Inferring Grammars from Interlinear Glossed Text: Extracting Typological and Lexical Properties for the Automatic Generation of HPSG Grammars

Kristen Howell

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2020

Reading Committee:

Emily M. Bender, Chair

Gina-Anne Levow

Edith Aldridge

Program Authorized to Offer Degree:
Department of Linguistics

University of Washington

Abstract

Inferring Grammars from Interlinear Glossed Text:
Extracting Typological and Lexical Properties for the
Automatic Generation of HPSG Grammars

Kristen Howell

Chair of the Supervisory Committee:
Professor Emily M. Bender
Department of Linguistics

This dissertation presents a grammar inference system that leverages linguistic knowledge recorded in the form of annotations in interlinear glossed text (IGT) and in a meta-grammar engineering system (the LinGO Grammar Matrix customization system) to automatically produce machine-readable HPSG grammars. Building on prior work to handle the inference of lexical classes, stems, affixes and position classes, and preliminary work on inferring case systems and word order, I present an integrated grammar inference system called BASIL that covers a wide range of fundamental linguistic phenomena. System development was guided by 27 geneologically and geographically diverse languages, and I test the system's cross-linguistic generalizability on an additional 5 held-out languages, using datasets provided by field linguists. My system out-performs three baseline systems in increasing coverage while limiting ambiguity and producing richer semantic representations than the baselines or previous work in grammar inference.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

The completion of this dissertation would not have been possible without the help of many people. I am grateful to all of my friends, family and colleagues who provided me with encouragement, support and guidance during this project.

To my advisor, Emily M. Bender, thank you for the opportunities you provided me with from the beginning, suggesting me for my first Research Assistantship on the EL-STEC grant and bringing me onto the AGREGGATION team. These projects shaped my perspective of language endangerment and revitalization and ultimately this dissertation. Thank you for your advise and guidance throughout the process and for unblocking me whenever I got stuck. Your kindness and rigor have made me a better linguist and better person.

I'm grateful to my committee for their invaluable perspectives that challenged me as a researcher. Gina-Anne Levow, you introduced me to the challenges of working with low-resource languages and guided me in evaluating my work when a clear point of comparison was difficult to find. Edith Aldridge, thank you for helping me to identify phenomena that were relevant to my work and pushing me to consider analyses across frameworks.

I am building on years of previous work in the AGREGGATION project and am beyond grateful for the researchers whose systems I leveraged and built on. Thank you Ryan Georgi, Michael Goodman, and Joshua Crowgey for your patience explaining to me how your systems work and making changes to them even after you graduated. Thanks to Olga Zamaraeva who patiently explained the inner-workings of the MOM morphological inference system to me and who talked through the inference algorithms with me over the years. Finally, thanks to Fei Xia for her guidance using the previous AGG systems and developing inference algorithms.

In addition to all of the support I received from the AGG team, I grateful to all of my colleagues who gave me advice and feedback over the years: Laurie Dermer, David Inman, Courtney Mansfield, Angelina McMillan-Major, Anna Moroz, Elizabeth Nielson, Woodley Packard and Glenn Slayden.

I want to express my gratitude to the many linguists who were willing to share their data with me for use in this project: Balthasar Bickel, Claire Bowern, Shobhana Chelliah, Andrew Cowell, Heidi Harley, Bryn Hauk, David Inman, Daniel Kaufman, František Kratochvíl, Lev Michael, Rachael Nordlinger and Nick Thieberger. The work that goes into these documentation projects is unbelievable and I want to thank all of their collaborators and everyone else who pointed me to a corpus or participated in its curation. Most of all, thank you to the speaker communities who agreed to have their language recorded and gave their time to share their language and culture with these linguists.

During my time in the linguistics department, I received so much help from all of the faculty and staff. In particular, thanks to Mike Furr, Joyce Parvi, Brandon Graves and Misha Burgess who went out of their way to help me so many times over the years.

This dissertation is based upon work supported by the National Science Foundation under Grant No. BCS-1561833 (PI Bender). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. This research was also funded by the University of Washington Department of Linguistics Excellence in Linguistic Research Graduate Award.

I want to thank my friends and family for all of their support through out this project. From your patience to your regular words of encouragement, I couldn't have done it without all of you. Finally, I am deeply grateful to my husband and partner who believed in me and invested in my career. Your support, encouragement and taking on the lion's share of our responsibilities made it possible for me to complete this degree. Thank you.

DEDICATION

To David Howell

Chapter 1

INTRODUCTION

Machine-readable grammars for human languages that are grounded in theoretical syntactic formalism can be useful tools in the context of endangered language documentation and revitalization. First, they support treebanking (Oepen et al., 2002), which in turn supports data exploration (Letcher and Baldwin, 2013; Bouma et al., 2015); and second, they facilitate the development of tools such as grammar checkers (da Costa et al., 2016) and automated tutors (Hellan et al., 2013). In spite of these advantages, the use of such grammars is hindered by the time-consuming process of developing them together with the need of a specific skillset required for grammar engineering, which is distinct from the skills involved in documentation itself. Fortunately, the development of Grammar Engineering toolkits, such as the LinGO Grammar Matrix (Bender et al., 2002, 2010), significantly reduces the up-front work of creating machine readable grammars. At the same time, many endangered languages, while not having a lot of written data from a quantitative perspective, have been documented thoroughly in the form of interlinear glossed text (IGT), which encodes rich linguistic information in the annotations. Drawing on these two linguistic resources, this dissertation investigates whether machine-readable grammars can be generated automatically by inferring lexical, morphological and syntactic properties about the language from existing linguistic corpora.

Endangered languages tend to also be low-resource languages (languages for which relatively little data is available) and therefore pose a challenge to typical natural language processing techniques that rely on large amounts of data to train a model. Furthermore, theoretically grounded grammars of the type I seek to create go beyond labeling and producing structured representations: They are well-formed formal objects that represent hypotheses

about human language. The goal of producing such a complex output from a rather limited (from a quantitative perspective) input is an ambitious one. Fortunately, I have two rich sources of linguistic knowledge to draw on: a grammar engineering toolkit that contains stored analyses for a range of syntactic phenomena, and the linguistic information, both explicit and implicit, recorded in IGT corpora.

While linguistic phenomena vary cross-linguistically, many can be formally modeled with the same analysis or using analyses that are customizable from a shared starting point. Based on this assumption, the LinGO Grammar Matrix customization system (Bender et al., 2002, 2010) contains syntactic analyses in Head-driven Phrase Structure Grammar (HPSG; Pollard and Sag, 1994) in the form of a core grammar, hypothesized to be shared across languages, and a series of typologically-informed libraries of analyses of cross-linguistically variable phenomena. Taking as input a grammar specification file which includes syntactic, morphological and lexical information about a specific language, the Grammar Matrix customization system identifies which analyses model the combinations of phenomena in that language. The output is a fully customized, machine-readable HPSG grammar which parses and generates strings in the target language. Leveraging this customization system for grammar generation, my work focuses on the task of inferring typological properties about languages from corpora of IGT data and encoding this information in grammar specifications which can be input into the customization system.

In order to infer the characteristics of a specific language, I draw on corpora of IGT (illustrated by the following example from Chintang [ISO 639-3: ctn]), whose annotations show what grammatical information is marked morphologically and provide further information implicitly via a translation into a language of broader communication.[1] In (1) the gloss line shows explicitly the person (third) and number (non-singular) of the subject/agent through the gloss 3nsS/A and the person (third) of the patient through the gloss 3P, as well as the mood (indicative; IND), tense (non-past; NPST) and the fact that the verb is negated (NEG).

[1]Here, and in all examples I work with, this language of broader communication is English.

Following Xia and Lewis (2007) and Georgi (2016), I glean implicit information from the English translation by parsing it to obtain part-of-speech tags and syntactic dependencies and projecting this information onto the sentence. Among other things, this provides a hypothesized argument structure of the verb: that it has a subject and complement, and that its complement is *aru*.

(1) Aru uŋisukVniŋ.
aru u-ŋis-u-kV-niŋ
another 3nsS/A-know-3P-IND.NPST-NEG
'They did not know another [language].' [ctn] (Bickel et al., 2013a)

One advantage of IGT corpora is that, while their curation is time intensive, they represent the standard method for recording and annotating linguistic data. As a result, such corpora exist for many endangered and low-resource languages, even in cases where it is the only form of written data. In fact, due to the efforts of field linguistics and archivists, a number of archives make IGT data publicly available (see inter alia, the Alaskan Native Languages Archive (ANLA), the Pacific and Regional Archive for Digital Sources (PARADISEC), the Endangered Languages Archive (ELAR), the Archive of Indigenous Languages in Latin America (AILLA), Kaipuleohone, Kratylos and Multi-CAST).[2] These resources, in addition to yet-to-be archived datasets which field linguists have generously shared, enabled me to test the system presented in this dissertation on a wide range of languages from all over the world.

Leveraging these resources, this dissertation investigates whether and how I can create machine-readable HPSG grammars for typologically diverse languages on the basis of IGT corpora and the Grammar Matrix. To answer these questions, I introduce BASIL, a system for Building Analyses from Syntactic Inference in Low-resource languages, which automatically infers grammar specifications from IGT data. I build on previous work by the AGGREGA-

[2]Links to these repositories are provided in Appendix A.

TION project (Bender et al., 2013, 2014; Wax, 2014; Zamaraeva, 2016; Howell et al., 2017; Zamaraeva et al., 2017, 2019a) to produce the following contributions: (1) I integrate all existing inference modules into a single system to which (2) I add modules for additional grammatical phenomena and (3) where previous end-to-end testing treated only a single language, I use 27 languages in development, doing end-to-end system testing on 9 of the 27, and then evaluate the system on 5 additional held-out languages not considered during system development.

I begin by situating my work on grammar inference against the broader background of automatic grammar generation (which includes grammar induction, grammar extraction and grammar inference) in Chapter 2. In Chapter 3 I provide background on grammar inference in the context of the AGGREGATION project and present an end-to-end pipeline that begins with an IGT corpus and results with a machine-readable-grammar which can be loaded into parsing and treebanking software to inspect the syntactic and semantic representations it produces for sentences in the target language.

I describe my methodology for grammar inference, including the algorithms and heuristics I implemented in BASIL, in Chapter 4. This methodology uses the morphotactic inference system developed by Wax (2014), Zamaraeva (2016) and Zamaraeva et al. (2017), which we integrated with syntactic inference in Zamaraeva et al. 2019a. I describe a number of additional algorithms that I developed for syntactic inference and integrated with morhpological inference. I developed this methodology based on the typological literature, the functionality provided by specific libraries in the Grammar Matrix and specific languages which represent these phenomona. For this reason, I include an overview of each phenomenon as well as examples from the development data. In Chapter 5, I describe the specific languages that I consulted during development, how the core development langaugage informed design decisions and the BASIL's performance on those languages.

I describe a methodology for using the DELPH-IN suite of software tools to evaluate my inferred grammars by parsing and treebanking held-out portions of the data for each language and describe three baseline system to which I compare BASIL in Chapter 6. I apply

this evaluation methodology to *held-out languages*, languages that I did not consider during development, which I also describe in Chapter 6. Finally, in Chapter 7, I present the results of this evaluation, providing discussion of the system's performance and error analysis. I conclude in Chapter 8 with a discussion of both the applications and limitations of this work, proposing both areas for future work and downstream uses of the system.

Chapter 2

AUTOMATIC GRAMMAR GENERATION

Interest in creating machine-readable grammars is likely as old as the field of computational linguistics itself, with published work in *grammar engineering* — the process of creating machine-readable grammars by hand — going back at least as far as Zwicky et al. (1965) and continuing into the present day. My work on grammar inference builds on grammar engineering work (in the form of the Grammar Matrix; Bender et al., 2002, 2010), but also fits into a tradition of work on *automatic grammar generation*, which is the development of systems that automatically create grammars on the basis of data. Within automatic grammar generation, I distinguish four broad categories of approaches, differentiated by the types of inputs they take: *grammar induction from strings* — automatic grammar generation based on text alone (§2.1); *grammar extraction* — automatic grammar generation based on treebanks (§2.2); *grammar induction from meaning representations* — automatic grammar generation based on strings paired with some form of semantic representation (§2.3); and *grammar inference* — automatic grammar generation based on text annotated with partial grammatical information but not full parse trees or logical forms (§2.4).

Just as these four approaches to grammar generation differ in their input, they also differ in the types of grammars they can produce. Grammar induction, if working from strings alone, will produce noisy representations that align only partially with structures created by linguists. Grammar extraction will produce grammars that provide the same kind of representations as given in the source treebank and similarly, grammar induction based on strings paired with semantic representations will produce grammars that can output those semantic representations. In each of these cases, the generated grammar will also include a parse selection model, based on observed patterns in the corpus. Grammar inference

systems, by contrast, draw on both partial annotation in their input data and some external source of grammatical knowledge. For this reason, inferred grammars can generate richer representations than what is in the input.

2.1 Grammar Induction from Strings

The term *grammar induction* has been used to describe a large body of work in grammar generation ranging from the simplest case of producing grammars on the basis of strings alone to pairing these strings with target semantic representations. Here, I differentiate between the two on the basis that richness of the resulting grammar is contingent on the level of linguistic information in the input and begin with grammar induction from just surface strings.

Often characterized as an incomplete data problem (see inter alia Klein and Manning, 2001), where the complete data would be a corpus of trees, grammar induction from surface strings seeks to produce grammars solely on the basis of text. Early grammar induction work focused on producing context-free grammars (CFGs), which involved two components: (1) identifying constituents and (2) identifying their categories (see Klein and Manning, 2001, 2002). Klein and Manning (2004) improved upon this work by inducing an unlabeled syntactic dependency grammar and combining it with the induced CFG for better performance parsing over English [eng], German [deu] and Mandarin [cmn]. This basic approach has informed work which further tuned the algorithm by preferring short vs. long dependencies and testing on additional languages, as in Smith and Eisner 2006. While unlabeled syntactic dependencies can be inferred from text and are useful for some tasks, they do not provide any information regarding the type of syntactic relationship between two constituents. Therefore, other methodologies of automatic grammar generation have focused on using inputs that are encoded with more linguistic information.

2.2 Grammar Extraction

In contrast with the impoverished input used by grammar induction from surface strings, grammar extraction uses the syntactic information available in treebanks, collections of syntactic trees, to define grammars. Typically these grammars are produced by walking the trees in a treebank, collecting rules that could produce those structures and pruning to remove redundant rules, taking rule probability into account in order to make the grammar more efficient (Krotov et al., 1998).

Because an extracted grammar is informed by the formalism implicit in the tree structures in the input, the extracted grammar will produce trees with roughly the same amount of syntactic information as the formalism used to create the treebank, whether that is a context-free grammar (CFG) as in Krotov et al. 1994 or a formal grammar such as HPSG (Pollard and Sag, 1994) as in Simov 2002. However, while the level of detail in the treebanked parses limits that of the resulting grammar, work has been done to extract a grammar in a different formalism than that represented in the input. Xia (1999), for example, proposed an algorithm to do additional bracketing on the Penn Treebank II-style trees (Marcus et al., 1994) in order to extract a Lexical Tree Adjoining Grammar (LTAG), which was more expressive than the CFG in the input. In particular, the Penn Treebank II trees did not explicitly mark a head/argument/modifier distinction. Using a head-percolation table, Xia's algorithm predicted the head daughter of each node and the type of relationship it had with its sister node(s). From this enriched input, the algorithm extracted a grammar with slightly more syntactic information than that in the original input.

While the above work has focused on grammars of English, grammar extraction is possible in principle for any language for which a treebanked corpus is available. Shirai et al. 1995, Chen and Hsieh 2004 and Satayamas and Kawtrakul 2004 are a few examples of early work extracting grammars of Japanese, Chinese and Thai, respectively. More recently, the Universal Dependencies Treebank (Nivre et al., 2016), a collection of treebanks which currently contains syntactic dependency annotations for sentences in 91 languages, has facilitated work

on grammar extraction for a much broader range of languages (see inter alia Agić et al., 2016; Noji et al., 2016; Han et al., 2019). My goals in this work, however, are to generate grammars for low-resource languages, many of which are not represented in the UD collection, and to produce syntactic and semantic representations, which are richer than dependency parses.

2.3 Grammar Induction from Meaning Representations

In contrast with grammar extraction which relies on a treebank of syntactic parses, grammar induction from strings paired with semantic representations relies on a treebank of sorts, typically pairing sentences with either semantic dependencies or logical forms. At the same time, this methodology, which does not induce syntactic structure from the input, induces syntactic grammars along the lines of those in Section 2.1, leveraging semantic information in the input as well.

Kate et al. (2005) and Kate and Mooney (2006) used strings paired with formal meaning representations as training data to induce grammars that could map strings of natural language to meaning representations in formal query languages. Jones et al. (2013) used an even richer type of semantic representation in the input. Working with the semantic dependencies in the Redwoods treebank (Oepen et al., 2004),[1] they induced a hyperedge replacement grammar that can produce the same semantic dependencies.

In order to generate grammars that are more robust with respect to morphological complexity, some grammar induction work has included external linguistic information in the input in addition to the surface strings and meaning representations. For example, Zettlemoyer and Collins (2005, 2007), who induced combinatory categorial grammars (CCG; Steedman, 2000) of English, used hand-generated lexical templates to extract lexical items from the logical forms in the input. Later work by Kwiatkowski et al. (2011) addressed the problem of lexical sparsity in a more language-agnostic manner, using an algorithm that mapped between lexemes, paired with a logical form, and lexical templates that captured inflection.

[1] This treebank also includes syntactico-semantic annotations in HPSG, but Jones et al. (2013) only used the semantic dependencies in their input.

Due to the richness of semantic information in the input, grammars induced from text paired with semantic representations rather than text alone are capable of capturing much more detailed and meaningful semantic relations than the unlabeled syntactic dependency relations produced by grammars induced only from surface forms. Even so, these semantic representations are less rich than those we can produce when we use a precise, linguistic grammar whose semantic parses include predicate-argument structures as well as syntactico-semantic features for each argument, as I will detail in Chapter 3.

2.4 Grammar Inference

Grammar inference systems take as input a collection of text with partial grammatical annotations and utilize some external source of grammatical knowledge that is not specific to the language at hand to produce grammars that give richer representations than those produced by grammar induction without requiring a treebank (the existence of which indicates that some grammar, machine-readable or otherwise, already exists and produced the trees in the treebank). While these systems generally are not probabilistic and do not necessarily include a parse-selection model, as is common with induced or extracted grammars, they allow us to automatically generate formal linguistic grammars on the basis of data with some linguistic annotation.

To produce grammars in the Minimalist Grammar formalism (MG; Stabler, 1996), a variant of the Minimalist Program (Chomsky, 1995), Indurkhya (2020) used a set of sentences annotated for part-of-speech (POS), agreement, predicate-argument structure and clause type (interrogative or declarative). This system inferred a lexicon for English on the basis of those annotations, pruned it with a set of Minimalist axioms, and combined it with a non-language-specific notion of merge (with internal and external subtypes) to create a machine-readable Minimalist grammar.

