

Towards A Universal Tool For NLP Resource Acquisition

Svetlana Sheremetyeva and Sergei Nirenburg

Computing Research Laboratory
New Mexico State University
Las Cruces, NM 88003 USA
{lana, sergei}@crl.nmsu.edu

Abstract

This paper describes an approach to developing a universal tool for eliciting, from a non-expert human user, knowledge about any language L. The purpose of this elicitation is rapid development of NLP systems. The approach is described 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 L into English. The preparation of knowledge for the MT system is carried out into two stages; the acquisition of descriptive knowledge about L and using the descriptive knowledge to derive operational knowledge for the system. Boas guides the acquisition process using data-driven, expectation-driven and goal-driven methodologies.

1. Introduction

Resource acquisition for NLP systems is a well-known bottleneck in language engineering. It is usually done or at least supervised by highly qualified specialists, requires a lot of time and effort and thus is very expensive. Resources are usually acquired tailored to a certain application for one, or, in case of multilingual applications two, rarely several languages. Methodologies of acquisition and scope of resources vary from application to application and are seldom reusable. It would be a clear advantage to have a tool based on the methodology which 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 who are not trained linguists.

The modules added through its operation include: **knowledge** about the ecology¹ of the source language (SL); its lexical stock; the inflectional and derivational morphology of SL and its morphotactics; the syntax of SL; and the correspondences (“transfer” knowledge) between the realizations of lexical, morphological and syntactic content in the SL and English.

2. 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). This is an attempt to develop a language learner for English linguists. The learner extracts distributional information (phrase structure) from a corpus annotated with parts of speech

(Tagged Brown Corpus) and is able to use this extracted information to accurately parse short sentences. Distributional analysis detects distributionally similar entities. For example one rule can be as follows: words are in the same class if they are licensed to occur in the same environments. There is an assumption that if two adjacent part of speech tags are distributionally similar to some single tag, then it is probable that the two tags form a constituent. It is decided, for example, that “Determiner Noun” is a constituent as it is distributionally similar to “Pronoun”, while “Noun Verb” is not. This system acquires a context-free grammar where each rule is assigned a score and needs a large tagged corpus for training.

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 & Havrilla, 1998) for a pair of English and a source language (SL). Acquisition is carried out by trained linguists. The goal of the work is to automatically assign clause syntactic patterns (phrase structure) to SL clauses on the basis of English-SL aligned sentences. The system finds equivalent words in English and parallel SL sentences and makes a phrase of two adjacent words in a SL clause if they correspond to any two adjacent words in a parallel English clause. Bracketing proceeds incrementally. Prerequisites for the acquisition of the twisted grammar should be a corpus of aligned clauses, a bilingual dictionary and disambiguation procedure. The latter is evidently the main obstacle to the success of the approach at this time. 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. The system is intended for linguists and is described on the example of English and Chinese. It relies on the assumption that languages are similar in many ways, and that most differences can be captured by the setting of certain parameters. An elicitation process is proposed that has the purpose of speeding up the development process based on language generalizations which are represented through

¹ Don Walker’s term relating to issues connected with writing systems, text mark-up, punctuation, special symbols, dates, numbers, proper names, etc.

default language-independent tree structures to be constructed manually. The goal of elicitation is to use to obtain language-dependent details from a native speaker. The language dependent details instantiate certain parameter settings, and thus generate blocks tailored to specific languages.

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. Even when the systems for automatically eliciting the various components of the knowledge required for MT attain acceptable levels of coverage and confidence, the task still remains to integrate their results and resolve the many mismatches that will be inherent in them. Indeed, there is seldom agreement even on such basic issues as the inventory of parts of speech in a language.

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, 1986, 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.

Attempts to implement the principles-and-parameters approach directly, see, for instance, (Dorr, 1993) do not seem to contribute much to NLP. They simply involve a relatively minor notation modification to accommodate the rules of syntactic transfer in MT. Just as its underlying theoretical approach, Dorr (1993) deemphasizes complete coverage of parametric information as a basis for syntactic analysis, introducing a few highly specific parameters and an incomplete set of values usable only for a subset of phenomena. While such attempts may have a special value for their proponents they are rather useless for Boas, which has developed a broad inventory of parameters and defined the complete sets of values for many of them.

The 1967-76 Stanford Project on Language Universals, spearheaded by Greenberg (1978) and continued by him and his associates since, focuses on substantive language universals rather than universals of grammatical descriptions within any specific theory. It provides useful information for the kind of parameter-value database that Boas needs. Unfortunately, the contributions to the Stanford Project focus on depth more than breadth, and yield fewer usable parameters and values, and especially value sets for specific languages, vital for Boas, which has to collect such information on its own, see, for instance, (Nirenburg, 1998; Nirenburg & Raskin, 1998).

