Minimal Dependency Grammar for Machine Translation

Abstract

This paper introduces Hiiktuu, an ongoing project to develop a framework and a set of tools for the creation of rudimentary bilingual lexicon-grammars for machine translation into and out of under-resourced languages. The basic units in Hiiktuu, called **groups**, are headed multi-item sequences. In addition to wordforms, groups may contain lexemes, syntactic-semantic categories, and grammatical features. Each group is associated with one or more translations, each of which is a group in a target language. During translation, constraint satisfaction is used to select a set of source-language groups for the input sentence and to sequence the words in the associated target-language groups.

1 Introduction

For the majority of the world's languages we lack adequate resources to make use of the machine learning techniques that have become the standard for modern computational linguistics. Languages with inadequate resources include not only those with few speakers, many of them endangered, but also a number of Asian and African languages with tens of millions of speakers, such as Telugu, Burmese, Oromo, and Hausa. For machine translation (MT) and computer-assisted translation (CAT), the lack is even more serious because what is required for machine learning is sentence-aligned translations.

For these reasons, work on many such languages will continue to consist in large part in the writing of computational grammars and lexica by people. Because such work normally requires significant training and is notoriously time-consuming, there is a need for tools to permit researchers and language technology users to "get off the ground" with these languages, that is, to create rudimentary grammars and lexica that will support basic applications and facilitate the language documentation process.

We focus on MT and CAT because a lack of linguistic resources correlates with a lack of written material, and we would like to develop tools to aid human translators in translating documents into these languages. Our long-term goal is a system that allows users with little or no linguistic experience to write bilingual lexicon-grammars for low-resource languages that can also be updated on the basis of corpora, when these are available, and that can be easily integrated into a CAT system.

In this paper we describe the initial steps in developing Hiiktuu,[1] a lexical-grammatical framework for MT and CAT. Although our focus is on the language pairs Spanish-Guarani and Amharic-Oromo, we illustrate Hiiktuu with examples from English-Spanish.

2 Lexica and grammars

2.1 Phrasal lexica

The idea of treating phrases rather than individual words as the basic units of a language goes back at least to the proposal of a Phrasal Lexicon by Becker (1975). In recent years, the idea has gained currency within the related frameworks of Construction Grammar (Steels, 2011) and Frame Semantics (Fillmore

[1] *Hiiktuu* is the Oromo word for a (female) translator.

and Baker, 2001) as well as in phrase-based statistical machine translation (PBSMT). Arguments in favor of phrasal units are often framed in terms of the ubiquity of idiomaticity, that is, departure from strict compositionality. Seen another way, phrasal units address the ubiquity of lexical ambiguity. If a verb's interpretation depends on its object or subject, then it may make more sense to treat the combination of the verb and particular objects or subjects as units in their own right.

Arguments based on idiomaticity and ambiguity are semantic, but they extend naturally to translation. If the meaning of a source-language phrase fails to be the strict combination of the meanings of the words in the phrase, then it is unlikely that the translation of the phrase will be the combination of the translations of the source-language words. Adding lexical context to an ambiguous noun or verb can sometimes permit an MT system to select the appropriate translation.

2.2 A simple phrasal lexicon

The basic lexical entries of Hiiktuu are multi-word units called **groups**. Each group represents a catena (Osborne et al. , 2012). Catenae go beyond constituents (phrases), including all combinations of elements that are continuous in the vertical dimension within a dependency tree. For example, in the sentence *I gave her a piece of my mind*, {*I, gave*} and {*gave, her, piece*} are among the catenae but not the constituents of the sentence.

A catena has a head, and each Hiiktuu group must also have a head, which indexes the group within the lexicon. A group's entry also specifies translations to groups in one or more other languages. For each translation, the group's entry gives an **alignment**, representing inter-group correspondences between elements, as in the phrase tables of PBSMT. Entry 1 shows a simple group entry of this sort.[2] The English group *the end of the world* with head *end* has as its Spanish translation the group *el fin del mundo* (which has its own entry in the Spanish lexicon). In the alignment, each word other than the fourth word (*the*) in the English group is associated with the position of a word in the Spanish group.