Whereas Indurkhya used a custom annotation scheme for the input data, Hellan (2010) and Bender et al. (2014) leveraged the rich annotation already present in interlinear glossed text (IGT), illustrated in (1) and repeated here as (2). IGT is a particularly rich source of

data because it includes morpheme segmentation, glosses for each morpheme which encode morpho-syntactic information and a translation into a high-resource language (frequently English). A particularly attractive fact about IGT data is that it is the format broadly used in linguistics to record data during collection and analysis, so IGT corpora exist for many languages that do not otherwise have very much written text.

(2) Aru uŋisukVniŋ.
aru u-ŋis-u-kV-niŋ
another 3nsS/A-know-3P-IND.NPST-NEG
'They did not know another [language].' [ctn] (Bickel et al., 2013a)

Hellan (2010) and Hellan and Beermann (2011) inferred grammars using a combination of specially annotated IGT and a grammar engineering toolkit called *TypeGram*. TypeGram is based on the DELPH-IN Joint Reference Formalism (Copestake, 2002a) which supports the development of typed feature structure grammars, typically within the framework of Head-driven Phrase Structure Grammar (HPSG; Pollard and Sag, 1994). Hellan (2010), however, positioned TypeGram as a hybrid of HPSG and Lexical Functional Grammar (LFG; Kaplan and Bresnan, 1982). The special annotations on the IGT involved labels which indicate syntactic properties (including, e.g. valence patterns and constructions such as passive) of the sentence annotated. The TypeGram resource included grammatical rules which are named by the same inventory of label types and thus could directly instantiate a grammar off of an appropriately annotated corpus. Finally, although these papers did not include evaluation of the system, they included examples from Ga [gaa] and Kiswahili [swh].

Bender et al. (2014) took a similar approach, producing HPSG grammars in the DELPH-IN formalism on the basis of IGT data. Their approach differed, however, both in the expectations of the input IGT and the grammar engineering toolkit. For IGT, they worked directly from the type of annotations typically produced by documentary linguistics projects, that is, IGT with thorough segmentation and glossing at the morpheme level, but no clause-level annotations. The grammar engineering toolkit they used was the LinGO Grammar

Matrix, (Bender et al., 2002, 2010), which allows the user to define a grammar specification that selects from a typologically broad catalogue of analyses for different syntactic phenomena and pairs these analyses with a core grammar used across languages. In addition to inferring a lexicon and morphological rules from IGT, Bender et al. (2014) inferred syntactic properties (using a couple of different approaches), which they encoded in grammar specifications to produce custom grammars with the appropriate analyses for Chintang.

My goal is the automatic generation of grammars for low-resource languages. I want to create grammars that are more syntactically precise than those created with grammar induction and in many cases neither a treebank nor a collection of sentences paired with semantic representations is available. However, for many low-resource languages, IGT corpora are available, and therefore provide the rich input that we can draw upon for grammar inference. Therefore, I build on the approach set forth by Bender et al. (2014), which I describe in detail in the next chapter. I present a pipeline for grammar inference from IGT which includes morphological and syntactic inference and extend the range of phenomena that the inferred grammars cover. In addition, my review of the grammar induction literature shows a strong preference in the field for work on relatively high-resource languages from a limited set of language families. While grammar extraction has been done for a wide range of languages, grammar extraction of syntactically precise grammars of the sort I am focused on (in this case using HPSG) would require a treebank in that formalism, which is fairly hard to come by. In the interest of generating grammars for low-resource languages, I use a diverse set of languages for system development, focusing on ones that are low-resource. Using this methodology, I present an inference system that is intended to work for any language for which an IGT corpus is available.

Chapter 3

THE AGGREGATION PROJECT

This dissertation is situated within the AGGREGATION Project[1] (Bender et al., 2013, 2014), whose goal is to provide the benefits of implemented, formal grammars to documentary linguists, without their having to invest time in developing those grammars by hand. Such grammars are useful for testing linguistic hypotheses against data (Bierwisch, 1963; Müller, 1999; Bender, 2008a; Fokkens, 2014) as well as building treebanks which are useful for discovering examples of phenomena in a language (Bender et al., 2012; Letcher and Baldwin, 2013; Bouma et al., 2015). The task of developing a grammar by hand is very time consuming and not likely to be taken up by field linguists already busy with the work of language documentation and description. However, the detailed analysis involved in annotating IGT data (another time consuming task that documentary linguists are doing anyway) provides a very rich starting point for producing these grammars automatically. Therefore, an end-to-end pipeline that begins with an IGT corpus of a language and results in a machine-readable grammar has the potential to serve the language documentation community without requiring additional work on their end, either in the form of data curation or grammar engineering.

In (3) I present an example of interlinear glossed text (IGT) from the Chintang Language Research Project (CLRP; Bickel et al., 2013d). Based on the information encoded in this IGT and others in the corpus, our goal is a grammar that parses this sentence to produce a syntactic representation in the Head-driven Phrase Structure Grammar formalism (HPSG; Pollard and Sag, 1994) like the one in Figure 3.1 and a semantic representation in Minimal Recursion Semantics (MRS; Copestake et al., 2005) as in Figure 3.2.

---

[1]Links to the repositories for this project and others used in this dissertation are provided in Appendix A.

(3) Aru uŋisukVniŋ.

aru u-ŋis-u-kV-niŋ

another 3nsS/A-know-3P-IND.NPST-NEG

'They did not know another [language].' [ctn] (Bickel et al., 2013a)

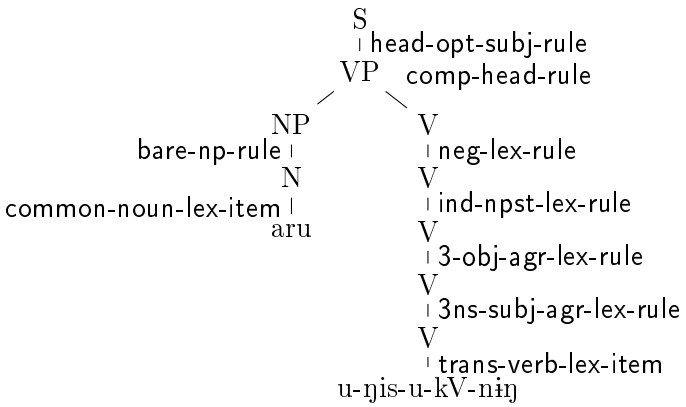


Figure 3.1: The parse tree for the sentence in (3), which was generated by an inferred grammar of Chintang and corresponds to the semantic representation in Figure 3.2

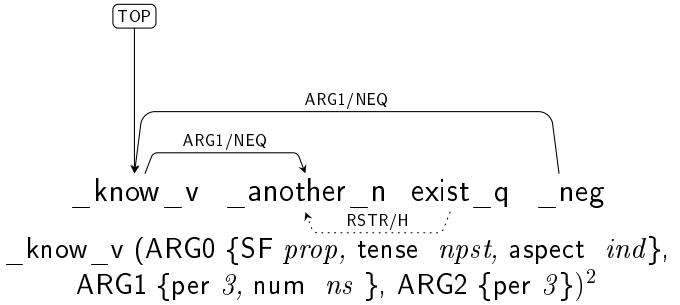


Figure 3.2: The semantic representation for the sentence in (3) which was generated by an inferred grammar of Chintang

[2]The graph representation above does not show morphosemantic features. The underlying feature structure that includes these is much larger, so I have listed the features on the predication for the verb here.

Without any external source of linguistic knowledge, going from an IGT corpus to an implemented HPSG grammar capable of producing the representations in Figures 3.1 and 3.2 would be a difficult inference problem to solve. However, in the AGGREGATION project, we have established a pipeline that leverages a number of existing resources to extract all of the necessary information from an IGT corpus and produce a customized grammar for that language. This pipeline, illustrated in Figure 3.3, expects as its starting point an IGT corpus[3] that was collected by a field linguist, which we convert to an extensible and flexible XML-based format for IGT data called Xigt (Goodman et al., 2015). We then enrich the IGT using INTENT (Georgi, 2016), which projects syntactic dependencies and part-of-speech (POS) tags onto words in the language from a parse of the English translation.

The enriched corpus provides four key components that are necessary for grammar inference: morpheme segmentation, glossing, POS tags and syntactic dependencies, as illustrated in the final box in Figure 3.4. The morpheme segmentation and glossing are provided by the linguist and are necessary to extract a lexicon, infer the morphotactic system and associate morpho-syntactic and morpho-semantic information with their respective morphemes. POS tags are often provided by the documentary linguist in the original corpus, but if they are not, they can be acquired from INTENT. Following the methodology set forth by Xia and Lewis (2007), INTENT creates alignments between the English translation and the sentence by looking for string matches between the translation and gloss line. It then parses the English sentence and projects the POS and syntactic dependency tags from the English parse onto the aligned words in the source language.[4] Finally, the projected dependencies allow us to discriminate between arguments, modifiers and conjuncts and to identify different types of constituents in the sentence in order to infer syntactic properties.

---

[3]Typically our source corpora are encoded as Toolbox (SIL International, 2015) or FLeX (Rogers, 2010); though in principal any source format could be used.

[4]Projected tags are sometimes incorrect or missing because they require a corresponding word in the English translation, which does not always exist, and rely on successful alignment between the translation and language, which can be challenging if different lemmas are used in the free-text translation than the gloss line. Nevertheless, these projected tags can provide valuable insight into a language's structure.

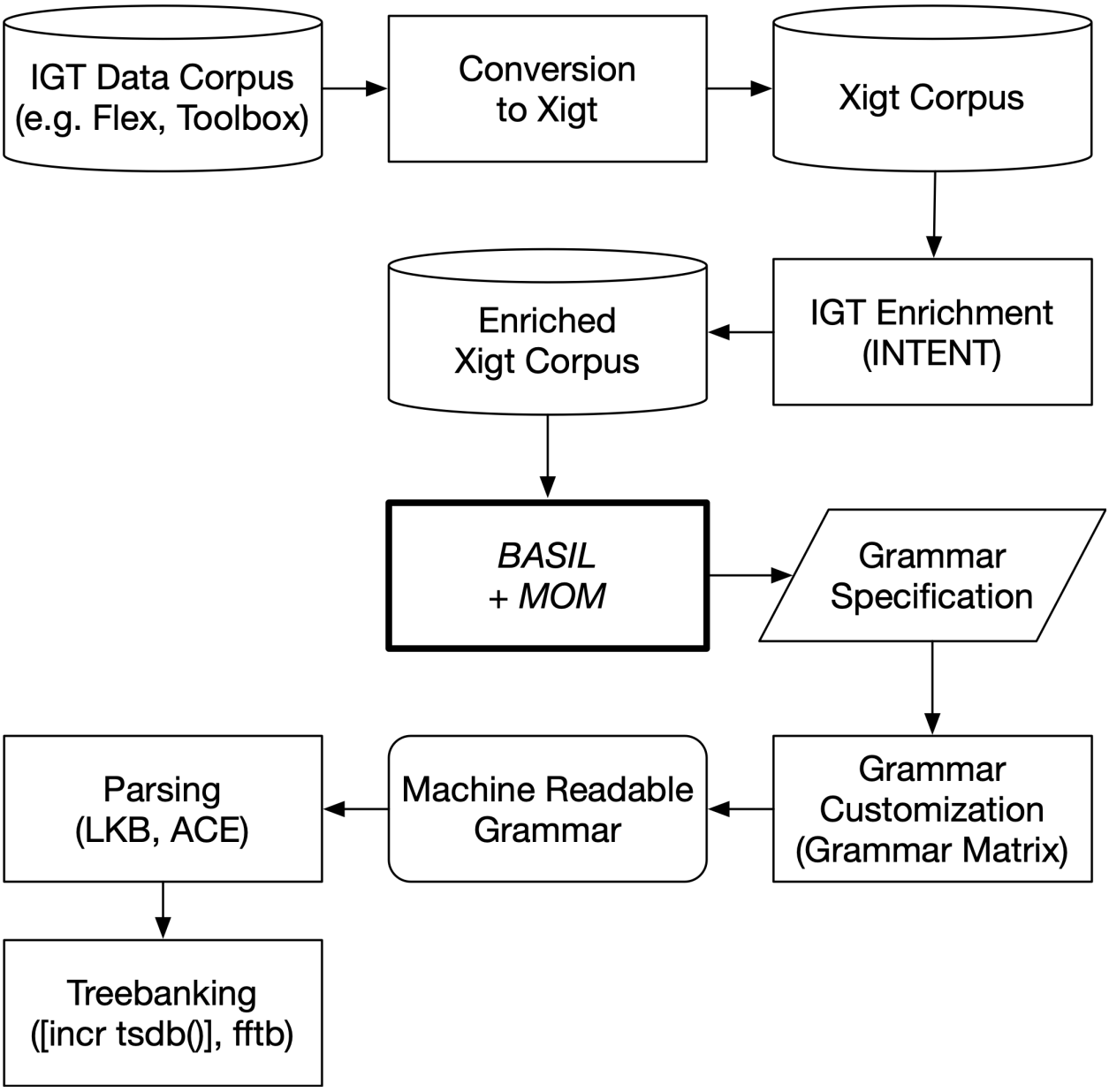


Figure 3.3: AGGREGATION Pipeline

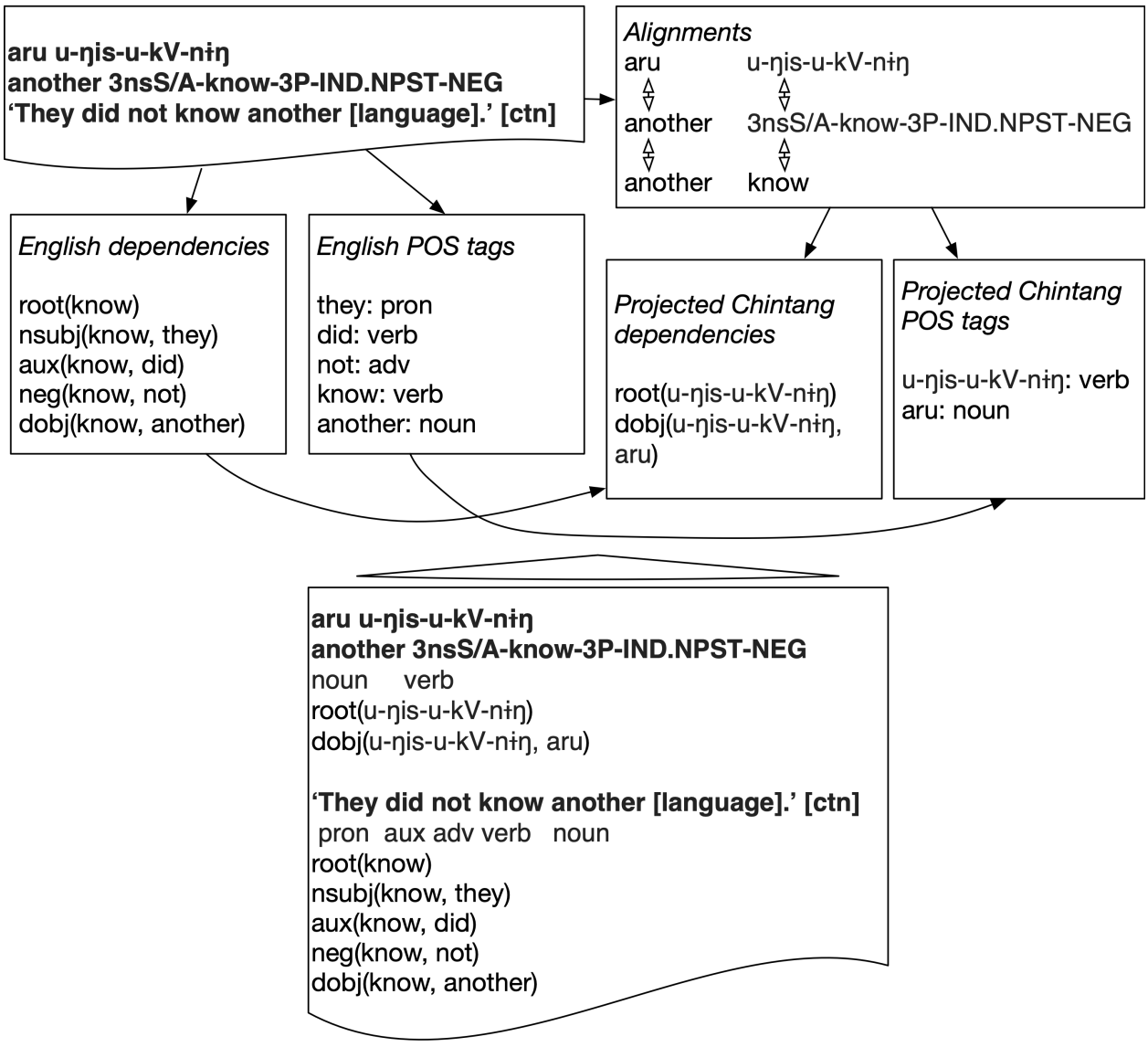


Figure 3.4: IGT Enriched with INTENT (Georgi, 2016)

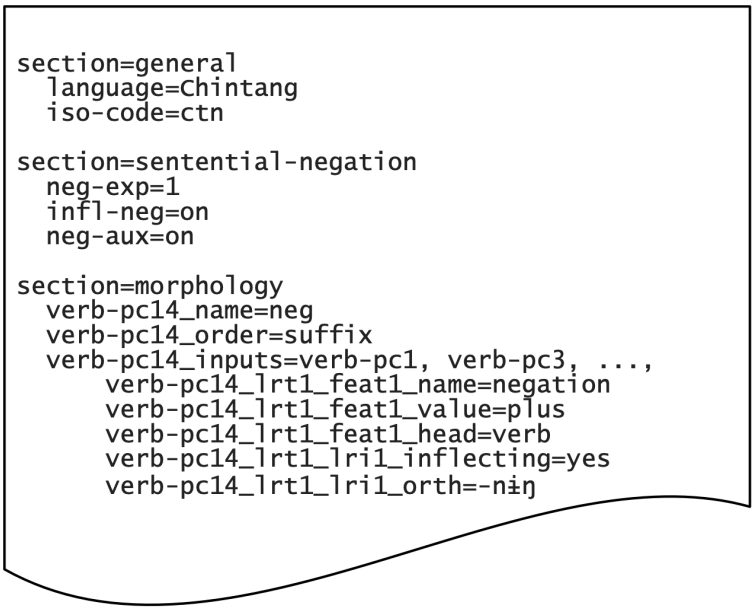
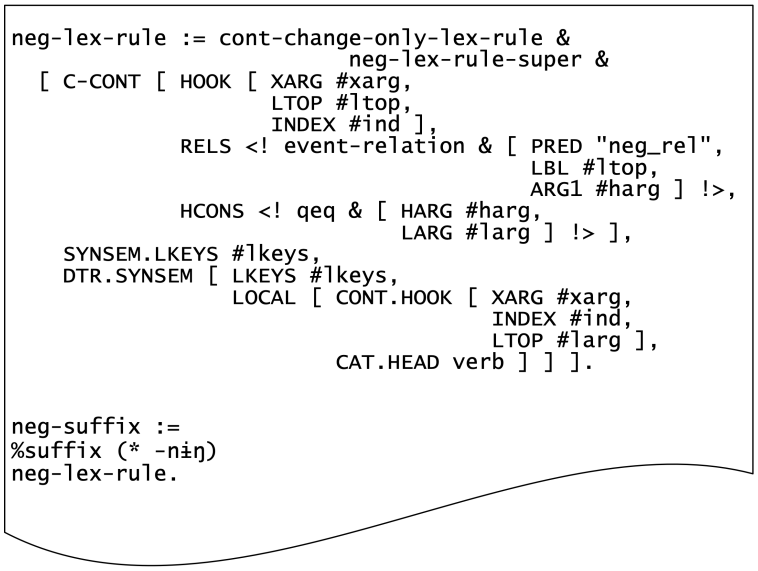


Figure 3.5: A portion of an inferred grammar specification containing (some of) the relevant specifications for sentential negation in Chintang

In this dissertation I present an inference system called BASIL, which uses an enriched corpus to produce a grammar specification file. As an example of BASIL's target output, Figure 3.5 illustrates some of the values it infers that are relevant to sentential negation in Chintang [ctn]. Sentential negation in Chintang is expressed with a suffix *-niŋ*. This is defined in the grammar specification by setting the negation exponence (`neg-exp`) to 1, indicating that negation is expressed with a single morpheme. The morphology section of the grammar specification contains one or more lexical rules for a morpheme with the orthography *niŋ* and morphosemantic feature `negation: plus`. This grammar specification can be input to the Grammar Matrix customization system (Bender et al., 2010), which uses stored syntactic analyses to produce customized grammars for languages based on the specification. The customized grammar generated by the Grammar Matrix for the specification represented in Figure 3.5 will contain the appropriate lexical rule(s) to model negation, which are illustrated in Figure 3.6.



```
neg-lex-rule := cont-change-only-lex-rule &
                          neg-lex-rule-super &
  [ C-CONT [ HOOK [ XARG #xarg,
                    LTOP #ltop,
                    INDEX #ind ],
             RELS <! event-relation & [ PRED "neg_rel",
                                        LBL #ltop,
                                        ARG1 #harg ] !>,
             HCONS <! qeq & [ HARG #harg,
                              LARG #larg ] !> ],
    SYNSEM.LKEYS #lkeys,
    DTR.SYNSEM [ LKEYS #lkeys,
                 LOCAL [ CONT.HOOK [ XARG #xarg,
                                     INDEX #ind,
                                     LTOP #larg ],
                         CAT.HEAD verb ] ] ].


neg-suffix :=
%suffix (* -nɨŋ)
neg-lex-rule.
```

Figure 3.6: The relevant lexical rule for negation in the Chintang grammar that was produced from the specification in Figure 3.5

The lexical rule in Figure 3.6 is expressed in the DELPH-IN joint reference formalism (Copestake, 2002a), which can be used to implement HPSG-style typed feature structures. A grammar encoded in this way can be loaded into the LKB (Copestake, 2002b) and ACE (Crysmann and Packard, 2012) for parsing and [incr tsdb()] (Oepen, 2001) and FFTB (Packard, 2015) for treebanking. This lexical rule licenses the topmost V node in Figure 3.1 and is responsible for introducing the **neg_rel** predication shown in Figure 3.2.

