Urdu and the Parallel Grammar Project

Miriam Butt
Cent. for Computational Linguistics
UMIST
PO Box 88
Manchester M60 1QD GB
mutt@csli.stanford.edu

Tracy Holloway King
Palo Alto Research Center
3333 Coyote Hill Rd.
Palo Alto, CA 94304 USA
thking@parc.com

Abstract

We report on the role of the Urdu grammar in the Parallel Grammar (ParGram) project (Butt et al., 1999; Butt et al., 2002).[1] The ParGram project was designed to use a single grammar development platform and a unified methodology of grammar writing to develop large-scale grammars for typologically different languages. At the beginning of the project, three typologically similar European grammars were implemented. The addition of two Asian languages, Urdu and Japanese, has shown that the basic analysis decisions made for the European languages can be applied to typologically distinct languages. However, the Asian languages required the addition of a small number of new standard analyses to cover constructions and analysis techniques not found in the European languages. With these additional standards, the ParGram project can now be applied to other typologically distinct languages.

1 Introduction

In this paper, we report on the role of the Urdu grammar in the Parallel Grammar (ParGram) project (Butt et al., 1999; Butt et al., 2002). The ParGram project originally focused on three closely related European languages: English, French, and German. Once grammars for these languages were established, two Asian languages were added: Japanese and Urdu.[2] Both grammars have been successfully integrated into the project. Here we discuss the Urdu grammar and what special challenges it brought to the ParGram project. We are pleased to report that creating an Urdu grammar within the ParGram standards has been possible and has led to typologically useful extensions to the project.

The ParGram project uses the XLE parser and grammar development platform (Maxwell and Kaplan, 1993) to develop deep grammars for six languages. All of the grammars use the Lexical-Functional Grammar (LFG) formalism which produces c(onstituent)-structures (trees) and f(unctional)-structures (AVMs) as syntactic analyses.

LFG assumes a version of Chomsky's Universal Grammar hypothesis, namely that all languages are governed by similar underlying structures. Within LFG, f-structures encode a language universal level of analysis, allowing for cross-linguistic parallelism. The ParGram project aims to test the LFG formalism for its universality and coverage limitations and to see how far parallelism can be maintained across languages. Where possible, the analyses produced for similar constructions in each language are parallel. This parallelism requires a standard for linguistic analysis. In addition, the LFG theory itself limits the set of possible analyses, thus restricting the possible analyses to choose from. The standardization of the analyses has the computational advantage that the grammars can be used in similar applications, and it can simplify cross-language applications (Frank, 1999).

The conventions developed within the ParGram grammars are extensive. The ParGram project dictates not only the form of the features used in the grammars, but also the types of analyses that are chosen for constructions. In addition, the XLE platform necessarily restricts how the grammars can be written. In all cases, the Urdu grammar has successfully, and straightforwardly, incorporated the standards that were originally designed for the European languages. In addition, it has contributed to the formulation of new standards of analysis. Below we discuss several aspects of this: morphology, lexicon, and grammar development for the Urdu grammar within the ParGram project.

[1] We would like to thank Mary Dalrymple, Ron Kaplan, Hiroshi Masuichi, and Tomoko Ohkuma for their comments.

[2] Norwegian was also added at this time.

2 Morphology

The grammars in the ParGram project depend on finite-state morphologies as input (Beesley and Karttunen, 2002). Without this type of resource, it is difficult to build large-scale grammars, especially for languages with substantial morphology. For the original three languages, such morphologies were readily available. As they had been developed for information extraction applications instead of deep grammar applications, there were some minor problems, but the coverage of these morphologies is excellent. An efficient, broad-coverage morphology was also available for Japanese (Asahara and Matsumoto, 2000) and was integrated into the grammar. This has aided in the Japanese grammar rapidly achieving broad coverage. It has also helped control ambiguity because in the case of Japanese, the morphology determines the part of speech of each word in the string with very little ambiguity.

While some morphological analyzers already exist for Hindi,[3] e.g., as part of the tools developed at the Language Technologies Research Centre (LTRC), IIT Hyderabad (http://www.iiit.net/ltrc/index.html), they are not immediately compatible with the XLE grammar development platform, nor is it clear that the morphological analyses they produce conform to the standards and methods developed within the ParGram project. As such, part of the Urdu project is to build a finite-state morphology that will serve as a resource to the Urdu grammar and could be used in other applications.

