodology: prompting the user by English words and phrases and requesting translations or other renderings in SL; data-

driven acquisition is the first choice, wherever feasible, because it is the easiest type of work for the users¹; In Boas, data-driven acquisition is guided by the resident English knowledge sources.

2.2 Types of Syntax Parameters in Boas

The parameters which are elicited through the syntax module of Boas include² what we call diagnostic and restricting parameters.

Diagnostic parameters are those whose values help determine clause structure for correct structural transfer and translation of clause constituents. For example, in languages which use grammatical case, the subject is usually marked by the nominative, ergative or absolutive case; direct objects are usually marked by the accusative case, etc. The list of the currently used diagnostic parameters in Boas includes:

basic sentence structure parameters: word order preferences, grammatical functions (subject marking direct object marking, indirect object marking, complement marking, adverbial marking, verb marking), clause element agreement marking, clause boundary marking, and **basic noun phrase structure parameters:** POS patterns with head marking, phrase boundary marking, noun phrase component agreement

Restricting parameters determine the scope of usage of diagnostic parameters. Some of the diagnostic parameter values can only occur simultaneously with certain restricting parameter values. For example, in languages with the ergative construction the case of grammatical subject is restricted by the tense and aspect of the main verb (Mel’chuk 1998).

¹Remember: they are not supposed to be trained linguists but **are** expected to be able to translate between the source language and English.

²Such traditionally morphological parameters as part-of speech, number, gender, voice, aspect, etc. are elicited by the morphological module of Boas and are prerequisites for the syntax module.

2.3 The Elicitation Procedure

Prerequisites for syntax elicitation. Data that drives syntax elicitation is obtained at earlier stages of elicitation, namely **morphology** -- parameters such as Part of speech, Gender, Number, Person, Voice, Aspect, etc., as well as value sets for these parameters; **lexical acquisition of a small SL-English lexicon** to help work with the examples; the entries in the dictionary contain all the word forms and feature values of a SL lexeme and its English equivalent³, and **a very small corpus of carefully preselected and pretagged English noun phrases and sentences**, used as examples.

The inventory of tags and representation format. The tags for NPs include head and parameter values. The parameter (feature) set consists of Part of speech, Case, Number, Gender, Animacy and Definiteness (the values of the latter two may pose restrictions on agreement of NP components). Every NP is represented in the Boas knowledge base in the form of a typed feature structure as illustrated by the following example (the sign “#” marks the head):

```
["a good #boy"= [structure:noun-phrase]
  ["a"= [pos:determiner,
    number:singular, root:"a"]]
  ["good"= [pos:adjective,
    root:"good"]]
  ["boy"= [pos:noun, case:nominative4,
    number:singular, animacy:animate,
    root:"boy", head:1]]]
```

Two kinds of tags are used for sentence tagging—tags that refer to the whole sentence and tags for clause elements. Sentences are assigned values of such restricting parameters as

³We include in the prerequisite knowledge as much overtly listed linguistic information as possible, to avoid the necessity of automatic morphological analysis and generation which cannot guarantee absolutely correct results. This is possible due to a small size of the lexicon used for syntax examples.

⁴As we use a set of English NPs out of context, we believe that every phrase will be understood as being in the nominative case.

“clause type,” “voice,” “tense” and “aspect”. Clause elements are tagged with the value of the diagnostic parameter “syntactic function” and values of the restricting parameters “clause element realization,” “animacy” and “definiteness”. Clause elements also inherit sentence tags. Sentences are tagged in Boas as shown by the following example (the form of representation is a typed feature structure):

```
["the boy gives a book to his teacher"=
[structure:sentence, form:affirmative, clause-type:main, voice:active
tense:present, aspect:indefinite]
  ["the boy"= [function:subject,
    realization:noun-phrase,
    animacy:animate,
    definiteness:definite, head-root:"boy"]]
  ["gives"= [function:verb,
    realization:verb, head-root:"give"]]
  ["a book"= [function:direct-object,
    realization:noun-phrase,
    animacy:inanimate,
    definiteness:indefinite, head-root:"book"]]
  ["to his teacher"=
  [function:indirect-object,
    realization:prepositional-phrase,
    animacy:animate,
    definiteness:definite, head-root:"teacher"]]]]
```

Following the expectation-driven methodology the sets of pretagged noun phrases and sentences are selected to cover many though, admittedly, not all expected combinations of parameter values for every phrase or sentence. The following two examples further illustrate the Boas elicitation procedure.

Noun phrase pattern elicitation. The user is given a short definition of a noun phrase and asked to translate a given English phrase, for example “*a good boy*” into SL using the words given in a small lexicon of selected SL lexical items translated from English. In case of the Russian language the result would be: a good boy

---> horoshij malchik. Next, Boas automatically looks up every input SL word in the lexicon and assigns part of speech and feature value tags to all the components of SL noun phrases. English translations of SL words help record the comparative order of noun phrase pattern constituents in SL and English and automatically assigns the head marker to that element of the SL noun phrase which is the translation of the English head. This is the final result of SL noun phrase pattern elicitation for a given English phrase. It includes a SL noun phrase pattern to be used in an MT parser and a pattern transfer information for an English generator. Possible ambiguities, i.e., multiple sets of feature values for one word is resolved actively. The module can also actively check correctness of noun phrase translations.