Previous work in the AGGREGATION Project has produced grammar specifications that contain a lexicon of nouns and verbs, morphological rules[5] and descriptions of the language's word order, case-system and case-frame. The lexicon and morphotactic rules are inferred using MOM (Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017), which I describe in Sections 4.2 and 4.3. This system targets the morpheme-segmented line of the IGT, identifies roots, prefixes and suffixes and creates a graphical representation of the possible morphotactic

[5]These rules abstract away from morphophonology. Accordingly the grammars are tested by parsing the morpheme-segmented line of the IGT.

combinations. This graph is output in the form of a grammar specification like the one in Figure 3.5 which can be input to the Grammar Matrix to produce a grammar.

Bender et al. (2013) present the first version of the AGGREGATION inference system, inferring just the word order and case system of a language, which they evaluate by comparing the inferred grammar specification with a gold standard specification, which was constructed on the basis of descriptive resources for each language. Bender et al. (2014) take this a step further by producing separate grammar specifications for Chintang [ctn] using the morphotactic inference of Wax (2014) and the syntactic inference of Bender et al. (2013) and evaluating the resulting grammars in terms of their coverage over held-out sentences. In Zamaraeva et al. 2019a, we integrate the morphological and syntactic inference systems to create separate verb classes based on transitivity to improve on the results of Bender et al. (2014).

With BASIL, I increase the number of phenomena that the system infers by building on the existing morphotactic and syntactic inference systems. I infer additional lexical items including determiners, case-marking adpositions, coordinators and auxiliaries as well as syntactic properties including argument optionality, sentential negation and coordination. I also integrate syntactic and morphological inference to model person, number and gender on nouns, enforcing agreement with verbs, and tense, aspect and mood contributed morphologically or by auxiliaries. Finally, whereas previous work has either evaluated the correctness of the grammar specifications on a variety of languages (Bender et al., 2013; Howell et al., 2017) or grammar performance on a single language (Bender et al., 2014; Zamaraeva et al., 2019a), I evaluate my system in terms of grammar performance on 14 genealogically and geographically diverse languages.

Chapter 4

METHODOLOGY: INFERRING GRAMMAR SPECIFICATIONS

This chapter focuses on my approach to inferring grammar specifications for a given language based on an interlinear glossed text (IGT) corpus. In particular, I present my inference system BASIL, Building Analyses from Syntactic Inference in Low-resource languages, and the specific algorithms it uses for syntactic inference. Situated within the larger pipeline described in Chapter 3 and illustrated in Figure 3.3, BASIL takes an IGT corpus which has been converted to the Xigt (Goodman et al., 2015) data type and enriched using INTENT (Georgi, 2016) as input and produces a grammar specification file from which the Grammar Matrix (Bender et al., 2002, 2010) can generate a machine-readable grammar for the language. I take as my starting point the system of Zamaraeva et al. (2019a) in which we integrated the morphological inference module (called MOM; Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017) with a module for inferring syntactic properties (Bender et al., 2014; Howell et al., 2017). To this integrated system, which handles intransitive and transitive verbs, nouns, case system and case frame, I contribute the inference of additional lexical items including auxiliaries, determiners, case-marking adpositions, coordinators and negation words as well as definitions for syntactic properties including argument optionality, sentential negation and coordination. Furthermore, leveraging syntactic and semantic features added to the lexical items and morphological rules by MOM, I integrate syntactic and morphological inference by defining person, number, gender, tense, aspect and mood in the grammar specification so that agreement can be enforced between nouns and verbs and the meaning contributed by these features is encoded in the semantics.

To establish the scope of my inference system, I begin with an outline of the phenomena covered by my grammar inference system and describe the components of the inferred

grammar specifications in Section 4.1. Next I describe the lexical (§4.2), morphotactic (§4.3), syntactico-semantic (§4.4) and syntactic (§4.5) components of the inference system, providing for each an overview of the typological range covered and what specifications the Grammar Matrix customization system requires. I follow the description of each phenomenon and its typological range with an explanation of the algorithms I use to infer the appropriate specifications for a language based on IGT data. To illustrate the phenomena and my approach to inference, I draw on examples from the set of development languages that I used to design and tune the algorithms described in this section. I will provide a more detailed description of these languages and datasets in Chapter 5.

4.1 Overview of the Grammar Specification

The grammar specification[1] contains specifications for a lexicon, a collection of morphological rules and definitions of syntactic properties for the language at hand. These specifications take the form of features with either fixed or open-ended values, depending on the linguistic characteristics being defined. While a number of phenomena can be defined in the Grammar Matrix, this dissertation focuses on a particular subset of lexical items and syntactic phenomena, which are modeled by a number of open-ended features, as well as 50 fixed features with 136 possible values in the Grammar Matrix grammar specifications.[2]

In this chapter, I describe cases where the system infers all of the possible values for a particular feature and cases where I chose a default value. For some features, multiple values lead to similar coverage in the resulting grammars, so I simplify the system by focusing on a subset of the possible values. In other cases a particular value may be difficult to infer with sufficient accuracy from the available data and in still others, it may be so typologically rare that it is more likely to be inferred in error than correctly. Due to these decisions, the inference system designed to target 99 of the 136 values, as summarized in Table 4.1.

[1]The Grammar Matrix literature often refers to this specification as a 'choices file'.

[2]While not all of the combinations of these features and their values are allowed by the Grammar Matrix's validation system or make sense from a linguistic perspective, the quantification in this chapter will only look at the counts of individual features and not at possible combinations.

Phenomenon	# possible values	# targeted by inference
noun lexical entry	4	2
verb lexical entry	4	2
auxiliary lexical entry	6	4
adposition lexical entry	3	3
morphological rule	5	5
person	9	8
tense	2	1
word order	10	9
determiner order	4	4
auxiliary order	9	9
case system	9	3
argument optionality	18	15
sentential negation	41	23
coordination	12	11
total	136	99

Table 4.1: The number of possible values for the features with a fixed set in the grammar specification and those targeted by the inference system

In addition to features with a fixed set of values, a number of features are open ended. While individual lexical entries and morphological rules have features that must be specified from a fixed set of values, the number of lexical items and morphological rules depends on the number of forms attested in the training corpus. Thus the size of the lexicon and morphology sections of the grammar specification varies depending on both the morphological complexity of the language and the size of the training corpus. Syntactico-semantic features are also open ended. For case, person,[3] number, gender, tense, aspect and mood, we[4] compiled a list of 99 common values from the Leipzig Glossing Rules (Bickel et al., 2008), the ODIN corpus (Xia et al., 2016), Unimorph (Sylak-Glassman et al., 2015) and our own observation. MOM adds these features to the morphological rules and BASIL defines them in the specification if they are found in the data.

[3]The Grammar Matrix accepts only a closed set of 9 values for person. However, I group these with the other syntactico-semantic features because there is variation in the way they are glossed in IGT data.

[4]This list comes from joint work with Olga Zamaraeva.

The final grammar specification includes definitions for syntactic phenomena, syntactico-semantic features, lexical entries and morphological rules, using both fixed and open-ended features. This grammar specification can be input to the Grammar Matrix customization system to produce a grammar with lexical entries and lexical and phrase-structure rules that model the phenomena defined in the specification. The remainder of this chapter focuses on how BASIL infers the appropriate specifications for a particular language, based on IGT data.

4.2 The Lexicon

The most accurate and fully detailed typological specification cannot produce a working grammar without a lexicon. At the same time, decent coverage over unseen texts for languages with any morphological complexity requires a lexicon built in terms of lexical entries for roots plus some model of morphological processes. The Grammar Matrix customization system elicits, as part of its input grammar specifications, descriptions of lexical classes and lexical rules. In this section, I describe lexical class specifications and how I infer them, leaving the inference of morphological rules to Section 4.3.

In brief, a lexical class is defined in terms of its part of speech, any features specific to the class and a set of lexical entries, which give the orthographic representations and semantic predicate symbols[5] for entries in that class. As an example, Figure 4.1 illustrates a lexical class for nouns in Meithei [mni].

The Grammar Matrix customization system interface provides for nouns (which may be identified as pronouns), intransitive verbs, transitive verbs, clausal complement verbs, auxiliaries, copulas, determiners, case-marking adpositions, and adjectives in its lexicon section. In addition, sections for particular syntactic phenomena allow for the definition of lexical entries for such items as conjunctions, subordinating conjunctions, complementizers, and

[5]I use the Grammar Matrix convention for predicate symbols, which includes an English word of similar sense followed by the part of speech. I do not assume that English is the best way to capture meaning cross-linguistically, however, this convention is useful for evaluation.

```
section=lexicon
  noun1_name=noun1
    noun1_feat1_name=person
    noun1_feat1_value=3rd
    noun1_det=opt
      noun1_stem1_orth=kekrú
      noun1_stem1_pred=_blackberry_n_rel
      noun1_stem2_orth=khoy
      noun1_stem2_pred=_bee_n_rel
```

Figure 4.1: An example of a lexical entry for a noun in an inferred grammar specification file for Meithei [mni]

negation adverbs. It is important to note that this classification of basic types of words brings with it a set of assumptions about what word classes exist in the world's languages and, for example, that nouns and verbs are distinct cross-linguistically. On the other hand, recent work using this formalism to model Lushootseed, a language sometimes claimed to not have a noun/verb distinction, shows that even languages with apparent category flexibility can be fruitfully analyzed in this way (Crowgey, 2019). Therefore, I make no claims regarding the actual parts of speech of the lexical items MOM (Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017) and BASIL infer, but attempt to model these words effectively in the resulting grammar.

BASIL infers only a subset of the lexical categories supported by the Grammar Matrix, which are shown in in Figure 4.2. In this section, I describe the process of extracting these definitions from the IGT corpus, with a focus on nouns and verbs and their subcategorization. I leave the remaining lexical types supported by the Grammar Matrix to future work.

4.2.1 Noun and verb extraction

At the highest level of abstraction, lexical inference involves the definition of classes of words and the allocation of words to classes. The goal of noun and verb extraction is to define lexical classes and lexical entries for these major parts of speech that properly model the

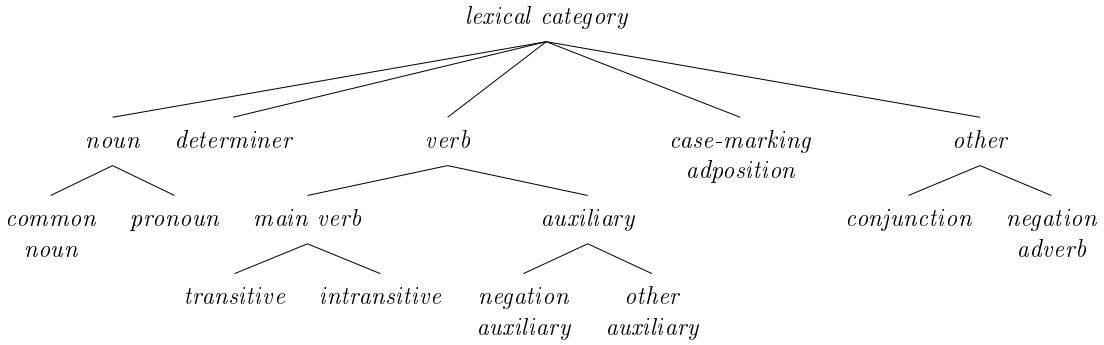


Figure 4.2: A overview of the lexical categories that BASIL infers, organized according to the inference process, as described in this chapter

morphotactic patterns for each class as well as the syntactic behavior. The lexicon portion of the grammar specification is rather open ended in that it is possible to create any number of lexical classes and those classes can have any number of lexical entries and features. Nevertheless, a number of features in the lexicon have a fixed set of possible values. Nouns have two features that are accounted for with a fixed set of values. Pronouns are indicated with a feature that has a single value[6] and whether the noun takes a determiner can be defined as optional, obligatory or impossible. Verbs have one feature with fixed values: the verb's valence can be intransitive with case assigned by the over-arching case system, transitive with case assigned by the over-arching case system, intransitive with case underspecified (in other words with no case constraints) or transitive with case underspecified. BASIL infers two of the four possible values for nouns as well as two of the four values for verbs.

The MOM morphological inference system (Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017) extracts a lexicon of nouns and verbs based on the POS tags in the corpus. For each noun or verb in the training data, MOM adds a node to the graph for each morpheme

[6]The pronoun features is only used by the adnominal possession library (Nielsen, 2018; Nielsen and Bender, 2018) in the Grammar Matrix customization system, which is not covered by BASIL. Nevertheless, I include this feature in the specifications because BASIL defines nouns and pronouns differently so it already does the work of differentiating between the two. Furthermore, this features will be useful for future work inferring adnominal posession.

and connects them with edges that represent their input and output relationships. This is done while simultaneously extracting each affix in the corpus, along with its inputs, which I will describe in Section 4.3. Nodes with overlapping edges (stems and affixes that are input to the same affixes) are merged based on a user-specified overlap value, such that two verb stems that are inputs to overlapping affixes are merged into a single lexical class.[7,8] MOM defines a lexical class for each lexical node which includes any features or other specifications associated with that class, and then adds the orthography and predication for each stem in the lexical class. The resulting lexicon comprises one or more lexical class for nouns and one or more lexical class for verbs, where the lexical classes are defined based on morphotactic patterns and the lexical entries for each class include an orthography and a predication, as in Figures 4.1 and 4.3.

4.2.2 Noun and verb subcategorization

Whereas lexical classes that are defined based on their morphotactic patterns are crucial for a grammar to analyze individual words, definitions of the classes' syntactic properties are necessary to analyze larger constituents such as phrases or sentences.

Transitive and intransitive verbs

The grammar specification file allows the definition of verbs based on their valence and the cases of their core arguments. For example, if a certain verb requires a subject with ergative case and a complement with absolutive case, the grammar specification file must include this information in the verb's definition. Figure 4.3 shows the definition of the verb class for Lezgi [lez] which includes a valence (transitive) and two features, each with a name (case), value (ergative or absolutive) and the argument or head the feature is associated with (the subject or object).

---

[7] Affixes go through a similar merging process, which I describe in Section 4.3.

[8] A detailed description of the process of creating and merging nodes is provided by Zamaraeva (2016).

```
verb1_name=verb1
  verb1_feat1_name=case
  verb1_feat1_value=erg
  verb1_feat1_head=subj
  verb1_feat2_name=case
  verb1_feat2_value=abs
  verb1_feat2_head=obj
  verb1_valence=trans
    verb1_stem1_orth=тамам
    verb1_stem1_pred=_finish_v_rel
    verb1_stem2_orth=чуьнуьх
    verb1_stem2_pred=_steals_v_rel
```

Figure 4.3: An example of a lexical entry for a transitive verb in an inferred grammar specification file for Lezgi [lez]

In order to achieve this result, syntactic inference for lexical items can be done before morphological analysis and included in MOM's input, or it can be performed on the output. In Zamaraeva et al. 2019a, we took the former approach to subcategorize verbs based on their valence properties, by inferring verbal case frame and passing this information to the morphological inference system. Due to this work, MOM does not merge verbs with different transitivity and case-frames, so the lexicon it produces includes separate classes for transitive and intransitive verbs as well as separate classes for verbs with quirky case, or case frames that differ from the over-arching case system. While the Grammar Matrix provides four options for the valence feature (intransitive with case assigned by the over-arching case system, transitive with case assigned by the over-arching case system, intransitive with case underspecified or transitive with case underspecified), it is possible to achieve the same result using only two of these options, which MOM does for the sake of simplicity. While the options to assign case automatically constrain the arguments' case based on the over-arching case system (e.g. verb1_valence=erg-abs would be the Matrix's way of specifying default case for a transitive verb an ergative-absolutive language), it is equivalent to specify transitive valence and then specify the subject and object cases individually as in Figure 4.3. Thus MOM and

BASIL produces grammar specifications with only two of the four possible valence values. These verb classes are subcategorized based on their morphotactics to produce a lexicon that captures the morphotactic and syntactic characteristics of verbs in the language.

Pronouns and other nouns

The definitions for pronouns also differ from those for other nouns in that while common nouns and proper nouns are third person, pronouns can vary in person, number and gender[9] features. Therefore, I account for pronouns separately from other nouns, but I take a different approach than Zamaraeva et al. (2019a). In this case, BASIL takes the lexical classes in MOM's output and divides them based on syntactic inference.