The development of the Urdu morphology involves a two step process. The first step is to determine the morphological class of words and their subtypes in Urdu. Here we hope to use existing resources and lexicons. The morphological paradigms which yield the most efficient generalizations from an LFG perspective must be determined. Once the basic paradigms and morphological classes have been identified, the second step is to enter all words in the language with their class and subtype information. These steps are described below. Currently we are working on the first step; grant money is being sought for further development.

The finite-state morphologies used in the ParGram project associate surface forms of words with a canonical form (a lemma) and a series of morphological tags that provide grammatical information about that form. An example for English is shown in (1) and for Urdu in (2).

(1) pushes: push +Verb +Pres +3sg
push +Noun +Pl

(2) bOlA bOl +Verb +Perf +Masc +Sg

(1) states the English surface form *pushes* can either be the third singular form of the verb *push* or the plural of the noun *push*. (2) states that the Urdu surface form *bOlA* is the perfect masculine singular form of the verb *bOl*.

The first step of writing a finite-state morphology for Urdu involves determining which tags are associated with which surface forms. As can be seen from the above examples, determining the part of speech (e.g., verb, noun, adjective) is not enough for writing deep grammars. For verbs, tense, aspect, and agreement features are needed. For nouns, number and gender information is needed, as well as information as to whether it is a common or proper noun. Furthermore, for a number of problematic morphological phenomena such as oblique inflection on nominal forms or default agreement on verbs, the most efficient method of analyzing this part of the morphology-syntax interface must be found (Butt and Kaplan, 2002).

After having determined the tag ontology, the patterns of how the surface forms map to the stem-tag sets must be determined. For example, in English the stem-tag set *dog +Noun +Pl* corresponds to the surface form *dogs* in which an *s* is added to the stem, while *box +Noun +Pl* corresponds to *boxes* in which an *es* is added. At this point in time, the basic tag set for Urdu has been established. However, the morphological paradigms that correspond to these tag combinations have not been fully explored.

Once the basic patterns are determined, the second stage of the process begins. This stage involves greatly increasing the coverage of the morphology by adding in all the stems in Urdu and marking them for which set of tags and surface forms they appear with. This is a very large task. However, by using frequency lists for the language and existing lexicons,[4] the most common words can be added first to obtain a major gain in coverage.

In addition, a guesser can be added to guess words that the morphology does not yet recognize (Chanod

[3] An on-line morphological analyzer is available at: http://ccat.sas.upenn.edu/plc/tamilweb/hindi.html

[4] A web search on `Hindi dictionary` results in several promising sites.

and Tapanainen, 1995). This guessing is based on the morphological form of the surface form. For example, if a form ending in *A* is encountered and not recognized, it could be considered a perfect masculine singular form, similar to *bOlA* in (2).

3 Lexicon

One advantage of the fact that the XLE system incorporates large finite-state morphologies is that the lexicons for the languages can then be relatively small. This is because lexicons are not needed for words whose syntactic lexical entry can be determined based on their morphological analysis. This is particularly true for nouns, adjectives, and adverbs.

Consider the case of nouns. The Urdu morphology provides the following analysis for the proper noun *nAdyA*.

(3) nAdyA +Noun +Name +Fem

The tags provide the information that it is a noun, in particular a type of proper noun (Name), and is feminine. The lexical entries for the tags can then provide the grammar with all of the features that it needs to construct the analysis of *nAdyA*; this resulting f-structure analysis is seen in Figures 2 and 4. Thus, *nAdyA* itself need not be in the lexicon of the grammar because it is already known to the morphological analyzer.

Items whose lexical entry cannot be predicted based on the morphological tags need explicit lexical entries. This is the case for items whose subcategorization frames are not predictable, primarily for verbs. Currently, the Urdu verb lexicon is hand constructed and only contains a few verbs, generally one for each subcategorization frame for use in grammar testing. To build a broad-coverage Urdu grammar, a more complete verb lexicon will be needed. To provide some idea of scale, the current English verb lexicon contains entries for 9,652 verbs; each of these has an average of 2.4 subcategorization frames; as such, there are 23,560 verb-subcategorization frame pairs. However, given that Urdu employs productive syntactic complex predicate formation for much of its verbal predication, the verb lexicon for Urdu will be smaller than its English counterpart. On the other hand, writing grammar rules for the productive combinatorial possibilities between adjectives and verbs (e.g., *sAf karnA* 'clean do'='clean'), nouns and verbs (e.g., *yAd karnA* 'memory do'='remember') and verbs and verbs (e.g., *kHA lEnA* 'eat take'='eat up') is anticipated to require significant effort.