Entry 1 Group entry for *the end of the world* and its Spanish translation

```
end:
  - words: [the, end, of, the, world]
    spa:
    - [el_fin_del_mundo, {align: [1,2,3,0,4]}]
```

2.3 The lexicon-grammar tradeoff

A rudimentary lexicon with entries of this sort is simple in two senses: a user with no formal knowledge of linguistics can add entries in a straightforward manner, and the resulting entries are easily understood. Such a lexicon permits the translation of sentences that are combinations of the wordforms in the group entries, as long as group order is preserved across the languages and there are no constraints between groups that would affect the form of the target-language words. However, such a lexicon permits no *generalization* to combinations of wordforms that are not explicit in the lexicon. It would require a group entry for every reasonably possible combination of wordforms.

At the other extreme from this simple lexicon is a full-blown grammar that is driven by the traditional linguistic concern with parsimony: every possible generalization must be "captured". Although it has the advantage of compactness and of reflecting general principles of linguistic structure, such a grammar is difficult to write, to debug, and to understand, requiring significant knowledge of linguistics.

In the Hiiktuu project, the goal is a range of possibilities along the continuum from purely lexical (and phrasal) to syntactic/grammatical, with the emphasis on ease of entry creation and interpretation.

2.4 Lexemes

We can achieve significant generalization over simple groups consisting of wordforms by permitting lexemes in groups. As an example, consider the English group *passV the buck*, where *passV* is the verb lexeme *pass*. In order to make such a group usable, the lexicon also requires **form** entries, giving the

[2] We serialize Hiiktuu lexical with YAML (`http://www.yaml.org/`)

lexeme roots as well as grammatical features for specific wordforms. Some of these, along with the group entry, are shown in Entry 2.

Entry 2 Group entry for *pass the buck* and three form entries

```
groups:
  passV:
  - words: [passV, the, buck]
    spa:
      - [escurrir_el_bulto,
         {align: [1,2,3], agr: [{tns: tmp, prs: prs, num: num}, 0, 0]}]
forms:
  pass:
  - root: passV, features: {prs: 1, tns: prs}
  - root: passV, features: {prs: 3, num: plr, tns: prs}
  - root: passN, features: {num: sng}
  passes:
    root: passV, features: {prs: 3, num: sng, tns: prs}
  passed:
    root: passV, features: {tns: pst}
```

Because this entry accommodates multiple sequences of English wordforms, we need to map these onto appropriate target-language sequences. This is accomplished through pairs of agreement features for the lexeme, constraining the corresponding target language form to agree with the source form on those features. In the example, the head *passV* and its translation in the Spanish group agree on tense and *tiempo*, person and *persona*, and number and *número* features. For example, if this group is selected in the translation of the sentence *Carl passes the buck*, the head of the corresponding Spanish group will be constrained to be third person singular present tense (*tiempo*): *Carl* ***escurre*** *el bulto*.

2.5 Lexical/grammatical categories

Another simple way to generalize across groups is to introduce syntactic or semantic categories. Consider the English expression *give somebody a piece of one's mind*. We can generalize across specific word sequences such as *gave me a piece of his mind* and *gave them a piece of my mind* by replacing the specific wordforms in positions 2 and 6 in the group with categories that include the wordforms that can fill those positions. This requires the forms dictionary to record the categories that wordforms belong to. Entry 3 shows how this appears in the lexicon. Category names are preceded by $.

Entry 3 Three group entries and two associated form entries

```
groups:
  giveV:
  - words: [giveV, $sbd, a, piece, of, $sbds, mind]
    agr: [[1, 6, {prs: prs, num: num}]]
  my:
  - words: [my]
  mayor:
  - words: [the, mayor]
forms:
  my: [{cats: [$sbds]}]
  mayor: [{cats: [$sbd]}]
```

Because group positions that are filled by categories do not specify a surface form, during parsing and generation of sentences they must be merged with other groups that match the category and do specify a form. For example, to parse or translate the sentence *I gave the mayor a piece of my mind* requires that positions 2 and 6 in the group *giveV_sbd_a_piece_of_sbds_mind* be filled by the heads of the groups *the_mayor* and *my*. This **node merging** process is illustrated in Figure 1.

This group illustrates another requirement of some groups containing categories. In *give somebody a piece of one's mind*, the possessive adjective in the place of *one's* must agree with the subject of the sentence. Since the group contains no subject, we constrain it to agree with the person and number of the verb. Thus the entry for this group also contains an agreement attribute specifying that the the sixth element must agree with the first on person and number features.