BASIL infers pronouns from the noun lexical items in MOM's output based on their predications (which MOM constructs from the gloss). Figure 4.1 includes two example predications: _blackberry_n_rel and _bee_n_rel. A pronoun in the MOM output might have the predication _she_n_rel or _3sg_n_rel. BASIL checks the nouns in MOM's lexicon and if it identifies a predication that includes an English pronoun or person, number or gender (PNG) features without an accompanying lemma,[10] it removes the lexical entry from the lexical class defined by MOM and creates a new lexical class which includes any PNG or case features found in the original predication, the DELPH-IN-style predication for pronouns _pron_n_rel, and the pronoun feature. To ensure that these pronouns can inflect normally, BASIL adds the pronoun lexical classes to the input list of the morphological rules that their original lexical classes were inputs to. I will provide more detail on morphological rules and their inputs in Section 4.3.

Finally, after creating separate lexical classes for the pronouns, I assume that the remaining classes are either common or proper nouns and put a third person feature on those classes.

---

[9] BASIL does not currently infer inherent gender on nouns other than pronouns.

[10] The current MOM system constructs predications from the lemma in the gloss or if there is no lemma, it uses the grams. Therefore there should not be any predications in MOM's output that contain a lemma and features, but BASIL checks anyway.

For all noun classes, I add the value optional for the determiner feature. Although impossible and obligatory are also possible values for this feature, using optional over obligatory improves coverage for sentences containing, for example, proper nouns and using optional over impossible does not have an impact on coverage.[11] Future work could consider identifying classes of nouns which always co-occur with determiners and making the determiner obligatory in those cases as well as identifying classes of nouns which never co-occur with determiners and making them impossible.

Auxiliaries

The last type of subcategorization that I do is separating auxiliaries from the main verb classes. The Grammar Matrix defines auxiliaries in a separate portion of the lexicon section from verbs and requires different features and values in the class definitions. There are two features with fixed values for auxiliaries: a semantics feature which indicates whether or not the auxiliary adds a predication and a subject feature which has four possible values regarding the case and head type of the subject.

In addition to the auxiliary lexical items themselves, the grammar specification requires information regarding the auxiliaries' syntactic distribution, such as what kind of constituent they attach to, and whether they occur before or after that constituent. To infer this information, BASIL must identify auxiliaries from the source IGT, and I will describe this process in Section 4.5.1. After auxiliary inference, BASIL has a list of auxiliaries which includes their orthographies, subject specifications and any morpho-semantic features. BASIL extracts the auxiliaries from the verb lexicon in MOM's output (illustrated in Figure 4.3), using the same process as for pronouns: BASIL searches MOM's verb lexicon for the orthographies in the auxiliary list, and for each one, it defines a new auxiliary class with the orthography of the auxiliary and adds the appropriate semantics and subject value, according to auxiliary inference (§4.5.1), which targets four of the six possible values for these features. BASIL adds this

[11] This decision has the potential to lead to over-generation when attempting to generate strings from the semantic form. However, generation is not the focus of this dissertation.

new auxiliary class to the inputs of the same morphological position classes as the verb class it was extracted from. Because auxiliaries are often homophonous with main verbs, BASIL does not remove the main verb lexical entry.

4.2.3 Additional lexical items

In addition to the noun, verb and auxiliary lexical categories described above, BASIL also infers determiners, case-marking adpositions, negation adverbs and coordinators. The MOM morphotactic inference system (Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017) was designed specifically to define lexical classes and morphological rules for nouns and verbs. Although the Grammar Matrix supports morphological inflection for determiners, it does not for adpositions or negation and coordination particles. Because there is relatively little to gain from modeling determiners as involving extensive productive morphology and the remaining lexical classes don't have the option, BASIL infers these lexical classes by extracting full form orthographic representations and PNG and case features from the IGT corpus and defines them in the grammar specification.

Determiners

The grammar specification for determiners only requires the orthographic representation, semantic predication and any PNG or case features. Taking advantage of the widespread use of determiners in English, BASIL uses the projected dependencies to check each noun in the corpus for a determiner dependency or a determiner POS tag. It verifies that the words with these projected tags are determiners based on the assumption that determiners are glossed either with English determiner lemmas (such as 'the', 'a', etc) or with only grams and no lemmas (for example DEF.SG.NOM). If either of these conditions are met, BASIL creates a lexical class with the orthography, the predication _exist_q_rel,[12] and any PNG

[12]This predication is a Matrix convention for existential quantifiers. Future work should add different predications for quantifiers meaning 'no', 'every' and 'each' as well as cognitive status features for quantifiers that indicate definiteness.

or case features in the gloss. Currently, determiner inference only extracts determiners that are glossed with only PNG and case grams or with 'the', 'a', 'this', 'that', 'these', 'those', 'no', 'every', 'each'. Additional types of determiners and specifiers such as other quantifiers and question words (e.g. 'which') are left to future work.

Case-marking adpositions

Case-marking adpositions have two features with fixed values in the specification file: First a boolean feature for whether the adposition is optional or required and second, the order feature indicates whether the adposition occurs before or after the noun phrase it attaches to. BASIL identifies adpositions based on the part-of-speech tags and checks for a case gram in the gloss to determine if it is a case-marking adposition. If it is, BASIL looks for the nearest noun to see if it is before or after the adposition. After collecting all of the case-marking adpositions in the corpus, and recording their position with respect to the nearest noun, BASIL creates a lexical class for each orthography identified as an adposition, specifying each one as optional and indicating the order that was most common for that orthography in the corpus. To the lexical class, BASIL also adds the case feature that the word was glossed with.[13]

My inferred grammars also support negation and coordination particles. Because these are not included in the lexicon section of the Grammar Matrix grammar specification, but instead defines them with the other details of these syntactic phenomena, I will describe them in their respective subsections in Section 4.5.

4.2.4 Summary

In this section I described the process of inferring lexical items from IGT data and defining them in the grammar specification. I use the MOM morphotactic inference system (Wax,

[13]If more that one case was found, e.g. if the word was glossed sometimes with ergative and sometimes with nominative, BASIL adds both cases to the specification and the Grammar Matrix customization system will create a disjunctive type which allows the adposition to have either case.

2014; Zamaraeva, 2016; Zamaraeva et al., 2017) for noun and verb inference, defining lexical classes based on the morphotactic graph it infers, which I will describe with respect to the lexical rules in the next section. For nouns and verbs, I do additional work to further subcategorize in terms of their syntactic properties. I then infer determiners and case-marking adpositions based on their POS tags and their glosses and add their full-form (non-inflecting) orthographies to the lexicon. The lexicon interacts closely with both morphological and syntactic inference, which I will describe next.

4.3 Morphotactics

The morphological component of a machine-readable grammar ultimately needs to account for which morphemes can co-occur and in which order, what the syntactic and semantic contributions of each morpheme are, and the morpho-phonological processes that relate the actual word forms to the collection of morphemes that make them up. The Grammar Matrix customization system abstracts away from the last of these, assuming that the generated grammars will be interfaced with an external morpho-phonological analyzer (Bender and Good, 2005; Bender et al., 2010). Accordingly, this inference system is only concerned with morpheme order, co-occurence, and syntactico-semantic contributions.

The Grammar Matrix grammar specification files handle morpheme co-occurrence in terms of position classes (PCs) each of which specify what they can attach to (their 'input'), whether they are prefixes or suffixes, and which lexical rules they house (Goodman and Bender, 2010). The lexical rules are defined in terms of lexical rule types (LRTs) which bear morpho-syntactic or morpho-semantic features. These LRTs are instantiated by lexical rule instances (LRIs), which have specific affix spellings or are flagged as zero affixes or non-inflecting rules. An example of the specification for a position class in Meithei [mni] is shown in Figure 4.4.

For the most part, the morphology section of the specification file is open ended in that it doesn't have to contain any PCs at all and at the same time can include arbitrarily many. However, within the PCs there are some requirements. Each PC must have at least one

```
section=morphology
  noun-pc1_name=noun-pc1
  noun-pc1_order=suffix
  noun-pc1_inputs=noun1
    noun-pc1_lrt1_name=noun-pc1_lrt1
      noun-pc1_lrt1_feat1_name=case
      noun-pc1_lrt1_feat1_value=nom
      noun-pc1_lrt1_lri1_inflecting=yes
      noun-pc1_lrt1_lri1_orth=-pə
```

Figure 4.4: An example of a position class from an inferred grammar of Meithei [mni]

input (a lexical class or another PC) and a position (prefix or suffix)[14] and can be optional or obligatory. Each PC must also have one or more LRTs, which can specify features on the input or on the arguments of their inputs. Each LRT must have one or more LRIs, which includes an orthographic form if it is inflecting. While the orthographies and features are open as we saw for lexical items in the lexicon, there are three features with a total of six fixed values in the morphology: The position feature has the value prefix or suffix, the obligatory feature has only a single value and can be included or not, and the inflecting feature whose values are affix or no affix. Using MOM for morphotactic analysis as described in this section, BASIL targets all six of these possible values.

As I did for the lexicon, I use the MOM morphotactic inference system to infer the morphological rules of the language. MOM (Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017) infers a graph of the morphemes in the language by collecting the affixes for each word with a noun or verb POS tag. Using the same approach described for stems in Section 4.2, it creates a PC with an LRT which includes any features found in the gloss and an LRI with the appropriate orthographic representation. After creating a PC for each morpheme in the corpus, it merges PCs that have overlapping inputs, based on a user-specified overlap value.

[14]The Grammar Matrix does not handle circumfixes or infixes as separate types of morphemes. Circumfixes can be specified as pairs of individual prefixes and suffixes, while we assume that infixes are regularlized to prefixes or suffixes by an external morphophonological component.

While the morphotactic graph is essential for processing individual words, the morpho-syntactic or morpho-semantic features on those morphemes are key to producing the correct parse for larger phrases and sentences. In order to collect these features, MOM uses a feature dictionary which contains a large number of known glosses, grouped by their type, to map common grams to features.[15] For example, the grams 'IPFV', 'IMPFV' and 'IMPERF' are all mapped to imperfective aspect. When MOM constructs the lexical rule types, it checks the morpheme's gloss for any PNG,[16] TAM or CASE grams and adds them to the lexical rule.

Non-inflecting lexical rules pose a particular challenge because they are not typically glossed as separate morphemes in IGT but rather indicated with a gram attached to the previous element with a "." (when they are indicated at all).[17] MOM only creates non-inflecting rules for glosses it is able to map to PNG, case or TAM features, and only when such a gloss is found attached to the gloss for a stem. For example, if a noun is glossed as 'dog.NOM', MOM creates a non-inflecting lexical rule to add nominative case. All PCs which contain a non-inflecting LRI are made obligatory, so that forms without overt affixes do not end up only optionally bearing the features associated with that part of the paradigm which would increase spurious ambiguity in the grammar.

The result of morphological inference with MOM is a set of position classes based on the position classes that they can be input to. Within those position classes are lexical rules that contribute features and lexical rule instances, which either correspond to a particular orthography or are non-inflecting. Both the morphological rules in this section and the lexical entries in Section 4.2 contain morpho-syntactic features which interact with the syntactic inference in Section 4.5. The next section is concerned with how I define those features in the grammar specification, so that they will interact properly in the resulting grammars.

---

[15] Collecting the initial set of grams and adding features to lexical rules is unpublished work by Olga Zamaraeva.

[16] For more information on how the argument (subject or object) associated with agreement features on verbs is inferred, see Section 4.5.3.

[17] Exhaustive glossing of zero-marked features is difficult, and field linguists frequently and reasonably decide not to gloss them.

4.4 Syntactico-semantic Features

A great deal of semantic information is expressed morphologically in the form of person, number and gender (PNG) marking on nouns or agreement on verbs and tense, aspect and mood (TAM) inflection on verbs and auxiliaries. These PNG and TAM features are both reflected in the semantics and can function in the syntax. For example the agreement marking on the verb must be compatible with the PNG features of its arguments.

In order to model these features, the grammar specification must contain two types of definitions: First, the features themselves must be defined as belonging to the appropriate PNG or TAM category; and second, they must be associated with the appropriate lexical entries or morphological rules as described in Sections 4.2 and 4.3. This section focuses on the first set of definitions so that the syntactic constraints contributed by these features can be used in the grammar and so that their semantic contributions will be reflected in the semantic representations.

The syntactico-semantic features are only used by the inferred grammars if they are part of a lexical item or morphological rule. For this reason, BASIL identifies these features from the lexical and morphological specifications, rather than from the IGT corpus directly. PNG features can be expressed on nouns or on verbs, auxiliaries and determiners in the form of agreement with a nominal argument, while TAM features are expressed on verbs and auxiliaries. For each of the features described in this section, BASIL starts with a collection of features collected from these lexical items and their morphological rules.

4.4.1 Person, number and gender

PNG features reflect information about the person, number and gender of an entity in a discourse context and can be inherent to nouns or marked morphologically on nouns, verbs and determiners. These features are often glossed in the IGT, as illustrated in the Wambaya [wmb] sentence in (4). Here the auxiliary *nga* is marked for the first person and singular

number of the agent (via the gloss 1.SG.A) as well as past tense (via the gloss PST).[18] *Marnugujama* 'conkerberry' is glossed with class III gender (or vegetable gender) and *alagulija* 'child' is inflected with dual number (DU).

(4) Yanybi nga marnugujama alagulija.
Yanybi ng-a marnugujama alag-uli-ja
get 1.SG.A-PST conkerberry.III.ACC child-DU-DAT
'I got the conkerberries for the two children.' [wmb] (Nordlinger, 1998)

My goal is for the resulting grammar to encode all of this information in the semantic representations, and for the appropriate agreement to be enforced, such that *marnugujama* and *alagulija* could not be analyzed as the subject because they are not compatible with first person agreement. Because MOM (Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017) puts the features found in the glosses on the lexical items and morphological rules, the remaining task is to define those features in the grammar specification as belonging to the person, number and gender categories.

Person

Generally speaking, person is a feature that marks the entities in an utterance with respect to discourse participants (Siewierska, 2004), where *first* is the speaker, *second* is the addressee and *third* is someone or something outside of the discourse context. Combinations of these persons, such as *first* + *second* 'I and you' and *first* + *third* 'I and they' are also possible and are often referred to as *inclusive* and *exclusive* (Cysouw, 2013). The Grammar Matrix's library for person (Drellishak, 2009) provides a set of six options for person distinction: first, second, third; first, second, third and fourth; first and non-first; second and non-second; third and non-third; and none. It also allows three options with regard to subtypes in the

[18]Nordlinger (1998) analyzes this as a second position clitic cluster. Bender (2008b) models the second position clitic cluster as a second position auxiliary in the context of the Grammar Matrix.

first person: none, inclusive vs. exclusive[19] and other, where subtypes not modeled by an inclusive/exclusive distinction can be defined.

After collecting all of the person features from the lexical items and morphological rules, BASIL posits that the language contains first, second, third and fourth person if it found 4th person; first, second and third person if it found 3rd and either 1st or 2nd; and then first and non-first if it found 1st; second and non-second if it found 2nd; third and non-third if it found third; and otherwise none. BASIL then checks for inclusive and exclusive features and if it finds any, it defines an inclusive/exclusive distinction,[20,21] and otherwise it posits that there is no distinction. The Grammar Matrix allows the definition of additional subtypes (other than inclusive/exclusive); however, I leave the addition of additional person distinctions to future work. Thus BASIL can infer eight of the nine possible values for this section of the grammar specification.

Number

Number indicates how many entities are being referred to. This distinction can be as simple as singular vs. plural or may be more modular distinguishing dual (two), paucal (a few) and other numbers of entities (Corbett, 2000). The numbers distinguished by a language vary cross-linguistically and it is possible for these features to form a hierarchy (for example non-singular might subsume dual and plural). Thus, the Grammar Matrix allows number features to be freely added to the specification file, forming a hierarchy if desired (Drellishak, 2009).

---

[19]Here the specification also requires the number categories in which this distinction applies.

[20]Because the Matrix requires that the numbers in which this distinction is applicable be listed, BASIL adds all non-singular numbers it defines to this list. This does not involve any inference, as allowing this distinction for numbers where it doesn't apply will not affect the coverage of the inferred grammars if those combinations are not defined on any lexical entries or morphological rules.

[21]When an inclusive/exclusive distinction is added to the specification, the Grammar Matrix customization system automatically creates a feature called PERNUM, which bundles person and number. Because the morphological inference is done before person inference, BASIL does additional work to rename any person or number features in the morphology or lexicon sections with the corresponding pernum feature, if the language was found to have a inclusive/exclusive distinction.

BASIL defines a number value for each of the numbers found in the morphology and lexicon. Currently, it defines each of these as sister types, rather than inferring a hierarchy of supertypes and subtypes, which I leave to future work.

Gender

Gender is another fairly open-ended category in the world's languages. While some languages like Russian distinguish just masculine, feminine and neuter, Bantu languages such as Kiswahili distinguish a complex system of genders (Corbett, 1991). Linguists also vary in their annotation of gender features either using grams like M or MASC or using numerals for more complex systems. For example, while Wambaya has only four genders (masculine, feminine, vegetable and neuter), Nordlinger (1998) uses a numeric system for glossing as seen in (4). To accommodate this flexibility in the gender distinctions in language and linguists' annotation preferences, the Grammar Matrix allows the addition of any number of genders by any name, and also allows for the specification of a hierarchy (e.g. to support agreement markers that are ambiguous between two or more gender values). As with number, BASIL defines a gender value for each of the genders found in the morphology and lexicon, but does not infer a hierarchy.

4.4.2 Tense, aspect and mood

Every language has some grammatical expression of time, which falls into the categories of tense, aspect and/or mood, and these features can be marked either morphologically on the verb, with an auxiliary or morphologically on an auxiliary, and a single utterance may include a combination of these expressions (Hopper, 1982). In the IGT from Matsigenka [mcb] in (5), the verb *oataira* is marked with regressive aspect (REG) and realis mood (REALIS), while the verb *oponiakara* is marked with perfective (PERF) aspect and realis mood (REALIS). Michael (2008) characterizes the regressive aspect as a subtype of perfective aspect that indicates motion back to a salient point of origin.

(5) ovashi oataira oponiakara.
ovashi o-a-t-a-i=ra o-poni-ak-a=ra
so 3fS-go-EPC-REG-REALIS=SUB 3fS-come.from-PERF-REALIS.REFL=SUB
'Then she went back to where she came from.' [mcb] (Michael et al., 2013)

The TAM categories contain a number of possible values cross-linguistically and, as illustrated by the regressive aspect in Matsigenka, can form hierarchies. As with the number and gender libraries, the TAM library of the Grammar Matrix (Poulson, 2011) also allows the definition of any values for each tense, aspect and mood and also allows the definition of hierarchies. BASIL defines each TAM feature as either tense, aspect or mood in the respective section of the grammar specification, leaving the inference of hierarchies to future work.

4.4.3 Summary

In this section, I described six categories of syntactico-semantic features: person, number, gender, tense, aspect and mood. These features are added to the specifications of lexical entries or morphological rules according to the methodologies described in Sections 4.2 and 4.3 and defined as belonging to their respective categories as I described in this section. The result of these definitions is a grammar that produces semantic representations that contain this information and enforce agreement between heads and their arguments in the grammar.