There are a number of ways to obtain a broad-coverage verb lexicon. One is to extract the information from an electronic dictionary. This does not exist for Urdu, as far as we are aware. Another is to extract it from Urdu corpora. Again, these would have to be either collected or created as part of the grammar development project. A final way is to enter the information by hand, depending on native speaker knowledge and print dictionaries; this option is very labor intensive. Fortunately, work is being done on verb subcategorization frames in Hindi.[5] We plan to incorporate this information into the Urdu grammar verb lexicon.

4 Grammar

The current Urdu grammar is relatively small, comprising 25 rules (left-hand side categories) which compile into a collection of finite-state machines with 106 states and 169 arcs. The size of the other grammars in the ParGram project are shown in (4) for comparison.

(4)

Language	Rules	States	Arcs
German	444	4883	15870
English	310	4935	13268
French	132	1116	2674
Japanese	50	333	1193
Norwegian	46	255	798
Urdu	25	106	169

It is our intent to drastically expand the Urdu grammar to provide broad-coverage on standard (grammatical, written) texts. The current size of the Urdu grammar is not a reflection of the difficulty of the language, but rather of the time put into it. Like the Japanese and Norwegian grammars, it is less than two years in development, compared with seven years[6] for the English, French, and German grammars. However, unlike the Japanese and Norwegian grammars, there has been no full-time grammar writer on the Urdu grammar. Below we discuss the Urdu grammar analyses and how they fit into the ParGram project standardization requirements.

Even within a linguistic formalism, LFG for ParGram, there is often more than one way to ana-

[5] One significant effort is the Hindi Verb Project run by Prof. Alice Davison at the University of Iowa; further information is available via their web site.

[6] Much of the effort in the initial years went into developing the XLE platform and the ParGram standards. Due to these initial efforts, new grammars can be developed more quickly.

lyze a construction. Moreover, the same theoretical analysis may have different possible implementations in XLE. These solutions often differ in efficiency or conceptual simplicity. Whenever possible, the ParGram grammars choose the same analysis and the same technical solution for equivalent constructions. This was done, for example, with imperatives. Imperatives are assigned a null pronominal subject within the f-structure and a feature indicating that they are imperatives.

Parallelism, however, is not maintained at the cost of misrepresenting the language. Situations arise in which what seems to be the same construction in different languages cannot have the same analysis. An example of this is predicate adjectives (e.g., *It is red.*). In English, the copular verb is considered the syntactic head of the clause, with the pronoun being the subject and the predicate adjective being an XCOMP. However, in Japanese, the adjective is the main predicate, with the pronoun being the subject. As such, these constructions receive non-parallel analyses.

Urdu contains several syntactic constructions which find no direct correlate in the European languages of the ParGram project. Examples are correlative clauses (these are an old Indo-European feature which most modern European languages have lost), extensive use of complex predication, and rampant pro-drop. The ability to drop arguments is not correlated with agreement or case features in Urdu, as has been postulated for Italian, for example. Rather, pro-drop in Urdu correlates with discourse strategies: continuing topics and known background information tend to be dropped. Although the grammars do not encode discourse information, the Japanese grammar analyzes pro-drop effectively via technical tools made available by the grammar development platform XLE. The Urdu grammar therefore anticipates no problems with pro-drop phenomena.

In addition, many constructions which are stalwarts of English syntax do not exist in Asian languages. Raising constructions with *seem*, for example, find no clear correlate in Urdu: the construction is translated via a psych verb in combination with a *that*-clause. This type of non-correspondence between European and South Asian languages raises challenges of how to determine parallelism across analyses. A similar example is the use of expletives (e.g., *There is a unicorn in the garden.*) which do not exist in Urdu.

4.1 Existing Analysis Standards

While Urdu contains syntactic constructions which are not mirrored in the European languages, it shares many basic constructions, such as sentential complementation, control constructions, adjective-noun agreement, genitive specifiers, etc. The basic analysis of these constructions was determined in the initial stage of the ParGram project in writing the English, French, and German grammars. These analysis decisions have not been radically changed with the addition of two typologically distinct Asian languages, Urdu and Japanese.

The parallelism in the ParGram project is primarily across the f-structure analyses which encode predicate-argument structure and other features that are relevant to syntactic analysis, such as tense and number.[7] A sample analysis for the sentence in (5) is shown in Figures 1 and 2.