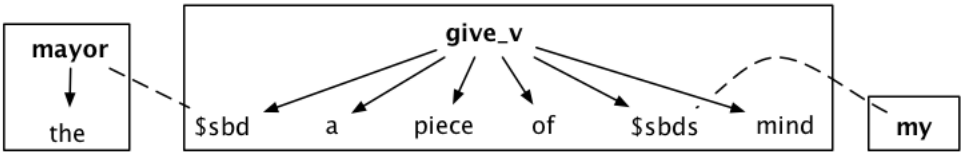

Figure 1: Merging of three groups in *gave the mayor a piece of my mind*

3 Constraint satisfaction and translation

Translation in Hiiktuu takes place in three phases: analysis, transfer, and realization. Analysis of the source-language sentence begins with a lexical lookup of the wordforms in the forms dictionary for the source language.[3] The words or lexemes resulting from this first pass are then used to look up candidate groups in the groups dictionary. Next the system assigns a set of groups to the input sentence. A successful group assignment satisfies several constraints: (1) each word in the input sentence is assigned to zero, one, or (in the case of node merging) two group elements. (2) Each element in a selected group is assigned to one word in the sentence. (3) For each selected group, within-group agreement restrictions are satisfied. (4) For each category element in a selected group, it is merged with a non-category element in another selected group (see the two examples in Figure 1). Analysis is a robust process; some words in the input sentence may end up unassigned to any group.

Analysis is implemented in the form of constraint satisfaction, making use of insights from the Extensive Dependency Grammar framework (XDG) (Debusmann, 2007). Although considerable source-sentence ambiguity is eliminated because groups incorporate context, ambiguity is still possible, particularly for figurative expressions that also have a literal interpretation. In this case, the constraint satisfaction process undertakes a search through the space of possible group assignments, creating an analysis for each successful assignment.

During the transfer phase, a source-language group assignment is converted to an assignment of target-language groups. In this process some target-language items are assigned grammatical features on the basis of agreement constraints. For example, in the translation of the English sentence *the mayor passes the buck*, the Spanish verb that is the head of the group *escurrir_el_bulto* would be assigned the tense (*tiempo*), person and number features `tmp=prs, prs=3, num=1`: *escurre*. A source-language group may have more than one translation. The transfer phase creates a separate target-language group assignment for each combination of translations of the source-language groups.

During the realization phase, for each target-language group assignment, surface forms are generated based on the lexemes and grammatical features that resulted from the transfer phase. In the current version of the system, this is accomplished through a dictionary that maps lexemes and feature sets to surface forms.[4] Finally, target-language words are sequenced in a way that satisfies word-order conditions in target-language groups. The sequencing process is implemented with constraint satisfaction.

4 Related work

Our goals are similar to those of the Apertium (Forcada et al. , 2011) project. As with Apertium, we are developing open-source, rule-based systems for MT, and we work within the framework of relatively shallow, chunking grammars. We differ mainly in our willingness to sacrifice linguistic coverage to achieve our goals of flexibility, robustness, and transparency. We accommodate a range of lexical-grammatical possibilities, from the completely lexical on the one extreme to phrasal units consisting of a single lexeme and one or more syntactic/semantic categories on the other, and we are not so concerned that Hiiktuu grammars will accept many ungrammatical source-language sentences or even output ungrammatical (along with grammatical) translations.

In terms of long-term goals, Hiiktuu also resembles the Expedition project (McShane et al. , 2002), which makes use of knowledge acquisition techniques and naive monolingual informants to develop

[3] In future versions of the system, it will be possible to call a morphological analyzer on the input forms at this stage.

[4] In future versions of the system, it will be possible to call a morphological generator at this stage.

MT systems that translate low-resource languages into English. Our project differs first, in assuming bilingual informants and second, in aiming to develop systems that are unrestricted with respect to target language. In fact we are more interested in MT systems with low-resource languages as target languages because of the lack of documents in such languages.

Although Hiiktuu is not intended as a linguistic theory, it is worth mentioning which theories it has the most in common with. Like Construction Grammar (Steels, 2011) and Frame Semantics (Fillmore and Baker, 2001), it treats linguistic knowledge as essentially phrasal. Like synchronous context-free grammar (SCFG) (Chiang, 2007), it associates multi-word units in two languages, aligning the elements of the units and representing word order within each. Hiiktuu differs from SCFG in having nothing like rewrite rules or non-terminals. Hiiktuu belongs to the family of dependency grammar (DG) theories because the heads of its phrasal units are words or lexemes rather than non-terminals. It shares most with those computational DG theories that rely on constraint satisfaction (Bojar, 2005; Debusmann, 2007; Foth and Menzel, 2006; Wang and Harper, 2004). However, it remains an extremely primitive form of DG, permitting only flat structures with unlabeled arcs and no relations between groups other than through the merge operation described in 2.5. This means that complex grammatical phenomena such as long-distance dependencies and word-order variability can only be captured through specific groups.