4.5 Syntactic Properties

Sections 4.2 and 4.3 briefly described my methodology for inferring lexical and morphological information and Section 4.4 described the processing of defining the syntactico-semantic features on the lexical entries and morphological rules so that they can be accessed by the syntax. In this section, I describe the algorithms used for inferring each of the syntactic phenomena accounted for in my grammars. Using the projected dependency tags provided by INTENT (Georgi, 2016) and typologically-informed heuristics, I make generalizations about the distribution of constituents in the language and use this to infer the appropriate defini-

tions for the grammar specification to model a range of syntactic phenomena. These include both broad-brush, language-level properties (e.g. 'the case alignment is ergative-absolutive') and properties associated with specific constructions (e.g. this form can coordinate VPs in a monosyndetic pattern) and specific lexical items (e.g. negation is marked via an auxiliary with this orthography that combines with a VP and raises the subject).

4.5.1 Word order and auxiliaries

Languages vary in both their degree of word-order flexibility and, if only specific orders are allowed, which ones are (e.g. Dryer, 2011). When linguists talk about the 'word order' of a language, they are frequently referring to the relative order of a verb and its arguments (subject and complement), but there are also cross-linguistic differences in e.g. the order of determiners (if present) with respect to their head nouns, adpositions with respect to NPs, and others. As an example, (6) illustrates the subject object verb (SOV) order of Haiki [yaq].

(6)

Maria	ian	hu'ubwa	ume	toto'im	suataite.
Maria	ian	hu'ubwa	ume	toto'i-m	sua-taite
Maria	now	just	DET.PL	chicken=PL	kill(PL.OBJ.)-INCEP
noun	adv	adv	det	noun	verb[22]
nsubj	advmod	advmod	det	dobj	root

'Maria is just now starting to kill the chickens.' [yaq] (adapted from: Harley, 2019)

The 'word order' section of a Grammar Matrix grammar specification takes information about major constituent (subject and complement) word order, the order of determiners with respect to nouns and the order of auxiliaries with respect to their complements (Bender et al., 2010).

[22]These are a combination of part-of-speech tags from the annotated corpus (provided by the linguist, which are incomplete for this IGT) supplemented with the POS tags provided by INTENT (Georgi, 2016). The projected dependencies shown in IGT throughout this chapter are provided by INTENT (Georgi, 2016).

Basic word order

The Grammar Matrix allows ten values for basic word order: SOV, SVO, OSV, OVS, VSO, VOS, v-final, v-initial, free, finite verb second and non-finite verb clause-finally, and finite verb or auxiliary in second position and otherwise free (Fokkens, 2010). BASIL is designed to infer nine of these ten, mapping both of the finite verb in second position word orders to a single option: finite verb or auxiliary in second position and otherwise free.

The algorithm I use for inferring the basic word order comes from Bender et al. 2013, which I re-implemented to take an enriched Xigt corpus of the type described in Chapter 3 as an input. BASIL first identifies each verb via the POS tags and then uses the projected dependencies to identify the subject (S) and object (O). Thus for the IGT in (6), it identifies the verb *suataite* via the tag verb, the subject *Maria* via the dependency nsubj and the object *toto'im* via the dependency label dobj.[23] Based on which of these constituents are present and their relative order, BASIL categorizes the IGT as one of ten possible observed word order patterns: SOV, SVO, OSV, OVS, VSO, VOS, SV, VS, OV and VO. From these observed patterns, it maps the IGT to one or more compatible binary patterns: SV, VS, OV, VO, SO and OS.[24] Because all three constituents are overt in (6) and their order is subject, object, verb, BASIL would categorize this IGT as SOV and then map that to the binary patterns SV, OV and SO.

Following Bender et al. (2013), BASIL expects sentences from a language that has a canonical or over-arching word order of SOV to exhibit only SO, OV or SV orders, while a canonically free word order language would have an even distribution over all six binary orders. Plotting the observed binary orders observed for each each language in a three dimensional space with one axis for SV to VS, one for OV to VO and one for SO to OS,

[23]Dependencies are in fact not properties of particular items, but relationships between a head and a dependent. In (6), the dependency label is shown on the dependent only, but in the Xigt encoding, the heads are also identified.

[24]The binary patterns are not expected to be observed in the data in isolation, but instead are useful for representing the order of each constituent (subject, object and verb) with respect to each of the other constituents.

BASIL compares the euclidean distance for the observed word orders with the expected word order of each canonical word order (SOV, SVO, OSV, OVS, VSO, VOS, v-initial, v-final and free). BASIL posits that the language has the canonical word order with the shortest distance to the observed word order. If BASIL posited free word order, it then compares the observed ternary patterns to infer whether the languages has verb-second order. If the verb is most often the second element in the observed ternary patterns (in other words SVO and OVS are the most common ternary patterns), BASIL posits V2 word order (specifically the V2 option where the finite verb is in second position and the order is otherwise free), instead of free. I compared this approach with a simpler approach, where the inference system counted the binary orders as described above, but just selected the canonical order compatible with the most common binary orders, and found that the approach of Bender et al. (2013) outperformed the simpler approach on the development languages (which are described in Chapter 5).

Determiner-noun order

The Grammar Matrix uses two features to define the distribution of determiners in the language: First it uses a boolean feature to indicate whether or not the language has determiners as separate words and if they do, the feature for the determiner's order with respect to the noun must be assigned the value before or after. BASIL is designed to infer all four of these values.

As with basic word order, BASIL uses the projected dependencies to collect each noun and determiner pair: in (6) the determiner *ume* would be identified via the det dependency and its head (not shown in this example) would be the noun *toto'im*. If any determiners were found, the feature that indicates if the language has determiners is specified as True. Then BASIL counts the number of observed determiners before vs. after the noun and posits whichever order is most common.

Auxiliaries

Auxiliaries can be characterized as grammaticalized verbs which can express tense, aspect, mood, negative polarity and voice (Anderson, 2006). For the purpose of the inferred grammars, I consider auxiliaries to be a subcategory of verb which either add TAM or PNG agreement features or contribute a modal or negative predication. An example of a negation auxiliary from Matsigenka [mcb] is shown in (7). Here the auxiliary *gara* contributes negation and irrealis mood to sentence.[25]

(7) Maika gara pipokai, shintsi nontsonkatakero.
maika ga pi-pok-a-i shintsi no-n-tsonka-t-ak-i=ro
now NEG.IRREAL 2S-come-REG-REALIS fast 1S-IRREALIS-finish-EPC-PERF-REALIS=3fO
'Don't come again (i.e. today, to the place where the man is felling the tree), I'm going to finish it quickly.' [mcb] (Michael et al., 2013)

The Grammar Matrix requires the following specifications for auxiliaries: a boolean feature indicates whether or not there are auxiliaries in the language; the type of complement must be specified as a verb (V), verb phrase (VP) or sentence (S); and the order of the auxiliary with respect to this complement must be before or after. BASIL infers all of these values. In addition, it infers the values required for the lexical entry described in Section 4.2.2, which indicate whether or not the auxiliary contributes a semantic predication and what kind of subject it requires as follows.

BASIL identifies auxiliaries in the corpus by looking for words glossed with an English auxiliary or modal or words whose gloss contains grams for features, but no lemma. While collecting auxiliaries from the corpus based on this heuristic, BASIL also identifies the main verb and its subject and object from the projected dependencies. BASIL uses the main verb and its arguments to infer what type of constituent the auxiliary attaches to and whether

[25]Michael et al. (2013) use the POS tag part (for particle) for this word. However, in order to model this in an HPSG grammar, it is necessary to model it using a lexical type defined in the grammar, such as auxiliary.

it occurs before or after that constituent. More specifically, it checks whether an auxiliary occurs before or after the verb and looks for auxiliaries intervening between the verb and its subject, which would indicate that the auxiliary takes a VP complement instead of an S, or intervening between a verb and its object, which would indicate that the auxiliary attaches to a V, rather than a VP. In the example of (7), BASIL would identify the auxiliary *gara* from its gloss which does not contain a lemma. Because *gara* occurs before the verb *pipokai*, BASIL records its position as before. However, in the absence of overt arguments, it cannot make any judgments as to the type of constituent this auxiliary attaches to. After making this judgement for all auxiliaries in the corpus, it will posit V or S attachment if it found evidence for either. Because the argument-composition analysis that the Grammar Matrix uses to model auxiliaries with V complements is computationally very expensive (see Bender 2010), BASIL posits V attachment only if it finds explicit evidence (i.e. an auxiliary between the verb and its object). Otherwise it will err on the side of assuming VP attachment which results in a more efficient output grammar. BASIL also posits S attachment only if there is evidence (i.e. a subject between an auxiliary and the verb), and otherwise posits VP attachment, based on the hypothesis that S attaching auxiliaries are relatively rare.

After identifying the auxiliaries in the corpus, BASIL considers a post-hoc change to the main word-order, which was already inferred, to account for second position clitic clusters. Clitics are syntactically independent but phonetically dependent words (Zwicky and Pullum, 1983), and while the Grammar Matrix does not specifically account for second position clitics, which are not uncommon in the world's languages, it does support an analysis (set forth by Bender (2008b) for Wambaya) of second position clitics/clitic clusters as auxiliaries in a V2 language, when those clitics express TAM or agreement features. Clitic clusters that contain PNG agreement and TAM information are picked up by our auxiliary inference algorithm and if they occur overwhelmingly as the second word of each sentence, BASIL posits V2 word order for the language to leverage this analysis.

Finally, in addition to specifying the auxiliary's order with respect to the main verb, the lexical entry must also include a value for the case of its subject. As noted in Section 4.2.2,

the Grammar Matrix requires values for the features that indicate the semantic contribution of the auxiliary and the case of the subject. When BASIL constructs the auxiliary lexical items from verb lexical items inferred by MOM (Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017), it will set the boolean semantic predication feature to true and add the predication value from the verb, if the original verb has a predication that contains an English lemma (e.g. _should_v_rel), rather than just features. It will also add a negation predication if the auxiliary contributes negation (see Section 4.5.4 for a description of inferring negation).

The subject feature has four possible values: a noun phrase without case restrictions, a noun phrase bearing the case the verbal complement assigns to its subject, a noun phrase receiving a specific case from its auxiliary, and an adpositional phrase. The adpositional phrase option does not seem to be fully implemented in the Grammar Matrix, while the specifications indicating noun phrases do in fact allow adpositions,[26] so I do not infer adpositional subjects. After some testing, checking for a different subject case when a verb co-occurs with the auxiliary, rather than with the verb alone, I found that this inference is subject to confounding by other factors that can affect the subject's case, so I removed this from the inference system and leave a more accurate inference algorithm to future work. Thus BASIL posits a noun phrase without case restrictions if A) the language does not have a case system or B) the auxiliary always occurs with a different case than the one inferred for the verb's case frame.[27] Otherwise BASIL posits that the auxiliary takes a noun phrase with the case restrictions assigned by the main verb.

4.5.2 Case system and case frame

A language which marks case has variations in the form of the noun phrases correlated with their function in the sentence (Comrie, 1989; Dixon, 1994). A typical case system will

[26] In the Grammar Matrix customization code, this is accomplished through the use of a disjunctive type called +np, which subsumes the types noun and adp.

[27] This leads to some ambiguity, but avoids the loss in coverage that results from positing a case that was assigned due to other syntactic factors.

involve both the case required for core arguments of typical verbs, as well as additional cases used when NPs function as modifiers (e.g. locative case) and sometimes selected for idiosyncratically by specific verbs. Case systems are differentiated according to the alignment they provide for the core arguments of intransitive and transitive verbs. The Grammar Matrix's case library (Drellishak, 2009) requires a specification for case-system from a set of nine values: nominative-accusative, ergative-absolutive, split-ergative, tripartite, split-s, fluid-s, split conditioned on features of the noun phrase arguments, split conditioned on features of the verb, focus-case and none. It also allows the definition of any number of additional cases. The definition for each verb in lexicon requires a specification for the case of the core arguments, which can default to the over-arching case frame, be specified as a different case, or be left underspecified, indicating that the verb has no case requirements. Because most of these case systems require nuanced information to be modeled properly,[28] BASIL infers only three of the nine possible over-arching case systems but leverages the option to define case frames for individual verbs by inferring a case frame for each verb in the lexicon.

To infer the over-arching case system, I use an algorithm developed by Bender et al. (2013) and re-implemented by Howell et al. (2017) to take an enriched Xigt corpus as input. In Howell et al. 2017, we considered two approaches, one of which compares the distribution of grams on arguments throughout the corpus and one that simply uses a heuristic based on the total counts of known case grams in the data. For BASIL, I adopt the latter approach as it outperforms the former both in Howell et al. 2017, which considered 31 languages, and on my development languages.

The approach is simple: if only nominative and accusative grams are found in the corpus, the inferred case system is nominative-accusative; if only ergative and absolutive grams are found in the corpus, the inferred case system is ergative-absolutive; if nominative and/or accusitive grams and ergative and/or absolutive grams are found, the inferred case system

[28]For example defining a case system that is split on features of the verb would require BASIL to identify the nature of this split, including which verbal features (e.g. specific tenses or apstects) govern this split and then define the case restrictions associated with those verbal feature in the morphology.

```
verb1_name=verb1
  verb1_valence=erg-abs
    verb1_stem1_orth=guDaqy
    verb1_stem1_pred=_reveal_v_rel
```

Figure 4.5: An example of a lexical entry for a transitive verb with transitive, ergative-absolutive valence (for Tsova-tush [bbl])

```
verb1_name=verb1
  verb1_valence=trans
  verb1_feat1_name=case
  verb1_feat1_value=erg
  verb1_feat1_head=subj
  verb1_feat2_name=case
  verb1_feat2_value=abs
  verb1_feat2_head=obj
    verb1_stem1_orth=guDaqy
    verb1_stem1_pred=_reveal_v_rel
```

Figure 4.6: An example of a lexical entry for a transitive verb with transitive valence and case specified as features on the arguments (for Tsova-tush [bbl])

is split-ergative; and finally, if none of these grams are found, the inferred case system is none. This approach infers four case systems, but because I leave the inference of the nature of the split in a split-ergative system to future work, BASIL maps it to ergative-absolutive. In addition to defining the over-arching case system, BASIL collects all case grams in the corpus and adds them to the case section of the grammar specification.[29] This allows BASIL to correctly handle verbs that require alternate case frames, as described below.

The Grammar Matrix allows verbs in the grammar specification to have a case frame that

[29]The Grammar Matrix only allows the definition of case features if an over-arching case system other than 'none' is specified. For this reason, if BASIL did not identify a case system (e.g. because no nominative, accusative, genitive or ergative grams were found in the corpus), but other case grams (e.g. locative) were found, it will change the over-arching case system to nominative-accusative. Without adding nominative or accusative to the lexical definitions of any nouns or verbs, this will have no effect on the performance of the grammar, except to allow any case grams that were found to be specified.

uses the cases determined by the over-arching case system, indicated by valence=erg-abs in Figure 4.5, or an underspecified case frame, indicated by valence=trans in Figure 4.6. However, it is possible to specify additional features on a verb's arguments, as shown by features one and two in Figure 4.6. When the Grammar Matrix customization system generates customized grammars, the lexical types produced from the specifications in Figures 4.5 and 4.6, are equivalent. Because BASIL checks each verb to see whether it has a case-frame according to the over-arching system or quirky case, it defines all lexical types the same way, adopting the convention shown in Figure 4.6.

To find the case frame for each verb in the corpus, BASIL guesses the valence of the verb in the source language based on the valence of the corresponding English verb in the translation. More specifically, it uses the dependency parse of the English sentence to identify any verbs that have no direct object or one direct object, skipping any verbs that are passive or have an indirect object or clausal complement. Example (8) from Tsova-Tush [bbl], an ergative-absolutive language,[30] includes a verb with one direct object, identified via the dependency dobj and a subject, identified with the dependency nsubj. BASIL finds the case of those arguments from the gloss line, erg for the subject *meč'ares* in the example. If no case gram is found in the gloss, as is the case for the object *magnit'opona* in (8), BASIL leaves the case underspecified (in case it is found glossed on another example with the same verb). If a case gram was found, BASIL posits the attested case for that verb's arguments, and when this differs from the case of the over-arching case system, this allows the grammar to account for verbs in the language with quirky case.

[30]Hauk (2020) describes the pattern as strictly ergative-absolutive for third person, with an active-stative system for 1st and 2nd person.

(8) meč'ares magnit'opona guyaqi^{n}.
meč'ar-e-s magnit'opon=e guDaqy-in
fisherman-SG-ERG.S cassette.player=and reveal-CM
nsubj dobj root
'The fisherman pulled out a cassette player.' [bbl] (adapted from Hauk, 2016–2019))

This approach to identifying verb case frame differs from the one we used in Zamaraeva et al. 2019a, in that I use projected dependency parses, while we used phrase structure parses in 2019. I find that the dependency parses are more reliable and unlike our previous work, I account for quirky case. I do however, take advantage of our work integrating case frame with morphotactic verb classes, which I described in Section 4.2.3.

The case constraints added in this way to the verb class definitions interact with the case features on noun-phrases when verbs unify with their arguments. These case features may be licensed by the morphological rules on nouns which were inferred by the morphological component described in Section 4.3 or may be contributed to the NP by the determiner or by a case-marking adposition (§4.2.3). If, for example, the lexical rule attaching the accusative case marker to a noun is correctly associated with the feature specification [CASE acc] or if this case feature is contributed by a determiner or adposition, those NPs or APs will be incompatible with argument which require [CASE nom].

4.5.3 Argument optionality

Having leveraged the available inference algorithms and systems for phenomena such as morphotactics, word order and case, I now turn to the entirely new inference modules that I contribute in this dissertation, beginning with argument optionality.

Languages vary in the extent to which and under which conditions they allow dropped arguments: some languages allow core arguments of any verb to be dropped freely, while others are more restrictive if argument dropping is possible at all. These restrictions range from the specific verbs for which argument dropping is allowed, subject vs. non-subject

arguments, specific syntactic contexts (e.g. only in certain tenses), or whether the verb is required to agree with overt vs. dropped arguments (Ackema et al., 2006; Dryer, 2008). Whereas (8) above gave an example of a verb with two overt arguments, the verb in (9) has no overt arguments, but is inflected for agreement with both the subject and object.[31]

(9) oogaigavakari
o-og-a-ig-av-ak-a=ri
3FS-eat-EPV-PL-TRNS-PERF-REALIS.REFL=3MO
root
'She ate them.' [mcb] (adapted from Michael et al., 2013)

The Grammar Matrix accounts for subject and object dropping allowing it to be either lexically licensed (allowed for certain verbs) or possible for any verb (Saleem, 2010; Saleem and Bender, 2010). This is accomplished through two features: one for subject dropping and one for object dropping. The Grammar Matrix also allows argument dropping to be constrained by agreement markers on the verb: agreement on the verb when the subject is overt can be optional, required or not allowed, and similarly when the subject is dropped and when the object is overt or dropped, for a total of 12 values. Finally, subject dropping can be constrained by specific syntactic contexts with two values: the subject can be dropped in all contexts or the subject can be dropped in some contexts.

My inference focuses on determining whether argument dropping is permitted for subjects and objects in a language and leaves constraints on the context to future work. I infer whether agreement is required for dropped vs. overt arguments, which requires differentiating subject agreement markers and object agreement markers. I have not integrated this inference with the morphological rules that license this agreement (for example to require agreement when an argument is dropped versus overt), as the process for constraining morphological rules in the Grammar Matrix to properly require or prohibit dropped versus overt arguments is

[31] I analyze the pronominal clitics in Matsigenka as affixes, rather than independent words, following Inman (2015).

non-trivial. Nevertheless, I provide the inference for whether agreement is optional, required or not allowed for dropped and overt subjects and objects to be built upon in future work. Furthermore, I pass the subject and object agreement markers, which BASIL infers as part of this process, to the morphological inference system so that it can predict which features should be marked on the subject versus object more reliably.