(5) nAdyA kA kuttA AyA
Nadya Gen.M.Sg dog.Nom come-Perf.M.Sg
'Nadya's dog came.'

The Urdu f-structure analysis of (5) is similar to that of its English equivalent. Both have a PRED for the verb which takes a SUBJ argument at the top level f-structure. This top level structure also has TNS-ASP features encoding tense and aspect information, as well as information about the type of sentence (STMT-TYPE) and verb (VTYPE); these same features are found in the English structure. The analysis of the subject is also the same, with the possessive being in the SPEC POSS and with features such as NTYPE, NUM, and PERS. The sentence in (5) involves an intransitive verb and a noun phrase with a possessive; these are both basic constructions whose analysis was determined before the Urdu grammar was written. Yet, despite the extensive differences between Urdu and the European languages—indeed, the agreement relations between the genitive and the head noun are complex in Urdu but not in English—there was no problem using the standard analysis for the Urdu construction.

4.2 New Analysis Standards

Analyses of new constructions have been added for constructions found in the new project languages.

[7] The c-structures are less parallel in that the languages differ significantly in their word orders. Japanese and Urdu are SOV while English is SVO. However, the standards for naming the nodes in the trees and the types of constituents formed in the trees, such as NPs, are similar.

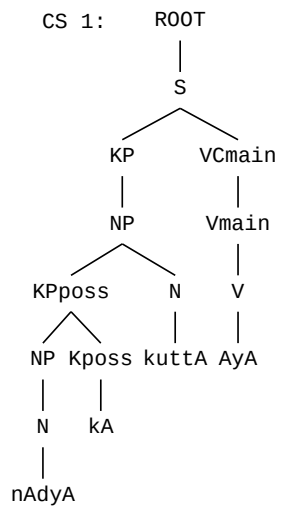


Figure 1: C-structure tree for (5)

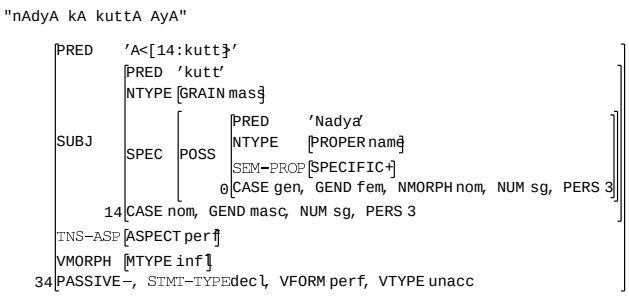


Figure 2: F-structure AVM for (5)

These analyses have not only established new standards within the ParGram project, but have also guided the development of the XLE grammar development platform. Consider the analysis of case in Urdu. Although the features used in the analysis of case were sufficient for Urdu, there was a problem with implementing it. In Urdu, the case markers constrain the environments in which they occur (Butt and King, to appear). For example, the ergative marker *ne* only occurs on subjects. However, not all subjects are ergative. To the contrary, subjects can occur in the ergative, nominative, dative, genitive, and instrumental cases. Similarly, direct objects can be marked with (at least) an accusative or nominative, depending on the semantics of the clause. Minimal pairs such as in (6) for subjects and (7) for objects suggest a *constructive* (Nordlinger, 1998) approach to case.

(6) a. rAm kHÃs-A
Ram.Nom cough-Perf.M.Sg
'Ram coughed.'

b. rAm nE kHÃs-A
Ram=Erg cough-Perf.M.Sg
'Ram coughed (purposefully).'

(7) a. nAdyA nE gArI calAyI
Nadya=Erg car.Nom drive-Perf.F.Sg
hai
be.Pres.3.Sg
'Nadya has driven a car.'

b. nAdyA nE gArI kO calAyA
Nadya=Erg car=Acc drive-Perf.M.Sg
hai
be.Pres.3.Sg
'Nadya has driven the car.'

We therefore designed the lexical entries for the case markers so that they specify information about what grammatical relations they attach to and what semantic information is needed in the clausal analysis. The lexical entry for the ergative case, for example, states that it applies to a subject.

These statements require inside-out functional uncertainty (Kaplan, 1988) which had not been used in the other grammars. Inside-out functional uncertainty allows statements about the f-structure that contains an item. The lexical entry for *nE* is shown in (8).