5 Status of project, ongoing and future work

The code for Hiiktuu and a set of lexical-grammatical examples are available at [*URL omitted to preserve anonymity*] under the GPL license. To date, we have only tested the framework on a limited number of translations using various language pairs. In order to develop more complete lexicon-grammars for Amharic-Oromo and Spanish-Guarani, we are working on methods for automatically extracting groups from dictionaries in various formats and from the limited bilingual data that are available. As a part of this work, it will be crucial to determine whether it is simpler to extract Hiiktuu groups from data than to extract grammars of other sorts, for example, SCFG. We are also implementing a GUI that will allow naive bilingual users to create Hiiktuu entries. Again we will want to evaluate the framework with respect to the simplicity of entry creation. For the longer term, our goal is tools for the intelligent elicitation of lexical entries; for example, when two entries resemble one another, users could be queried about the value of collapsing them into a more abstract entry.

As far as the grammatical framework is concerned, the lack of dependencies between the heads of groups leaves the system without the capacity to represent some agreement constraints, for example, agreement between a verb+object group and the verb's subject, or major constituent order differences between source and target language.[5] To alleviate this problem, we will be implementing dependencies between group heads, much as in the "interchunk module" of Apertium.

6 Conclusions

Relatively sophisticated computational grammars, parsers, and/or generators exist for perhaps a dozen languages, and usable MT systems exist for at most dozens of pairs of languages. This leaves the great majority of languages and the communities who speak them relatively even more disadvantaged than they were before the digital revolution. What is called for are methods that can be quickly and easily deployed to begin to record the grammars and lexica of these languages and to use these tools for the benefit of the linguistic communities. The Hiiktuu project is designed with these needs in mind. Though far from achieving our ultimate goals, we have developed a simple, flexible, and robust framework for bilingual lexicon-grammars and MT/CAT that we hope will be a starting point for a large number of under-resourced languages.

[5]The only way to implement such constraints in the current version of Hiiktuu is through groups that incorporate, for example, subjects in verb-headed groups, as in *$sbd kickV $sth*.

References

Joseph Becker. 1975. The phrasal lexicon. In Roger Schank and Bonnie Nash-Webber, editors, *Theoretical Issues in Natural Language Processing*, pages 38–41. Association for Computational Linguistics.

Ondřej Bojar. 2005. Problems of inducing large coverage constraint-based dependency grammar. In H. Christiansen et al., editor, *Constraint Solving and Language Processing, First International Workshop, CSLP 2004*, pages 90–103, Berlin. Springer Verlag.

David Chiang. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33(2):201–228.

Ralph Debusmann. 2007. *Extensible Dependency Grammar: A Modular Grammar Formalism Based On Multigraph Description*. Ph.D. thesis, Universität des Saarlandes.

Charles J. Fillmore and Collin F. Baker. 2001. Frame semantics for text understanding. In *Proceedings of WordNet and Other Lexical Resources Workshop*, Pittsburgh, June. NAACL, NAACL.

Mikel L. Forcada, Mireia Ginestí-Rosell, Jacob Nordfalk, Jim O'Regan, Sergio Ortiz-Rojas, Juan Antonio Pérez-Ortiz, Felipe Sánchez-Martínez, Gema Ramírez-Sánchez, and Francis M. Tyers. 2011. Apertium: a free/open-source platform for rule-based machine translation. *Machine Translation*, 25(2):127–144.

Kilian Foth and Wolfgang Menzel. 2006. Hybrid parsing: Using probabilistic models as predictors for a symbolic parser. In *Proceedings of the Annual Conference of the Association for Computational Linguistics*, Sydney, Australia.

Marjorie McShane, Sergei Nirenburg, James Cowie, and Ron Zacharski. 2002. Embedding knowledge elicition and MT systems within a single architecture. *Machine Translation*, 17:271–305.

T. Osborne, M. Putnam, and T. Groß. 2012. Catenae: Introducing a novel unit of syntactic analysis. *Syntax*, 15(4):354–396.

Luc Steels, editor. 2011. *Design Patterns in Fluid Construction Grammar*. John Benjamins, Amsterdam.

Wen Wang and Mary Harper. 2004. A statistical constraint dependency grammar (CDG) parser. In *Proceedings of ACL04 Incremental Parsing Workshop*, Barcelona, Spain.