In order to identify whether subject and/or object dropping is possible in the language, BASIL begins by collecting all of the transitive and intransitive verbs in the corpus together with their overt arguments, based on the projected dependencies as it did for case-frame inference (§4.5.2).[32] (8) and (9), repeated below as (10) and (11), provide examples of a verb with two overt arguments and another with no overt arguments, both of which would be collected by BASIL.

(10)

meč'ares	magnit'opona	guyaqi^{n}.
meč'ar-e-s	magnit'opon=e	guDaqy-in
fisherman-SG-ERG.S	cassette.player=and	reveal-CM
nsubj	dobj	root

'The fisherman pulled out a cassette player.' [bbl] (adapted from Hauk, 2016–2019))

(11)

oogaigavakari
o-og-a-ig-av-ak-a=ri
3FS-eat-EPV-PL-TRNS-PERF-REALIS.REFL=3MO
root

'She ate them.' [mcb] (adapted from Michael et al., 2013)

Whereas the case-frame inference methodology determines if a verb is transitive based solely on the presence of an overt object in English translation, here I account for the fact

[32]For more detail on the interraction between transitivity and argument optionality inference, see Section 5.2.

that some English verbs allow object dropping. If the English verb has one direct object, I assume that the verb is transitive, but if it has no object, it is cross-referenced with a list of English object-dropping verbs, assembled from the lexical entries in the English Resource Grammar (ERG v. 1214; Flickinger, 2000, 2011) of the type `v_np*`. If the verb is found in this list, it BASIL posits that the verb is transitive and otherwise posits that the verb is intransitive.

In both (10) and (11) the verbs would be judged as transitive based on the presence of an object in the English translation. After making transitivity judgments based on the English verbs, BASIL uses the projected dependencies to look for overt arguments in the source language by checking for a projected (non-clausal) subject or direct object dependency on the source verb. In (10) it would find both a subject and direct object, whereas in (11) it would find neither. If there is no subject, I assume that the subject has been dropped. If there is no object and BASIL inferred that the verb is transitive, I assume that the object has been dropped.

If BASIL finds any verbs in the corpus that it infers as having a dropped subject, it posits that subject dropping is possible for all verbs in the language. Similarly it finds least one verb in the corpus has a dropped object and its translation is not in the list of English object-dropping verbs, BASIL posits that object dropping is possible for all verbs in the language. However, if the verbs with object dropping all correspond with English object-dropping verbs, BASIL posits that object dropping is lexically licensed (only possible for some verbs),[33] and finally, if no verbs with dropped objects are found BASIL posits that object dropping is not possible.

The heuristic described above has two potential sources of error. First, because transitivity of verbs does not map directly across languages, a verb that is transitive in English is not necessarily transitive in the source language and vise-versa. Second, the projected dependen-

[33]This heuristic is intended to capture languages with lexically licensed object dropping similar to English. However, the Grammar Matrix requires that those verbs be indicated in the lexicon. I have not integrated this with lexical inference, so it does not have an impact on the resulting grammar. Instead, all verbs in the grammar will be able to drop their objects.

cies provided by INTENT are sometimes missing because an alignment was not found (e.g. because the word in the translation does not match the lemma in the gloss line). However, I find that because argument dropping is so common in the cross-linguistically, examples that are erroneously classified as argument dropping are far outweighed by true examples of argument dropping. I confirmed this by doing intrinsic evaluation on gold grammar specifications, constructed by consulting descriptive resources, and found that this algorithm correctly predicted the subject and object dropping choices for nine out of ten languages: Hausa [hau] (Afro-Asiatic); Indonesian [ind] (Austronesian); Dutch [nld], Greek [ell], Polish [pol], Russian [rus] (Indo-European); Japanese [jpn] (Japonic); Korean [kor] (Korean); Finnish [fin] and Hungarian [hun] (Uralic). The one error comes from Dutch, which has lexically licensed subject and object dropping (Fokkens, 2014), but BASIL inferred subject and object dropping to both be possible for all verbs.

To determine whether a verbal complex has subject and/or object marking, BASIL collects any auxiliaries associated with each verb and collects all agreement markers, using a hand-compiled list of common agreement glosses.[34] Although agreement is not the only way arguments are marked on verbs,[35] it is the most common form and the easiest to identify. In addition to collecting all agreement markers, I use a heuristic to identify whether the agreement markers correspond to more than one argument: if the set of agreement glosses has multiple glosses of a particular category, such as person, number or gender, then BASIL says that the verb is marked for more than one argument. This approach is particularly valuable when a single morpheme is used to mark two arguments or multiple morphemes are used to mark a single argument. For example in (12) from Basque [eus], *dio* is glossed as 3ABS-3DAT.3ERG, containing three third person glosses, so BASIL counts three agreement glosses on that verb.

---

[34]I started this list with the agreement glosses used by MapGloss (Lockwood, 2016) and expanded it based on my own knowledge and observed glosses in the development data.

[35]For example, in Hausa the verb's inflected form depends on whether or not an overt object is present, but this form does not include any PNG information (Newman, 2000).

(12) Eduk neska Toniri aipatu dio
Edu-k neska Toni-ri aipatu d-io
Edu-ERG girl.ABS Toni-DAT mention 3ABS-3DAT.3ERG
'Edu has mentioned the girl to Toni.' [eus] (adapted from Xia et al., 2016)

I use the presence of agreement features on any verb in the set to predict argument marking on the main verb, as follows. Intransitive verbs with any agreement gloss are classified as having subject marking. The orthographies associated with these glosses are collected and stored in a set of known subject markers. After all of the subject markers on intransitive verbs have been collected, BASIL looks at the transitive verbs. Transitive verbs with more than one agreement gloss (like that in (12)) are classified as having subject and object marking. Transitive verbs with only one agreement gloss which corresponds to an orthography in the set of known subject markers are classified as having subject marking and those whose agreement gloss corresponds with an orthography that is not in the set of known subject markers are classified as having object agreement. The set of known subject glosses is included in the input to MOM (Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017). When deciding if a PNG gram should be identified with the subject or object, MOM consults this list and associates it with the subject if the verb is intransitive or the morpheme is in the set of subject morphemes and with the object otherwise.

BASIL's inference for argument optionality has two components: (1) inferring whether subjects and objects can be dropped and if that is possible for any verb or only some verbs, and (2) inferring whether argument marking on the verb is possible or even required when arguments are dropped or overt. The latter involves identifying argument markers in the form of agreement morphemes and discriminating between subject and object agreement markers. The approach described in this section focuses on increasing the coverage of the inferred grammars, while future work to enforce or prohibit argument marking on verbs with overt versus dropped arguments would decrease ambiguity.

4.5.4 Sentential negation

All human languages have a means of expressing sentential negation, but they vary in how many markers are used and whether those markers are independent words, bound morphemes (affixes) (Dahl, 1979; Dryer, 2005; Miestamo, 2008) or a missing morpheme in the paradigm, such as the absence of a tense marker indicating negation in some south Dravidian languages (Master and General, 1946; Crowgey, 2012). Chintang [ctn] exhibits morphological negation, as illustrated in (13), while negation in Nuuchahnulth [nuk] uses an adverb, as in (14).

(13) Aru uŋisukVniŋ.
aru u-ŋis-u-kV-niŋ
another 3nsS/A-know-3P-IND.NPST-NEG
'They did not know another [language].' [ctn] (Bickel et al., 2013a)

(14) wik̇ii· qʷaaʔap.
wik=!i· qʷaaʔap
NEG=CMMD.2 do
'Don't do that.' [nuk] (adapted from Inman, 2019b)

Crowgey (2012) models sentential negation in the Grammar Matrix, allowing it to be marked with 0, 1 or 2 morphemes (calling these strategies zero, simple or bipartite based on the number of markers), which can be bound morphemes, syntactic heads (auxiliaries) or uninflected particles (adverbs). Thus there are three possible values for exponence: 0, 1, 2 and four values for morpheme type: affix, auxiliary, adverb and adverb-like particle.[36] The negation affix and auxiliary are specified in the morphology and lexicon sections, respectively, while the adverbs are defined in the negation section with additional specifications for the type of constituent they attach to, the order in which they attach and their orthography. All

[36]The adverb-like particle is a complement selected by the verb it negates, rather than a traditional modifier which selects for the constituent it modifies.

together this results in a total of 1 possible value for zero negation, 15 for simple and 25 for bipartite.[37] Of these 41 possible values, BASIL is designed to infer 23, as follows.

To infer a language's negation pattern, BASIL first identifies sentences with sentential negation based on the English translation and then targets the gloss line of the IGT to find negation morphemes, based on common glosses, such as 'NEG' and 'not'. In examples (13) and (14), it would identify the morphemes *-niŋ* and *wik* based on the gloss 'NEG'. Negation glosses that are found on affixes, as in (13), are considered to be inflectional negation. I expect that zero-marked negation[38] will be annotated with a negation gloss on the stem or on another morpheme and will therefore be modeled with a non-inflecting lexical rule as described in Section 4.3, so I specify it as morphological simple negation, rather than as zero negation. The distributional properties for negation affixes (including zero-negation) are inferred and specified by the morphological inference system in Section 4.3, which puts a negation feature on the appropriate lexical rule.

The Grammar Matrix can model negation morphemes that are roots as either auxiliaries or adverbs. For this reason, further inference is required for negation morphemes like that in (14). The English dependency parse does not help in this decision, as it simply encodes facts about negation in English. Instead, BASIL compares negation roots with the auxiliaries collected in Section 4.5.1. If they were inferred as auxiliaries, BASIL treats them as such and otherwise it defines them as adverbs. The required specifications for the negation auxiliaries' lexical entries were already inferred in the process described in Section 4.5.1, so there is no additional work to be done. In the case of negation adverbs, the Grammar Matrix models two lexical types: an adverb and what Crowgey (2012) refers to as an adverb-like particle. While this distinction is useful from a syntactic modeling perspective (for example, it allows the negation particle to only be selected by specific verbs), it is difficult to infer reliably from the corpus and I do not expect it have an impact on the coverage of the inferred grammar

[37]For the bipartite negation strategy Crowgey (2012) models the possible combinations of the of negation morpheme types based on combinatorics. These possible combinations are not necessarily attested in the world's languages.

[38]This is not attested in any of my development languages.

(unless BASIL were to also infer which verbs co-occur with the negation marker). Therefore, BASIL specifies all negation morphemes that are independent words and are not auxiliaries as regular adverbs. To define these adverbs, BASIL uses the same process as it did for auxiliaries to decide whether the adverb attaches to a VP or S (e.g. checking whether there is a subject between the auxiliary and the verb) and whether they occur before or after that constituent.

After collecting the negation markers for each IGT in the corpus that contains sentential negation, BASIL compares the number of sentences that include one negation marker with those that include more than one negation marker. While it only looks at sentences with sentential negation, it does not distinguish between sentential and constituent negation markers in these sentences, and can mistake a negated sentence with additional constituent negation for bipartite negation. However, I expect sentences that contain both sentential and constituent negation to be less common that those that only contain constituent negation. For this reason, BASIL posits the most common strategy it observes in the corpus (simple or bipartite), which I expect to only be bipartite if in fact there were true examples of bipartite negation.

If simple negation is the most common, the Grammar Matrix allows the definition of multiple strategies, so BASIL adds each strategy it found (affix, auxiliary, and adverb) to the grammar specification. For bipartite negation, it is only possible to specify one combination of markers, so if bipartite negation was the most common strategy found in the corpus, BASIL adds the two most common co-occurring types of negation markers (e.g. adverb and affix) to the grammar specification. While the Matrix only allows the addition of one orthography for a negation adverb requiring BASIL to use the most common, it can specify as many negation affixes and auxiliaries as were found in the corpus.

4.5.5 Coordination

Coordination is possible for a wide range of constituents, called coordinands, and can be marked with either free or bound morphemes, called coordinators. Coordinators attach to all (omnisyndedic), all but one (polysyndedic), one (monosyndedic) or none (asyndedic) of

the coordinands (Drellishak, 2004; Haspelmath, 2007). The following example from Abui [abz] shows two noun phrases which are coordinated with a free morpheme.

(15) Luka-luka nuku ya yoikoi nuku ba dining ayoku tefeela.
luka-luka nuku ya yoikoi nuku ba dining ayoku tefeela
monkey one and turtle one Rel 3.AGT-in.number two DISTR.AL-friend
'A monkey and a turtle, the two of them, they were friends.' [abz] (adapted from Kratochvíl, 2019)

The coordination section of the Grammar Matrix's specification file (Drellishak and Bender, 2005) allows the definition of any number of coordination strategies. Each strategy must include a value for the pattern: omnisyndedic, polysyndedic, monosyndedic or asyndedic and one or more constituent types: N, NP, V/VP[39] or S. The strategy must also indicate whether the coordinator is a word or an affix and if that morpheme attaches before or after the coordinand. BASIL infers 11 of these 12 values, excluding polysyndedic coordination.

As with sentential negation, BASIL identifies IGT that exhibit coordination based on the English translation and then finds the coordinators first by looking for the word aligned by INTENT (Georgi, 2016) with the English coorindator and then, because alignment isn't always successful, by looking for the glosses 'COORD', 'CONJ', 'CCONJ' and 'and'. In (15), it would find the coordinator *ya* via the gloss 'and'. BASIL determines whether the coordinator is an affix or independent word by checking to see if it is the root or if it is attached to some other word (in (15), *ya* is the only morpheme in the word, so it is considered an independent word). Then BASIL uses the projected dependencies to collect the dependent of each coordinator (in this case *luka-luka* and *yoikoi*) and these dependents are assumed to be the coordinands.[40] BASIL then compares the number of coordinators and coordinands to

[39]The coordination rules added by the Grammar Matrix do not distinguish between verbs and verb phrases.

[40]As a fallback, if BASIL cannot find coordinands via projected dependencies, it looks for them by collecting the words that occur in between coordinators. This approach is less successful for monosyndedic coordination.

decide if the sentence exemplifies asyndedic, monosyndedic or omnisyndedic coordination. Differentiating between mono- and polysyndedic coordination is rather difficult as most examples in the corpora only have two coordinands, and the construction 'A and B' could be either mono- or polysyndedic. However, monosyndedic coordination can be used to model polysyndedic (e.g. [[A and B] and C]), so BASIL can default to predicting monosyndedic in all cases that might be mono- or polysyndedic and the resulting grammar will parse the same sentences as if we were able to distinguish between the two.

For each coordination strategy, BASIL also identifies the lexical category of the coordinand (noun or verb) and uses heuristics to decide at what level the coordination takes place (word or phrase in the case of nouns and word/phrase or sentence for verbs). The category is determined based on the POS tags, while the level requires a little more work. The Grammar Matrix only allows coordination marked by an affix at the word-level, so if the coordinator is an affix, we specify the coordination strategy for nouns, and otherwise BASIL defaults to noun phrase. Thus as the coordinator in (15) is an independent word, BASIL would infer the two noun phrases are being coordinated. For coordination of verbs, BASIL defaults to verb phrases unless it sees explicit evidence for sentence level coordination. It checks for a projected subject dependency on the coordinand and if it finds a subject between the coordinator and the verb, it posits sentence coordination, and otherwise defaults to verb phrase coordination. Because the Grammar Matrix allows the definition of any number of coordination strategies, BASIL adds each distinct coordination strategy that is attested in the corpus to the grammar specification.

4.6 Summary

In this section I described four types of inference that produce necessary components of my inferred grammar specifications: lexical, morphological, syntactico-semantic and syntactic. For inference of noun and verb lexical classes and lexical entries, I rely primarily on the MOM morphotactic inference system (Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017), but make new contributions to lexical inference in the form of auxiliary, adposition

and determiner inference as well lexical types defined as part of syntactic inference such as negation adverbs or coordinators. I also use MOM to infer morphological rules for nouns and verbs, and build on the system by improving the detection of subject and object agreement, as described in Section 4.5.3 and adding the definitions of PNG and TAM features to the grammar specification, so that these syntactico-semantic features can be leveraged by the grammar and included in the semantic representations that I describe in the following chapter. I built on previous algorithms for inferring syntactic properties such as word order and case and added to this to account for argument optionality, negation and coordination.

The scope of this inference spans a large number of feature-value pairs in the grammar specification, as I illustrated in Table 4.1, and testing the inference for all of these on real data would require a vast set of datasets from typologically diverse languages. At the same time, its possible that specifications allowed by the Grammar Matrix or targeted by BASIL are not sufficient to correctly model some languages. In the following chapter, I describe my data-driven approach to development in which I considered corpora from a wide range of diverse languages from a variety of data formats to develop and test the algorithms detailed in this chapter.

Chapter 5

DATA-DRIVEN DEVELOPMENT

I developed the inference algorithms described in Chapter 4 using a data-driven approach in which I consulted the typological literature for each phenomenon and actively tested each algorithm on a diverse set of languages throughout implementation. In this chapter, I describe the languages and datasets I used during development (§5.1) and the cyclical process of development and testing that I used to refine the system (§5.2). Next, I describe BASIL's performance by first quantifying the set of feature/value pairs for the grammar specification (as described in Chapter 4) that the system inferred for the development languages (§5.3) and present the performance of the inferred grammars over held-out portions of data for these languages (§5.4).

5.1 Development Languages and Datasets

In order to thoroughly test BASIL on the diverse phenomena described in Chapter 4, it is necessary to use languages that are typologically varied, representing as many language families and geographic areas as possible. At the same time, acquiring datasets that are suitable for grammar inference requires either finding those datasets either in archives or getting them directly from documentary linguists. For development, I collected nine datasets for languages from seven language families and four continents, which constitute my core set of development languages and are shown in purple on the map in Figure 5.1. Large, thoroughly annotated datasets such as these are best for testing the performance of inferred grammars; however, inference for individual phenomena can be checked against smaller and less consistent datasets. Therefore, I supplemented my development languages with datasets for an additional 18 languages to span a total of 19 language families and six continents.