(8) nE K @(CASE erg) `line 1`
(SUBJ ($) ^) `line 2`
@VOLITION `line 3`

In (8), the K refers to the part of speech (a case clitic). Line 1 calls a template that assigns the CASE feature the value erg; this is how case is done in the other languages. Line 2 provides the inside-out functional uncertainty statement; it states that the f-structure of the ergative noun phrase, referred to as ^, is inside a SUBJ. Finally, line 3 calls a template that assigns the volitionality features associated with ergative noun phrases. The analysis for (9) is shown in Figures 3 and 4.

(9) nAdyA nE yassin ko mArA
Nadya=Erg Yassin=Acc hit-Perf.M.Sg
'Nadya hit Yassin.'

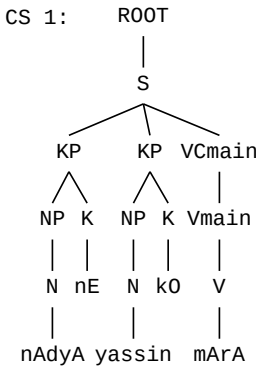


Figure 3: C-structure tree for (9)

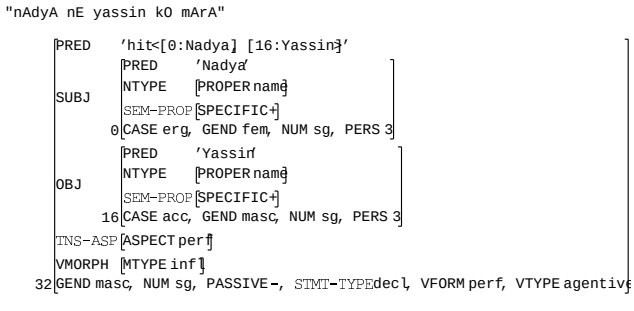


Figure 4: F-structure AVM for (9)

There are two intesting points about this analysis of case in Urdu. The first is that although the Urdu grammar processes case differently than the other grammars, the resulting f-structure in Figure 4 is similar to its counterparts in English, German, etc. English would have CASE nom on the subject instead of erg, but the remaining structure is the same: the only indication of case is the CASE feature. The second point is that Urdu tested the application of inside-out functional uncertainty to case both theoretically and computationally. In both respects, the use of inside-out functional uncertainty has proven a success: not only is it theoretically desirable for languages like Urdu, but it is also implementationally feasible, efficiently providing the desired output.

Another interesting example of how Urdu has extended the standards of the ParGram project comes from complex predicates. The English, French, and German grammars do not need a complex predicate analysis. However, as complex predicates form an essential and pervasive part of Urdu grammar, it is necessary to analyze them in the project. At first, we attempted to analyze complex predicates using the existing XLE tools. However, this proved to be impossible to do productively because XLE did not allow for the manipulation of PRED values outside of the lexicon. Given that complex predicates in Urdu are formed in the syntax and not the lexicon (Butt, 1995), this poses a significant problem. The syntactic nature of Urdu complex predicate formation is illustrated by (10), in which the two parts of the complex predicate *lɪkʰ* 'write' and *diya* 'gave' can be separated.

(10) a. [anjum nE] [saddaf kO] [ciTTHI]
Anjum.F=Erg Saddaf.F=Dat note.F.Nom
[**likHnE** **dI**]
write-Inf.Obl give-Perf.F.Sg
'Anjum let Saddaf write a note.'

b. anjum nE **dI** saddaf kO [ciTTHI **likHnE**]

c. anjum nE [ciTTHI **likHnE**] saddaf kO **dI**

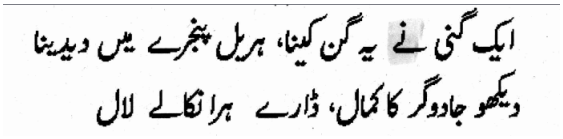
The manipulation of predicational structures in the lexicon via lexical rules (as is done for the English passive, for example), is therefore inadequate for complex predication. Based on the needs of the Urdu grammar, XLE has been modified to allow the analysis of complex predicates via the restriction operator (Kaplan and Wedekind, 1993) in conjunction with predicate composition in the syntax. These new tools are currently being tested by the implementation of the new complex predicates analysis.