Clause structure elicitation includes order of the words, subject markers (diagnostic feature values or particles), direct object markers, verb markers, and clause element agreement. Just like in the case of noun phrases, the user is asked to translate a given English phrase into SL using the words given in the lexicon. For the English sentence used in the example above the Russian translation will be:

```
the boy gives a book to his teacher ---
> malchik daet knigu uchitelju
```

As soon as this is done, Boas presents the user with English phrases corresponding to clause elements of the translated sentence, so that for every English-SL pair of sentences the user types in (or drags from the sentence translation) corresponding SL phrases, thus aligning clause elements. After the ractive alignment is done, the system automatically:

- transfers the clause element tags from English to SL⁵.
- marks the heads of every SL clause element, and
- assigns feature values to the heads of clause elements.

⁵This proved to be working in our experiment with 11 languages, such as French, Spanish, German, Russian, Ukrainian, Serbo-Croatian, Chinese, Persian, Turkish, Arabic, and Hindi.

- assigns sentence restricting parameter values (clause type, voice, tense and aspect, the last three are feature values of the verb).

In the case of assignment of multiple sets of feature values the user is asked to disambiguate them. As a result, every SL clause element is now tagged with certain values of diagnostic and restricting tags. The system stores these results as internal knowledge representation, in the form of a feature structure, for further processing. For example, for the above English-Russian sentence pair the mediate results (not shown to the user) will be:

```
[ "malchik daet knigu
uchitelju"= [ structure:sentence,
form:affirmative, clause-
type:main, voice:active,
tense:present,
aspect:imperfective]

[ "malchik"=
[function:subject, realization:noun-
phrase, animacy:animate,

head-1, root:"malchik",
case:nominative, number:singular,
gender:male, person:third]]

[ "daet"= [function:verb,
realization:verb, head-root:"davati"
,number:singular,
person:third]]

[ "knigu"= [function:direct-object,
realization:noun-phrase,
animacy:inanimate, head-root:"kniga",
case:accusative, number:singular,
gender:feminine, person:third]]

[ "uchitelju"= [function:indirect-
object, realization:noun phrase,
animacy:animate, head-root:"uchitel'",
case:dative, number:singular,
gender:male, person:third]]]
```

This data is further automatically processed to obtain the kind of knowledge which can be used in the parser or generator, that is, rules (not seen by the user), where the right-hand side contains a diagnostic parameter value (word order, clause element marking, agreement marking, etc.) and

the left-hand side contains the values of restricting parameters which condition the use of the corresponding diagnostic parameter value. A sample rule for the Russian example above is as follows:

```
DirectObjectMarker1= SL.Rule[
  lhs: SentenceForm [affirmative]
    ClauseType[main]
    Voice[active]
    Tense[present]
    Aspect[imperfective]
    Subject [realization:noun-phrase
animacy:animate]
    DirectObject[realization:noun-
phrase animacy:inanimate],
  rhs:<:SLDirectObjectMarker[case:accus
ative]:>];
```

These results are presented to the user for approval in a readable form. In Russian these rules mean the following:

in the affirmative sentence, main clause, active voice, present tense, when the subject is realized as NP and animate and direct object is realized as NP and inanimate,

- word order is SVO;
- subject is in nominative case;
- direct object is in accusative case;
- subject agrees with verb in number and person.

After all the sentence translations are processed in this way, the rules with the same right-hand side are automatically combined. At the next stage of processing the set of values for every restricting parameter in the right-hand side of the combined rule is checked on completeness. This means that in Russian in the affirmative main clause the preferred word order is SVO. The final results are presented for the user for approval or editing.

Conclusion

Boas is implemented as a WWW-based face, using HTML, Java Scripts and Perl. As of November 1999, the coverage of Boas includes the elicitation of inflectional morphology, morphotactics, open-class and closed-class

lexical items. Work on tokenization and proper names, syntax and feature and syntactic transfer is under way. Initial experiments have been completed on producing operational knowledge from the declarative knowledge elicited through Boas. Testing and evaluation of the system have been planned, and its results will be reported separately.

Acknowledgments

Research for this paper was supported in part by Contract MDA904-97-C-3976 from the US Department of Defense. Thanks to Jim Cowie and Rémi Zajac for many fruitful discussions of the issues related both to Boas proper and to the MT environment in which it operates.

References

- Brill, E., D Magerman, M Marcus and B Santorini. (1990) Deducing Linguistic Structure from the Statistics of Large Corpora. **Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics**. Berkeley, CA.
- Brown, P., J. Cocke, S. Della Pietra, V. Della Pietra, F. Jelinek, J.D. Lafferty, R.L. Mercer and P.S. Roossin. 1990. A statistical approach to machine translation. *Computational Linguistics*, 16: 79-85.
- Chomsky, N. 1981. *Lectures on Government and Binding*. Dordrecht: Foris.
- Chomsky, N. 1995. **The Minimalist Program**. Cambridge, MA: MIT Press.
- Goldsmith, J. 1998. Unsupervised Learning of the Morphology of a Natural Language. <http://humanities.uchicago.edu/faculty/goldsmith/Automorphology/Paper.doc>
- Jones, D. and R.Havrilla. 1998. Twisted Pair Grammar: Support for Rapid Development of Machine Translation for Low Density Languages. **AMTA'98**.
- Mel'cuk I. 1988. *Dependency Syntax: Theory and Practice*. State University of New York Press, Albany.
- Nirenburg, Sergei 1996. Supply-side and demand-side lexical semantics. Introduction to the Workshop on Breadth and Depth of Semantic Lexicons at **ACL'96**.
- Xia, Fei, M. Palmer, and K.Vijay-Shanker. 1999. Towards Semi-automatic Grammar Development. **Proceedings of the Natural Language Processing Pacific Rim Symposium**, Beijing, China.