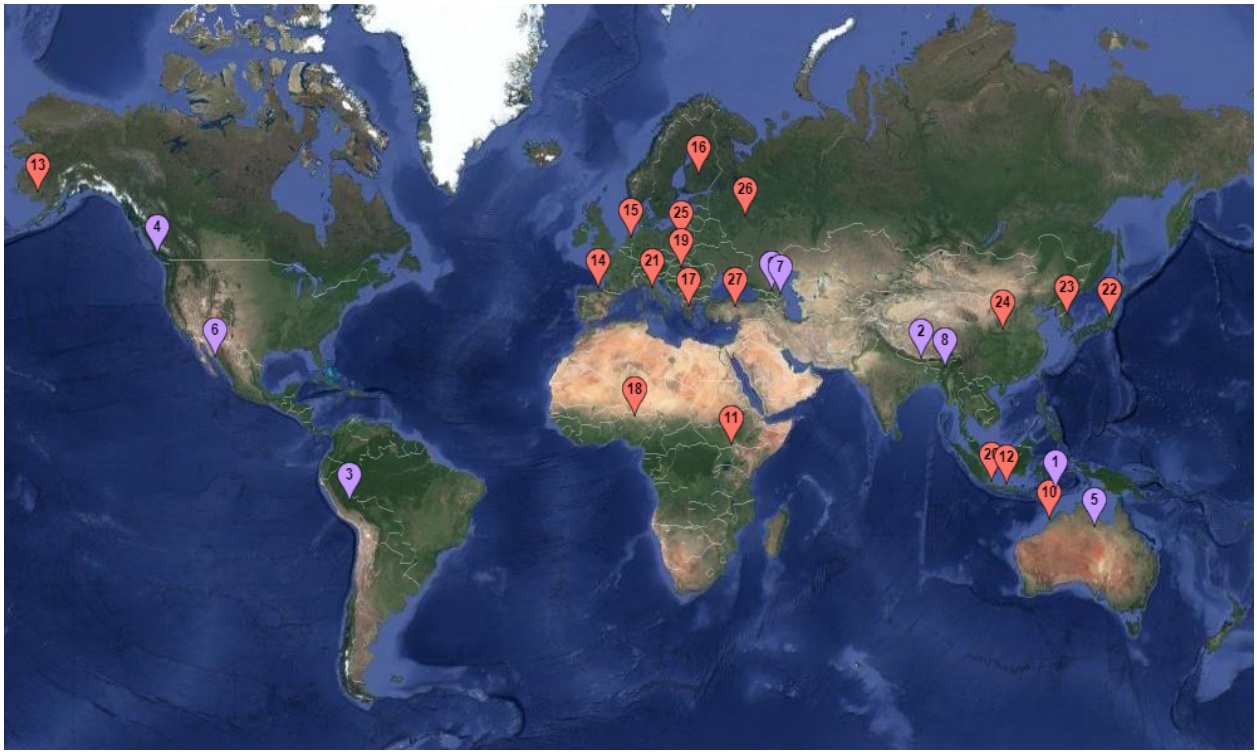


Figure 5.1: Map of the coordinates where languages used in the development are spoken

These languages, their language families and the corpora and descriptive resources I used are listed in Table 5.1. Their geographic distribution is shown on the map in Figure 5.1, with development languages in purple and additional consulted languages in red.[1]

I selected the development languages based on the size and quality of the dataset as well as for some of the syntactic phenomena exhibited by those languages. The majority of the corpora (Abui, Chintang, Haiki, Lezgi, Matsigenka, Meithei, Nuuchahnulth and Tsova-Tush) come from a FLeX or Toolbox corpus that was curated by a documentary linguist (or a group of linguists). However, during the development and implementation of inference for specific syntactic and morpho-syntactic phenomena, I consulted additional datasets for languages which represent those phenomena. These datasets not only contribute to the diversity of

[1]These coordinates come from WALS (Dryer and Haspelmath, 2013) in most cases and were estimated from other sources if information from WALS was not available.

	Language	iso	Family	Corpus	Descriptive Resource
	Development				
1	Abui	abz	Trans-New Guinea	Kratochvíl 2019	Kratochvíl 2007
2	Chintang	ctn	Sino-Tibetan	Bickel et al. 2013d	Schikowski 2013
3	Matsigenka	mcb	Arawakan	Michael et al. 2013	Michael 2008
4	Nuuchahnulth	nuk	Wakashan	Inman 2019b	Inman 2019a
5	Wambaya	wmb	Mirndi	Nordlinger 1998	Nordlinger 1998
6	Haiki	yaq	Uto-Aztecan	Harley 2019	Sanchez et al. 2015 Dedrick and Casad 1999
7	Lezgi	lez	Nakh-Daghestanian	Donet 2014b	Donet 2014a
8	Meithei	mni	Sino-Tibetan	Chelliah 2019	Chelliah 2011
9	Tsova-Tush	bbl	Nakh-Daghestanian	Hauk 2016–2019	Hauk and Harris forthcoming Hauk 2020
	Consulted				
10	Bardi	bcj	Nyulnyulan	Bowern 2012	Bowern 2012
11	Ik	ikx	Eastern Sudanic	Schrock et al. 2014	Schrock et al. 2014
12	Old Javanese	jav	Austronesian	Acri 2018	
13	Yup'ik	esu	Eskimo-Aleut	Miyaoka 2012	Miyaoka 2012
14	Basque	eus	Basque	Xia et al. 2016	de Urbina 1989
15	Dutch	nld	Indo-European	Xia et al. 2016	Booij 2002
16	Finnish	fin	Uralic	Xia et al. 2016	Sulkala and Karjalainen 1992
17	Greek	ell	Indo-European	Xia et al. 2016	Holton et al. 2012
18	Hausa	hau	Afro-Asiatic	Xia et al. 2016	Newman 2000
19	Hungarian	hun	Uralic	Xia et al. 2016	Kenesei et al. 2002
20	Indonesian	ind	Austronesian	Xia et al. 2016	Sneddon et al. 2012
21	Italian	ita	Indo-European	Xia et al. 2016	Monachesi 1996
22	Japanese	jpn	Japonic	Siegel et al. 2016 Xia et al. 2016	Siegel et al. 2016 Hinds 1986
23	Korean	kor	Korean	Xia et al. 2016	Sohn 1994
24	Mandarin	cmn	Sino-Tibetan	Xia et al. 2016	Li and Thompson 1989
25	Polish	pol	Indo-European	Xia et al. 2016	
26	Russian	rus	Indo-European	Xia et al. 2016	
27	Turkish	tur	Altaic	Xia et al. 2016	Kornfilt 1997

Table 5.1: Languages used in development

Language	iso	Source Type	Size	POS tags in source
Abui	abz	Toolbox	1568	yes
Chintang	ctn	Toolbox	9785	yes
Haiki	yaq	FLeX	2235	yes
Lezgi	lez	FLeX	1168	yes
Matsigenka	mcb	FLeX	349	yes
Meithei	mni	FLeX	955	yes
Nuuchahnulth	nuk	FLeX	641	no
Wambaya	wmb	Book	818	no
Tsova-Tush	bbl	FLeX	1601	yes

Table 5.2: Source, size and presence of POS tags for the development datasets

the languages I worked with, but also to the variety of source formats and dataset styles. A number of the datsets I consulted for individual phenomena (languages 14-27) come from the ODIN corpus (Xia et al., 2016), which is a collection of IGT scraped from academic papers. I also extracted corpora from descriptive grammars for Wambaya, Bardi, Japanese,[2] Ik and Yup'ik, using the pipeline for extracting IGT from text and converting it to the Xigt data model developed by Xia et al. (2016).

In addition to listing the corpus itself, Table 5.1 also lists the descriptive resources that I consulted during development. In some cases these are descriptive grammars developed by the same linguist who collected the data for the corpus (e.g. for Abui, Matsigenka, Meithei), and in other cases they are the descriptive grammars from which I extracted the data (e.g. Wambya, Bardi, Ik). At other times, as is the case for the ODIN datasets, I consulted grammars or papers for the description of a specific phenomenon by authors who were not involved in the data collection or curation.

In this chapter, I describe the process of refining the system using the nine development datasets and the coverage of the inferred grammars over those datasets. In order to contextualize that discussion, I will first give a brief overview of each of these languages and

[2]At various stages, I consulted two corpora of Japanese, one from ODIN (Xia et al., 2016) and one that I extracted from the IGT used in Siegel et al. 2016.

their respective datasets. Some relevant characteristics of those datasets (or the subset of the dataset that I used) are summarized in Table 5.2.

Abui

Abui [abz] is an Alor-Pantar language in the Trans-New Guinea language family. It has about 16,000 speakers and is primarily spoken on the Alor island of Indonesia (Kratochvíl, 2007). This dataset (Kratochvíl, 2019) comes from a Toolbox corpus which contains about 18,000 sentences from both elicitation and transcribed speech. This is part of an ongoing documentation effort, so the dataset is only partially glossed. For this reason, I filtered the data based on the presence of full segmentation and glossing to create a dataset of 1,500 sentences after removing ungrammatical examples and duplicates.

Chintang

A Kiranti language of the Sino-Tibetan family spoken in Nepal, Chintang [ctn] has 4,000-5,000 speakers (Schikowski, 2013). The Toolbox dataset is quite large, coming from a long-term documentation effort (Bickel et al., 2013d). I use a fully segmented and glossed subset of the data containing almost 10,000 sentences. The type of language represented in the corpus is extremely diverse, containing transcribed conversations, ritual language, narratives and a few other genres.

Haiki

Haiki [yaq] is a Taracahitic language of the Uto-Aztecan family. There are multiple spellings of this language name, including Yaqui, which is the official name of the tribe in the United States and Mexico; however Haiki is the correct spelling in the Pascua Yaqui orthography (Sanchez et al., 2015). It is spoken by about 21,000 people in Mexico and the United States (Eberhard et al., 2019). The corpus (Harley, 2019) is quite large with almost 11,000 IGT, but as with most ongoing projects, is only partially annotated with interlinear glosses and part-

of-speech tags. After filtering IGT with no glosses and removing ungrammatical examples and duplicates, I worked with a set of just over 2,000 IGT.

Lezgi

Lezgi is a member of the Lezgian subgroup of the Nakh-Daghestanian language family (Donet, 2014a). It is spoken by about 400,000 people (Eberhard et al., 2019), primarily in Daghestan and Azerbaijan (Donet, 2014a). The glossing and POS tagging in this corpus (Donet, 2014b) are also fairly complete, resulting in a set of over 1,100 IGT after minor filtering and removing ungrammatical examples and duplicates.

Matsigenka

Matsigenka [mcb] is a Maipurean language of the Arawakan family spoken in Peru by about 10,000 people (O'Hagan, 2018). The FLeX corpus (Michael et al., 2013) is made up of narratives that are fully segmented and glossed. Of the approximately 5,000 IGT in the corpus, some have English translations, while the vast majority include Spanish translations. Because the AGGREGRATION pipeline described in Chapter 3 relies on a system called INTENT (Georgi, 2016) which parses the English translation of an IGT and projects the dependency parses onto the language, BASIL indirectly requires IGT with English translations. From the full Matsigenka corpus, we[3] identified about 350 IGT with English translations, which I use as a development dataset.

Meithei

Meithei [mni] is a Kuki-Chin-Naga language of the Sino-Tibetan language family. It is spoken predominately in Manipur State, but has about 56 million speakers living across a wide region, including in China, India, Nepal and Myanmar (Chelliah, 2011). The FLeX corpus (Chelliah, 2019) contains about 1,800 IGT, but as part of an ongoing documentation effort,

[3]Most of these were identified by previous research assistants on the AGGREGATION project and more were extracted by Angelina McMillan-Major.

is only partially annotated. After filtering for fully-glossed IGT and removing duplicates and ungrammatical examples, the corpus has about 1,000 items. Compared to other corpora in my development set, this corpus contains a high proportion of complex sentences, which include subordinate clauses that are not covered by inference. Nevertheless, it is a strong example for how much typological information can be learned from a corpus, even when many of the sentences contain phenomena that are out of scope for the inference system.

Nuuchahnulth

Nuuchahnulth [nuk] is Southern Wakashan language of Vancouver Island in Canada and has only about 130 fluent speakers (Eberhard et al., 2019). The FLeX dataset (Inman, 2019b) was curated in connection with a dissertation on multi-predicate constructions and contains both transcribed narratives and elicitations, many of which target this construction. The dataset includes about 650 examples which are fully glossed and segmented. However, Inman's corpus did not include POS tags, which are required by MOM (Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017) to build the lexicon of nouns and verbs. For many IGT, these are available from the projected part of speech tags from INTENT. However, because INTENT does not always successfully find an alignment (this can be particularly challenging for polysynthetic languages), I use an additional heuristic to identify verbs. Because single-word sentences are very common in this poly-synthetic language, I supplemented the projected POS tags by pre-processing the corpus to assign a verbal POS tag to the only word in any one-word IGT, if the dependency parse for the translation was headed by a verb. In other words, if the parse of the translation indicated that the IGT is a one-word sentence, or one-word verb phrase, I assigned that instance of the word a verbal POS tag.

Wambaya

Wambaya [wmb] is a West Barkly language in the Mirndi family, which has about 60 speakers (Eberhard et al., 2019). The Wambaya dataset is distinct from my other development datasets because it was extracted from the examples in a descriptive grammar (Nordlinger,

1998). As such, it does not contain linguist-provided POS tags and the possibility of alignment errors in the interlinearization is higher, due to the process of extracting IGT from text rather than an aligned corpus. Nevertheless, this language illustrates a number of phenomena that guided my development, such as the second position clitic cluster, which informed my methodology for extracting auxiliaries that are not aligned with an English auxiliary in the translation, as well as inferring verb-second word order based on the frequency of a second position auxiliary (as described in Section 4.5.1). Furthermore, the use of a descriptive grammar allowed me to explore the possibility of inferring grammars to accompany descriptive resources along the lines of Bouma et al. 2015.

Tsova-Tush

Tsova-Tush [bbl], also referred to by the endonym Bats or Batsbi, is a Northeast Caucasian language of the Nakh subgroup of the Nakh-Daghestanian language family (Hauk and Harris, forthcoming). It is spoken in Georgia by about 2,500-3,200 people (Hauk and Harris, forthcoming). The corpus (Hauk, 2016–2019) contains elicitation and transcribed text and the glossing and part of speech tags are almost complete, including over 1,600 IGT after removing ungrammatical examples and duplicates.

5.2 Development-testing Cycle

Developing BASIL involved a cyclical process of developing and implementing an algorithm for a specific phenomenon, testing it on data from multiple languages and refining the algorithm or its implementation as needed. As a first pass, the most straightforward way to do this is to construct gold-standard versions of the grammar specifications based on descriptive resources for each language and compare the inferred specification to the gold.[4] However, it is difficult to know what the best specification for a language is by merely looking at the features themselves, and in some cases, which value is "correct" for a feature is debat-

[4]This approach to evaluation has been systematically applied by Bender et al. (2013) and Howell et al. (2017), inter alios.

able. This dissertation focuses on maximizing the coverage of the inferred grammars, while minimizing spurious and incorrect parses. Therefore, after the initial implementation of the inference modules, I tested them in combination with each other using *coverage*, the number of sentences that parse, and *ambiguity*, the number of parses per sentences, to identify areas for improvement in the inference system.

While visually inspecting the grammar specifications for correctness is valuable, using the grammar generated from these specifications to parse sentences in the language can reveal ways in which a combination of specifications that are not necessarily wrong on an individual level combine in an undesirable way. I found one example of this in the interaction of verb transitivity and argument optionality inference. In Section 4.5.2, I described my approach to classifying verbs as transitive or intransitive based on the presence or absence of an object in both the language line and the English translation of the sentence. The methodology for inferring object dropping (§4.5.3) also compares the English verb's valence with the presence or absence of an object in the language line to infer whether object dropping is possible. Consider the following examples from Chintang.

(16) thi thuŋma kondno.
thi thuŋ-ma kond-no
beer drink-INF should-IND.NPST
'...we have to drink beer.' [ctn] (adapted from Bickel et al., 2013c)

(17) Thuŋno.
thuŋ-no
drink-IND.NPST
'(He) drinks' [ctn] (adapted from Bickel et al., 2013b)

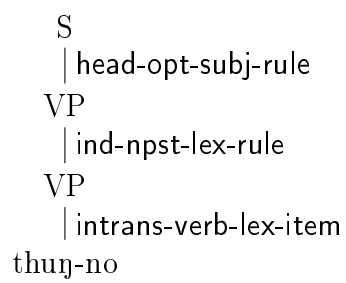
The module for transitivity and case frame infers that the verb *thuŋ* 'drink' in (16) is transitive because of the overt direct object *thi* 'beer'. However, it would also infer that the verb

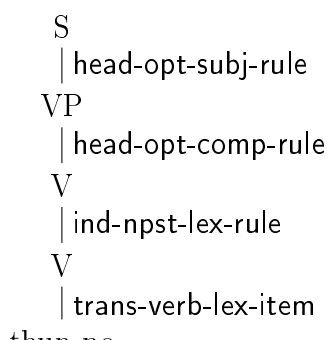
thuŋ 'drink' in (17) is intransitive because there is no direct object in the language line or the translation.

My initial implementation of transitivity inference would create two separate lexical entries for this verb. Modeling this orthography with two lexical entries (one transitive from (16) and one intransitive from (17)) is appropriate from the perspective that the verb in (17) could be analyzed as an unergative verb meaning 'do drinking', where the verb is its own syntactic argument (Harley (2011) summarizing Hale and Keyser 1993, 2002). However, from example (16), BASIL would infer that object dropping is possible in the language because there is no direct object in the language line and because BASIL consults a list of verbs that are modeled as object dropping in the English Resource Grammar (Flickinger, 2000, 2011) and *drink* is on that list. BASIL would also likely infer that object dropping is possible from other verbs in the corpus with dropped objects.

The grammar produced by this combination of specifications (two lexical entries for *thuŋ* and object dropping allowed) will have ambiguity in that a sentence like (17) would have two parses.[5] The first would be licensed by intransitive lexical entry, as in (18), and the second by the transitive lexical entry and the object dropping phrase structure rule, as in (19). To eliminate this ambiguity, I modified BASIL so that argument optionality inference takes place before the lexicon is constructed. Now BASIL creates both lexical entries if different transitivities are inferred from different IGT only if object dropping was not inferred. If object dropping is possible, it creates only the transitive lexical entry so that the resulting grammar will only produce the parse in (19).

[5]Here I say "two parses" for simplicity in the explanation. In reality, it would have double the parses, as additional ambiguity might be introduced e.g. by multiple analyses for the *-no* suffix.

(18)


(19)


This approach effectively reduces ambiguity in the inferred grammars, but whether the semantic representation produced by these parses correctly represents the semantic arguments of the verbs is debatable. In grammars produced by the Grammar Matrix, a verb will have as many semantic arguments as syntactic arguments, so the semantic representation for (19) has a null object argument (ARG2). In future work, the inference algorithm could be further refined to identify sub-categories of verbs that indeed drop their arguments in contrast with verbs that are better modeled with separate transitive and intransitive lexical entries. This would allow the grammar to produce different semantic representations for verbs for which a dropped object is recoverable from context and those for which there is no object.

Another problem that was revealed from parsing sentences with inferred grammars was ambiguity that resulted from non-inflecting lexical rules in the inferred Haiki grammars. As illustrated by the 3.PL.NOM gloss on *vempo* 'they' in (20), pronouns in this corpus are glossed with PNG and case grams on the root.

(20) Vempo bwiikataite.
vempo bwiika-taite
3.PL.NOM sing-INCEP
'They are starting to sing.' [yaq] (adapted from Harley, 2019)

This is a very common glossing practice and is reflected in many of the development datasets. The result of MOM's original algorithm for non-inflecting lexical rules was to create a zero

morpheme lexical rule for each set of features glossed on the root. From (20), MOM would create a non-inflecting lexical rule with third person, plural number and nominative case features and it would do the same for the glosses of every pronoun in the corpus. Because the Haiki corpus has so many pronouns glossed with this convention, the ambiguity which resulted from these rules was abundantly evident when we[6] parsed sentences with the inferred grammar.