5 Script

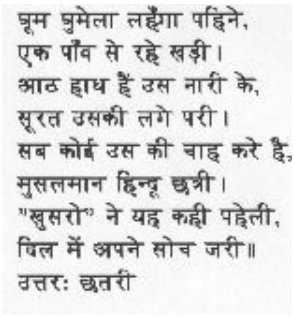
One issue that has not been dealt with in the Urdu grammar is the different script systems used for Urdu and Hindi. As seen in the previous discussions and the Figures, transcription into Latin ASCII is currently used by the Urdu grammar. This is not a limitation of the XLE system: the Japanese grammar has successfully integrated Japanese Kana and Kanji into their grammar.

The approach taken by the Urdu grammar is different from that of the Japanese, largely because two scripts are involved. The Urdu grammar uses the ASCII transcription in the finite-state morphologies and the grammar. At a future date, a component will be built onto the grammar system that takes Urdu (Arabic) and Hindi (Devanagari) scripts and transcribes them for use in the grammar. This component will be written using finite-state technology and hence will be compatible with the finite-state morphology. The use of ASCII in the morphology allows the same basic morphology to be used for both Urdu and Hindi. Samples of the scripts are seen in (11) for Urdu and (12) for Hindi.

(11)



(12)



6 Conclusion

The ParGram project was designed to use a single grammar development platform and a unified methodology of grammar writing to develop large-scale grammars for typologically different languages. At the beginning of the project, three typologically similar European grammars were used to test this idea. The addition of two Asian languages, has shown that the basic analysis decisions made for the European languages can be applied to typologically distinct languages. However, the Asian languages required the addition of a few new standard analyses to the project to cover constructions and analysis techniques not found in the European languages. With this new set of standards, the ParGram project can now be applied to other typologically distinct languages.

The parallelism between the grammars in the ParGram project can be exploited in applications using the grammars: the fewer the differences, the simpler a multi-lingual application can be. For example, a translation system that uses the f-structures as input and output can take advantage of the fact that similar constructions have the same analysis (Frank, 1999). The standardization also aids further grammar development efforts. Many of the basic decisions about analyses and formalism have already been made in the project. Thus, the grammar writer for a new language can use existing technology to bootstrap a grammar for the new language and can parse equivalent constructions in the existing languages to see how to analyze a construction. This allows the grammar writer to focus on more difficult constructions not yet encountered in the existing grammars.

References

Masayuki Asahara and Yuji Matsumoto. 2000. Extended models and tools for high-performance part-of-speech tagger. In *Proceedings of COLING*.

Kenneth Beesley and Lauri Karttunen. 2002. *Finite-State Morphology: Xerox Tools and Techniques.* Cambridge University Press. To Appear.

Miriam Butt and Ron Kaplan. 2002. The morphology syntax interface in LFG. Presented at LFG02, Athens, Greece; to appear in the proceedings (CSLI Publications).

Miriam Butt and Tracy Holloway King. to appear. The status of case. In Veneeta Dayal and Anoop Mahajan, editors, *Clause Structure in South Asian Languages.* Kluwer.

Miriam Butt, Tracy Holloway King, María-Eugenia Niño, and Frédérique Segond. 1999. *A Grammar Writer's Cookbook.* CSLI Publications.

Miriam Butt, Helge Dyvik, Tracy Holloway King, Hiroshi Masuichi, and Christian Rohrer. 2002. The parallel grammar project. In *Proceedings of COLING 2002*. Workshop on Grammar Engineering and Evaluation.

Miriam Butt. 1995. *The Structure of Complex Predicates in Urdu.* CSLI Publications.

Jean-Pierrre Chanod and Pasi Tapanainen. 1995. Creating a tagset, lexicon, and guesser for a French tagger. In *Proceedings of the ACL SIGDAT Workshop: From Texts To Tags. Issues in Multilingual Language Analysis*, pages 58–64.

Anette Frank. 1999. From parallel grammar development towards machine translation. In *Proceedings of MT Summit VII*, pages 134–142.

Ron Kaplan and Jürgen Wedekind. 1993. Restriction and correspondence-based translation. In *Proceedings of the Sixth European Conference of the Association for Computational Linguistics*, pages 193–202.

Ron Kaplan. 1988. Correspondences and their inverses. Presented at the Titisee Workshop on Unification Formalisms: Syntax, Semantics, and Implementation, Titisee, Germany.

John T. Maxwell, III and Ron Kaplan. 1993. The interface between phrasal and functional constraints. *Computational Lingusitics*, 19:571–589.

Rachel Nordlinger. 1998. *Constructive Case: Evidence from Australian Languages.* CSLI Publications.