In some other approaches, parameters and parameter values are either not sought out or are expected to be obtained automatically. The former option has been tested in the early versions of the Candide MT system (Brown et al., 1990) and found lacking. The latter capability has not yet passed the stage of initial exploratory experimentation on rather narrowly defined topics, e.g., (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.

3. Acquisition of Syntax in Boas

3.1. Methodologies for Selection of Syntax Parameters

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.

As Boas is supposed to support a practical application, practical constraints are added to the equation, most importantly, time constraints. The natural inclination of elicitation system designers to make elicitation more accurate and fine-grained must be held in check by the realization that detailed descriptions take more time than coarser-grained ones and by the fact that the work of the acquirer becomes more difficult as the sets of parameters, values, and realization options grow. The user becomes exposed to a number of questions which are not relevant to the particular source language.

One possibility to minimize the parameter acquisition effort is to use the knowledge about universals of word order. Since the work of Greenberg (1966), it has been clear that languages tend to group into certain classes with regard to word order. For instance, OV languages (in which the object precedes the verb, such as Japanese or Turkish) will be likely to have the head noun at the end of an NP, and to have postpositions rather than prepositions. It has been argued that there are two language types, head-initial and head-final, depending on the word order properties (Smith, 1978). But though these are definite statistical trends, they are not universals in a true sense, for there are exceptions (e.g. Persian is OV yet has prepositions).

Thus, even knowing to what language family a particular source language might belong does not always help, as related languages are often quite different typologically. This means that parameter acquisition for a non-toy MT system cannot rely on inferences based on linguistic universals and should explicitly elicit every parameter it needs.

Three basic methodological approaches are used in Boas:

- expectation-driven,
- goal-driven and
- data-driven.

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: French, Spanish, German, Russian, Ukrainian, Polish, Czech, Bulgarian, Serbo-Croatian, Albanian, Mokilese, Tagalog, Irish, Welsh, Finnish, Swedish, Swahili, Chinese, Japanese, Persian, Turkish,

Arabic, Hebrew and Basque. A representative list of parameters has already been acquired.

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.

The parser used is a Heuristic Clause Chunker which replaces the complex system of phrase structure rules in a traditional grammar. Such grammar is costly to acquire is not robust and is highly ambiguous. In contrast, the heuristic clause chunker is acquired quickly and cheaply, is robust to the extreme, and produces unambiguous results. The clause chunker is divided into two parts. The first part is responsible for postulating all reasonable candidate clause partitions for the input sentence. The second part uses heuristics to score each partition and pick the one with best overall score. Language specific information is used for both choosing candidate clause partitions and scoring them, among them word order (SVO vs. SOV), clause element (subject, object, etc.) marking, agreement marking, noun phrase structure pattern, position of a head.

Consider in some detail how information on constituent head marking can be useful for this purpose. It has been shown (Mel’cuk, 1964b) that in some languages the morphological properties of a constituent head depend on those of its dependents as, for example, in the Hungarian *professzor #konyv+e* (*professor’s #book*; # marks the head of the phrase) or in the Persian *#divar+e safid* (*white #wall*). Other languages choose the opposite strategy, for instance, Russian *#kniga professor+a* (*professor’s #book*) and French *#muraille blanche+e* (*white #wall*). Nichols (1986) thoroughly investigates these two patterns and demonstrates that they have significant implications for linguistic typology.

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 users²; In Boas, data-driven acquisition is guided by the resident English knowledge sources. In particular, sample realizations of syntactic constructions in English are used to elicit equivalent constructions in the source language, which helps to develop knowledge about the latter’s syntax.

To optimize quality, Boas uses all these methods in combination. Irrespective of the particular method used, Boas always relies on and is designed to record its findings using an overt specification—at the system level and, therefore, often opaquely for the user—of the abstract linguistic parameters and parameter values in terms of which the information is classified and organized for use by the processing modules.

² Remember: they are not supposed to be trained linguists but **are** expected to be able to translate between the source language and English.

3.2. Types of Syntax Parameters in Boas

The parameters which are elicited through the syntax module of Boas include³ 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. Special clause element markers may be not only morphological feature values but also special words or particles like in Japanese.

The list of the currently used diagnostic parameters in Boas includes

1. **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
2. **basic noun phrase structure parameters**
 - POS patterns with head marking
 - phrase boundary marking
 - noun phrase component agreement

Boas expects that any source language will feature at least some of the linguistic parameters from the Boas parameter set, for each parameter attested in the source language, Boas elicits the set of the parameter’s values in the source language and the possible realization(s) of each value.

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 (examples are from (Mel’cuk, 1988)). Thus in Georgian the subject is in the nominative case for the present tense, in ergative for the aorist, and in dative for the perfect:

Is amb+ob+s rom... (He says that...)
he_sg_nom say_pres_3sg that

Man tkv+a rom... (He said that...)
he_sg_erg say_aor_3sg that
(*amb-* and *tkv-* are suppletive roots of the same verb *tkma* “to say”)

³ Such 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.

Mas u+tkv+am+s rom...(He has said that...)
he_sg_dat say_perf_3sg that

One can also mention the Turkish language where the subject of the main clause is in nominative while the subject of active subordinate clause is always in genitive. Which means that the subject diagnostic parameter with the value “nominative case” is accompanied by a restricting parameter “clause type” with the value “main” and the subject diagnostic parameter with the value “genitive case” is used together with the restricting parameters “clause type” and “voice” with the values “subordinate” and “active”, correspondingly. In Vietnamese, the order of the words depends upon the main verb tense. In Persian, subject-verb agreement depends on the animateness of the subject. We can say that the subject diagnostic parameter value “nominative case” in English applies only with the restricting parameter “subject realization” with the values “noun” or “pronoun” and do not apply when the latter parameter has the value “non-finite clause”. The parameter “sentence form” with the values “affirmative”, “yes-no-question”, “wh-question”, “negation” is also relevant for MT.

Currently Boas recognizes the following restricting parameters:

1. sentence form
2. clause element realization
3. animateness
4. definiteness
5. clause type
6. voice
7. tense
8. aspect

We can expect that both sets of parameters may intersect, for example, word order may act a restricting parameter for, say clause element marking or agreement.

3.3. 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 wordforms and feature values of a SL lexeme and its English equivalent⁴, 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,

⁴ We 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.

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 very good #boy" = [
  structure:noun-phrase]
["a" = [
  pos:determiner,
  number:singular,
  root:"a" ] ]
["very" = [
  pos:adverb,
  root:"very" ] ]
["good" = [
  pos:adjective,
  root:"good" ] ]
["boy" = [
  pos:noun,
  case:nominative5,
  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 “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,
```

⁵ As we use a set of English NPs out of context, we believe that every phrase will be understood as being in the nominative case.

```

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 very 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 very good boy ---> ochen 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. For our Russian example the result of this procedure will be:

a very good #boy ----> ochen horoshij #malchik

```

Det_Sg Adv Adj #N_Nom_Sg_Masc_Anim -->
Adv Adj_Nom_Sg_Masc #N_Nom_Sg_Masc_Anim

```

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 interactively. The module can also interactively check correctness of noun phrase translations, e.g. catch missing words in either noun phrase. For example, in Turkish, the translation of the English phrase “two books” is “iki tane kitab.” There will be no match in the bilingual lexicon for the Turkish word “tane” as lexicon acquisition is English-driven and “tane” will not be given as a translation of any English word. In such a case, Boas asks the user to confirm that this word has not been typed in by mistake, after which it keeps it in the NP pattern as a string:

two #books --- > iki tane #kitab

```

Cardinal_Numeral #N_Nom_plu_Inanimate --->
Cardinal_Numeral "tane"#N_Nom_sg_Inanimate

```

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. For our English-Russian example the output of this stage will be:

```

the boy = malchik
gives = daet
a book = knigu
to his teacher = uchitelju

```

After the interactive alignment is done, the system automatically

- transfers the clause element tags from English to SL⁶.
- marks the heads of every SL clause element, and
- assigns feature values to the heads of clause elements.
- 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 intermediate 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-root:"malchik",
    case:nominative,
    number:singular,
    gender:masculine,
    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",

```

⁶ This 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.

```

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:masculine,
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. Some of these rules for the Russian example above are as follows:

```

WordOrder1= 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:<:SLSubjectMarker[case:nominative]
:>];

SubjectMarker1= 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:<:SLSubjectMarker[case:nominative]
:> ];

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:accusati
ve]:>];

Agreement1= 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:<:SLAgreement[clause-elements
[subject verb]
parameters [number person]]:>];

```

These results are presented to the user for approval in a readable form. In Russian these rules mean the following (italics are used to mark restricting parameters):

:
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. For example, a combined word order rule for Russian looks as follows:

```

WordOrder1= SL.Rule[
  lhs: SentenceForm[affirmative]
    ClauseType[main]
    Voice[active passive]
    Tense[present past future]
    Aspect[perfective imperfective]
    Subject[realization:noun-phrase
pronoun non-finite]
    animacy[animate inanimate]]
  DirectObject[realization:noun-
phrase pronoun non-finite]
    animacy[animate inanimate]],
  rhs:<:SLWordOrder[SVO]:>];

```

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. The parameter with a complete set of values for a given SL is deleted. After this check our example rule is reduced to the following:

```

WordOrder1= SL.Rule[
  lhs: SentenceForm[affirmative]
  ClauseType[main],
  rhs: <:SLWordOrder[SVO] :>];

```

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.

4. Conclusions

Boas is implemented as a WWW-based interface, 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 ecology, syntax and feature and syntactic transfer is under way (lexical transfer is a part of the treatment of open-class lexical items). Initial experiments have been completed on producing operational knowledge from the declarative knowledge elicited through Boas.

The methodology used in our research is related to that used in field linguistics, as it is also devoted to eliciting, from a speaker, knowledge about a language and using it to build grammars and lexicons, see, for instance,

(Samarin, 1967; Comrie & Smith, 1977; Bouquiaux & Thomas, 1992; Payne, 1997). However, in our approach, the grammars and the lexicons are machine tractable, that is, ready to be used as static knowledge sources in NLP systems. The kinds and amount of knowledge elicited from the speaker are determined by the requirements of the NLP applications in question. While the process of elicitation in field linguistics is entirely manual, the acquisition in descriptive computational linguistics is a mixture of manual, semi-automatic and automatic approaches.

Over the years, trained computational linguists have been acquiring knowledge for NLP applications, sometimes using increasingly sophisticated and automated tools. With Boas, the methodological initiative rests with the system: it is the system that must lead the acquirer, ordering the interactions (questions) and tracking the coverage needs and the nature of the output. While the acquirers will still have access to printed (or on-line) descriptive grammars, dictionaries and other reference materials, the responsibility for quality and coverage of the output now rests with Boas.

5. 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, Karine Megerdoomian and Rémi Zajac for many fruitful discussions of the issues related both to Boas proper and to the MT environment in which it operates.

6. References

- Bouquiaux, L. and J.M.C.Thomas. (1992). *Studying and Describing Unwritten Languages*. Dallas, TX: Summer Institute of Linguistics Press.
- 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. (1986). *Knowledge of Language: Its Nature, Origin, and Use*. New York: Praeger.
- Chomsky, N. (1995). *The Minimalist Program*. Cambridge, MA: MIT Press.
- Comrie, B., and N. Smith (1977). *Lingua Descriptive Studies: Questionnaire*. *Lingua* 42:1, pp. 1-72.
- Dorr, B. (1993). Interlingual Machine Translation: A Parameterized Approach. *Artificial Intelligence* 63, pp. 429-492.
- Goldstein, J. (1998). Unsupervised Learning of the Morphology of a Natural Language. <http://humanities.uchicago.edu/faculty/goldsmith/Automorphology/Paper.doc>
- Greenberg, J. H. (1966). Some Universals of Grammar with Particular Reference to the Order of Meaningful Elements. In *Universals of Language*, J. Greenberg (ed), Cambridge, Mass.: MIT Press.
- Greenberg, J. H. (ed.) (1978). *Universals of Human Language*, Vols. I-IV. Stanford, CA: Stanford University Press.
- Jones, D. and R.Havrilla. (1998). Twisted Pair Grammar: Support for Rapid Development of Machine Translation for Low Density Languages. AMTA'98.
- Knight, K. (1996). Learning Word Meanings by instruction. AAI '96.
- Knight, K., I. Chander, M. Haines, V. Hatzivassiloglou, E. Hovy, M. Iida, S. K. Luk, R. Whitney, and K. Yamada (1995). Filling Knowledge Gaps in a Broad-Coverage Machine Translation System. IJCAI '95.
- Mel'cuk I. (1964). Tipy svjazej mezdu elementami teksta i tipologia jazykov (Types of Links Between the Elements of a Text and Linguistic Typology). In *Materialy konferencii "Aktualnye voprosy sovremennogo jazykoznanija i lingvističeskoe nasledie E.D. Polivanova"*, 1, Samarkand: Samarkand University.
- Mel'cuk I. (1988). *Dependency Syntax: Theory and Practice*. State University of New York Press, Albany.
- Nichols J. (1986). Head-marking and Dependent-marking Grammar. In *Language*, 62:1.
- Nirenburg, Sergei (1996). Supply-side and demand-side lexical semantics. Introduction to the Workshop on Breadth and Depth of Semantic Lexicons at ACL'96.
- Nirenburg, S. (1998). Project Boas: "A Linguist in the Box" as a Multi-Purpose Language Resource. Proceedings of the First Lexical Resources and Evaluation Conference. Granada, Spain.
- Nirenburg, S., and V. Raskin (1998). Universal Grammar and Lexis for Quick Ramp-Up of MT Systems. *Coling-ACL '98*.
- 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.
- Payne, T. E. (1997). *Describing Morphosyntax. A Guide for Field Linguists*. Cambridge: Cambridge University Press.

Samarin, W. J. (1967). *Field Linguistics*. New York: Holt, Rinehart and Winston.