To mitigate this problem, I modified MOM so that non-inflecting rules are only created if they fill a gap in the inflectional paradigm. In other words, a non-inflecting rule for nominative case should only be created if there is an inflecting rule for some other case (e.g. accusative) and the non-inflecting rule should belong to the same position class as the rule that adds a feature from the same category. To accomplish this, I modified MOM so that it collects (but does not yet define) potential candidates for non-inflecting lexical rules before it infers inflecting lexical rules. Then each time it creates a position class for an inflecting lexical rule that contributes features, MOM checks its list of non-inflecting lexical rules for one that contributes the same types of features. If it finds one, it adds that non-inflecting rule to the position class. For example, MOM will identify a *potential* non-inflecting rule that adds third person, plural number, nominative case to nouns from the IGT in (20). Then, if MOM finds an affix that is marked with person, number and case features, it will add a non-inflecting lexical rule for third person, plural number and nominative case to the position class it creates for that inflecting lexical rule. If no such affix is found, however, a non-inflecting rule for these features will not be added to the grammar. Because the pronoun definitions already contain the appropriate features in their lexical entries due to pronoun inference (§4.2.2), this reduces ambiguity caused by non-inflecting rules without losing feature information on pronouns.

The process of evaluating the inference algorithms by parsing sentences with the inferred grammars and comparing their coverage and ambiguity was a key part of fine-tuning the

[6]This was discovered by Heidi Harley and Emily M. Bender when interacting with the inferred grammar at a workshop at UW in September 2019.

system. Using this approach, I assessed the inferred specifications in interaction with each other rather than in isolation. In doing this, I went beyond answering the question of what a correct grammar specification looks like for a language to answering the question of how can I infer the most robust and efficient grammar for a language.

5.3 Phenomena Tested by Development Languages

Having described my development languages and how I used data-driven development to fine-tune my inference algorithms, in this section I quantify the degree to which the inference system was tested on these phenomena during this process. In Section 4.1, I described the space of the inference task in terms of the number of features and values that BASIL is designed to add to the grammar specification to account for the phenomena it handles. I identified 50 features with a fixed set of values (listed in Table 4.1) totaling 136 possible values in the Grammar Matrix grammar specifications that are relevant to the phenomena targeted by BASIL. My system is designed to infer 99 of those 136 values. When inferring grammar specifications for the 9 development languages, 37 of the 50 features and 71 of the 99 values were tested. These values are broken down by phenomenon in Table 5.3. I also reported in Section 4.1 that my inference system can identify 99 morpho-syntactic and morph-semantic features from their glosses in the IGT. BASIL finds 66 of those 99 features in the development datasets, which are broken down by category in Table 5.4.

While the development languages allowed me to test most of the values BASIL targets, they did not represent all of the possible values for person, word order, auxiliary order, case-system, argument optionality, negation and coordination. In particular, none of these languages exhibited bipartite negation, which is relatively rare in the world's languages, accounting for only about 10% of the languages surveyed by Dryer (2013a). The other category in which the development languages did not span the majority of the targeted values was person. BASIL found examples of first, second and third as well as third and

[8]Person features are not included in this count because the Grammar Matrix defines these automatically based on the over-arching person system.

Phenomenon	**# possible**	**# targeted by inference**	**# inferred from dev languages**
noun lexical entry	4	2	2
verb lexical entry	4	2	2
auxiliary lexical entry	6	4	4
adposition lexical entry	3	3	3
morphological rule	5	5	5
person	9	8	4
tense	2	1	1
word order	10	9	6
determiner order	4	4	4
auxiliary order	9	9	7
case system	9	3	2
argument optionality	18	15	12
sentential negation	41	23	9
coordination	12	11	10
total	136	99	71

Table 5.3: The number of possible values for the closed set features to define phenomena in the grammar specification and, those targeted by the inference system and those attested in the development languages

Feature Category	# Found
Number	4
Gender	5
Case	21
Tense	6
Aspect	16
Mood	14
Total	66

Table 5.4: The number of morpho-syntactic features corresponding with PNG,[8] TAM and case found in the development languages

non-third person systems, but it did not find examples of the other systems (first, second, third and fourth; first and non-first; second and non-second; and none; Drellishak, 2009).

While the development languages do not exhaustively test every facet of the inference system, they do test a significant portion. I also consulted an additional 18 languages (represented in red in Figure 5.1) to test as many of the feature-value pairs as possible, in order to create a system that would generalize beyond the development languages.

5.4 Final State of Development Grammars

As described in Section 5.2, I used an iterative process that begins with inspecting the specifications produced by each inference module for correctness followed by evaluating the grammars inferred by the entire inference system on held-out sentences in the language. This section describes the performance of the inferred grammars at the end of the development process in order to contextualize the results of evaluation on held-out languages in Chapter 7.

To evaluate system performance on the development languages, I used 10-fold cross validation. That is, for each language/dataset, I divided the dataset into ten even test groups. For each fold, I held-out one of the test groups for evaluation and used the remaining 9 folds to infer a grammar. I assessed the inferred grammars by parsing sentences in their respective test folds, using five metrics:

- **lexical coverage** the proportion of sentences for which the grammar has an analysis for each word in the sentence, meaning that the lemma is in the lexicon and the morphology is accounted for by the grammar's inflectional rules

- **parse coverage** the proportion of sentences the grammar parses, meaning that words unify with the necessary syntactic rules to form a sentence

- **correct predicate-argument structure** the proportion of sentences the grammar parses, producing a semantic representation that includes appropriate predications and arguments for each semantic entity

- **correct predicate-argument structure and semantic features** the number of sentences for which the grammar produces the correct predicate-argument structure as well as the appropriate PNG and TAM features on those arguments

- **ambiguity** the average number of results per sentence that parses

5.4.1 Scope of development grammars

The phenomena targeted by BASIL, which I described in Chapter 4, are only a subset of the phenomena necessary to fully model a language or to parse all of the sentences in the corpora I work with. For this reason, understanding the types of sentences I do not expect to parse lays the groundwork for understanding what the inferred grammars should parse, but don't. First, a number of lexical types that BASIL does not infer will prevent the grammar from having lexical coverage over sentences that contain those types of words. These include but are not limited to adjectives and adverbs, as well as particles marking complementation, subordination, information structure, questions and possession. Because these words may be homophonous with words that BASIL does handle, sentences with these lexical types may have lexical coverage and the grammar might even produce one or more parses for them, but those parses will not be correct.

At the same time there are a number of syntactic phenomena that do not require additional lexical items, but without inferring the appropriate syntactic or lexical specifications, the inferred grammars will not be able to parse them or parse them with the correct predicate-argument structure. For example, morphologically-marked subordinate clauses (e.g. adverbial or relative clauses that do not contain a subordinator as a separate word) and imperatives do not require that any special lexical items be defined, but do require syntactic specifications (Howell and Zamaraeva, 2018). In other cases, the lexicon contains entries for the orthographies (inferred by MOM based on their POS tags), but the lexical classes themselves are not correct. These include wh-words (e.g. *who* defined as a regular noun or pronoun, but not as a question word), reciprocal or reflexive pronouns, possessives, quanti-

fiers and ditransitive or clausal complement verbs, which might be defined as regular nouns, verbs or determiners. Without defining the appropriate lexical subtypes, these lexical items will not be modeled correctly by the grammar. These phenomena are the biggest contributors to gaps between parse coverage and correct coverage in the development languages.

Some parses have the correct predicate-argument structure but lack some semantic features as a result of out-of-scope syntactic phenomena that contribute information to the semantic structure. As an example, yes/no questions and imperatives are traditionally modeled in the DELPH-IN formalism with a the SF (sentential force) feature, which can have the values prop (proposition), ques (question) or comm (command) (Flickinger et al., 2014). BASIL does not infer specifications for questions or imperatives, but the inferred grammars for some languages parse these sentences with the correct predicate-argument structure. The grammars do not, however, use the appropriate prop or comm value and therefore, I consider them to have incorrect features. In the Wamabaya corpus (Nordlinger, 1998), imperatives are annotated with imperative mood, which is represented in the inferred grammars as mood feature in the semantics. However, this does not equate to an SF comm feature, so I consider these parses to be missing a semantic feature. With this context established, the next subsection presents the performance of the development grammars.

5.4.2 Coverage for development languages

Table 5.5 presents the results using all five metrics for each of the development languages. Whereas calculating the lexical coverage, parse coverage and ambiguity are automated processes, calculating the correct predicate-argument structure and features require manual inspection of the semantic representations for each sentence.[9] For this reason, I provide results for correct predicate-argument structure and correct predicate-argument structure and features across all folds for the languages with less than 1,000 IGT, but for those with more than 1000 IGT, I provide those metrics only for the first of the ten folds.

[9]For a detailed description of these processes, see Section 6.1.

Language [iso]	Lexical Coverage (%)	Parse Coverage (%)	Correct Pred-Arg Structure (%)	Correct Pred-Arg Strugure and Features (%)	Ambiguity
Abui [abz]	53.19	41.96	10.19*	5.73*	2195
Chintang [ctn]	22.29	12.24	3.58*	1.94*	5562
Haiki [yaq]	17.49	10.29	1.79*	0.89*	161
Lezgi [lez]	7.88	6.08	0.00*	0.00*	10419
Matsigenka [mcb]	12.61	8.02	1.15	1.15	2333
Meithei [mni]	5.86	5.24	1.05	0.42	3722
Nuuchahnulth [nuk]	23.09	10.14	1.87	1.09	265
Wambaya [wmb]	9.41	2.08	0.98	0.12	4
Tsova-Tush [bbl]	28.79	24.05	4.35*	0.00*	3418

Table 5.5: Coverage and Ambiguity for Development Languages. Results are averages across 10 folds. * indicates results for only a single fold

The first four metrics shown in Table 5.2 subsume each other in that only sentences that have lexical coverage can be parsed and only sentences that are parsed can be parsed correctly. Thus, the sentences for which the grammar produces a semantic representation with the correct predicate-argument structure and features are a subset of those for which the grammar produces a semantic representation with the correct predicate-argument structure. In turn, those are a subset of the sentences with parse coverage, which are a subset of those with lexical coverage. This is illustrated by the bar graph in Figure 5.2.

To contextualize this performance, remember that the datasets come from a wide range of sources including transcribed speech, elicitation, and narratives. Transcribed speech and elicitations often include sentence fragments, which the grammar will not accept as sentences. For this reason, and because of the many out-of-scope pheonomena described above, I do not expect the inferred grammars to parse a very large portion of the held-out sentences they are tested on. Instead, the most useful comparison to consider is the number of sentences that parsed with the correct predicate-argument structure or correct predicate-argument structure and features versus the number of sentences that parsed, but did not have the correct semantic representation.

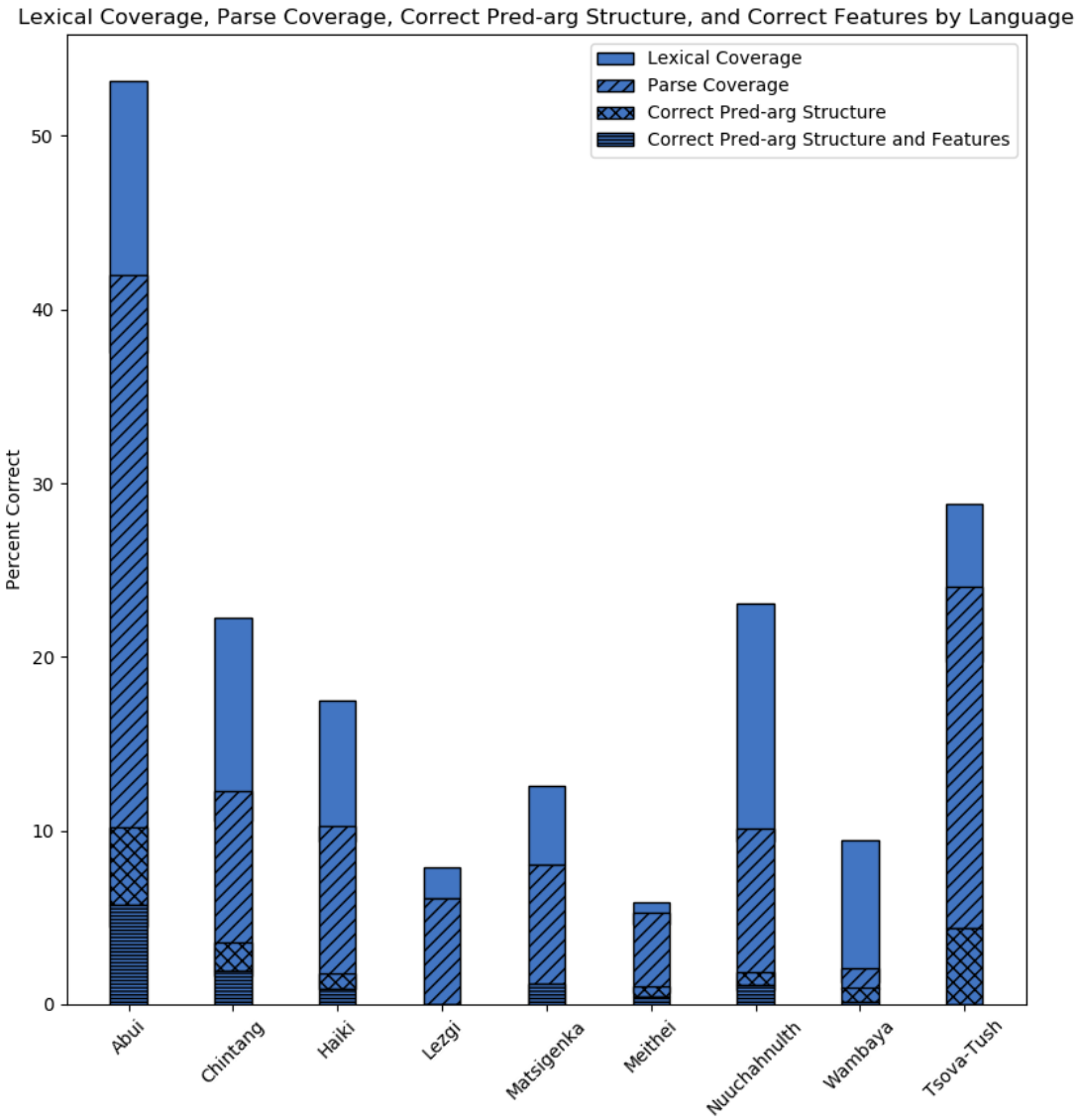


Figure 5.2: Results for Development Languages

Previously, little work has been done that evaluates inferred grammars on held-out test items. Hellan (2010) and Hellan and Beermann (2011) do not present any evaluation for their inference system and Indurkhya (2020) evaluates his grammars over the same sentences as were seen in the training set. However, Bender et al. (2014) and Zamaraeva et al. (2019a) evaluate inferred grammars over held-out portions of the Chintang dataset. Here I use the same dataset of Chintang as one of my development sets (although I divided the data differently in order to create ten even folds for cross-validation), so I use Zamaraeva et al. 2019a as a point of external comparison.

By creating lexical items for determiners, adpositions, coordinators and negation words, I doubled the number of test items for which the inferred grammars can analyze each word, compared to Zamaraeva et al. (2019a). This is critical as the grammar has no chance at syntactic analysis if lexical analysis fails. My lexical coverage ranges from below 10% for Lezgi, Meithei and Wambaya and above 50% for Abui (53.19%), but averages 20% across the development languages.[10] This means that about 20% of sentences in each test set have a chance at syntactic analysis in the first place.

The next thing to consider is what portion of the sentences for which the grammar can analyze each word can be analyzed syntactically. Zamaraeva et al.'s inferred Chintang grammar parsed 30% of the sentences it had lexical coverage for. My inferred grammars have significantly closed that gap, parsing 84% of the Chintang sentences that had lexical coverage and 67% of the items with lexical coverage on average across all of the development languages.

The most important metric is the proportion of test items the grammar parses correctly. On the development languages, the number of sentences with the correct predicate-argument structure ranges from 0% to 10%. The number of sentences with correct predicate-argument structure for Chintang is more than double what it was for Zamaraeva et al. (2019a) and the introduction of semantic features increases the quality of these parses. Next, consider the

[10]Here and throughout this section, I use macro-averages weighting each language equally.

proportion of sentences that had the correct predicate-argument structure as a subset of those that parsed. One of my goals is to minimize the number of incorrect parses in the grammar, so this proportion should be as high as possible. Here, my system has more spurious coverage than the system of Zamaraeva et al. (2019a), which correctly parsed 47% of the sentences that it parsed. BASIL produced parses with correct predicate-argument structure for only 19% of the sentences it parsed. However, unlike Zamaraeva et al. (2019a) who did not infer syntactico-semantic features, BASIL produced parses with correct predicate-argument structure and features for 9% of the sentences it parsed.

Finally, measuring ambiguity provides a way of understanding how many incorrect or redundant parses are produced by the grammar. Ideally, this should be minimal, as in Wambaya, for which my inferred grammars average four parses per sentence. However this average increases when there are multiple analyses for a morphological or syntactic phenomenon, some of which are valid and some of which are not. I will go into this in more detail on this in Chapter 7 when I compare the ambiguity of the inferred grammars with baseline inference systems. At this stage, I simply note that there is an inherent trade-off between coverage and ambiguity in inferred grammars, and during data-driven development I attempted to minimize increases in ambiguity, as I increased coverage.

5.5 Summary

In this chapter I described the languages and datasets that I used during development and assess BASIL in terms of how it performs on them. I primarily used nine development languages from seven language families, but at times consulted others for a total of 27 languages from 19 families, in order to make my inference system as robust to cross-linguistic variation as possible. I showed that the nine development languages tested most of the phenomena targeted by the inference system and performed well in terms of producing grammars that handle those phenomena correctly. With this performance at the end of development, I turn to evaluation on held-out languages to determine how well this inference system generalizes to previously unconsidered languages.

Chapter 6

EVALUATION METHODOLOGY

The goal of this dissertation is to automatically infer grammars from IGT for low resource and endangered languages. For this reason, I evaluate my inference system, BASIL, by inferring grammars for held-out languages and testing the performance of those grammars on held-out test items in the language. In this chapter I describe my evaluation methodology including a description of the metrics (§6.1), the baseline inference systems I use for comparison (§6.2) and the languages with which I test the cross-linguistic generalizability of my inference system (§6.3).

I evaluate BASIL using the full end-to-end pipeline described in Chapter 3 and repeated in Figure 6.1 below. For each language, I begin with a Toolbox (SIL International, 2015) or FLeX (Rogers, 2010) corpus, which I converted to the Xigt data model (Goodman et al., 2015) and enriched with projected POS and dependency tags using INTENT (Georgi, 2016). After dividing the data into ten 90%-10% train-test splits (described in Section 6.4), I use the training set to infer a grammar specification with BASIL which I input to the Grammar Matrix customization system (Bender et al., 2002, 2010) to produce a grammar. Then, I load the grammar into parsing and treebanking software to test its performance over the 10% test set as described in the following section. For parsing, I target the morpheme segmented line of the IGT data, leaving the integration of a morpho-phonological analyzer to future work.

6.1 Evaluation Metrics: Parsing and Treebanking

I assess the inferred grammars by parsing each sentence in the test set using five metrics: lexical coverage, parse coverage, correct predicate-argument structure, correct predicate-argument structure and features and ambiguity. *Lexical coverage* is the number of sentences

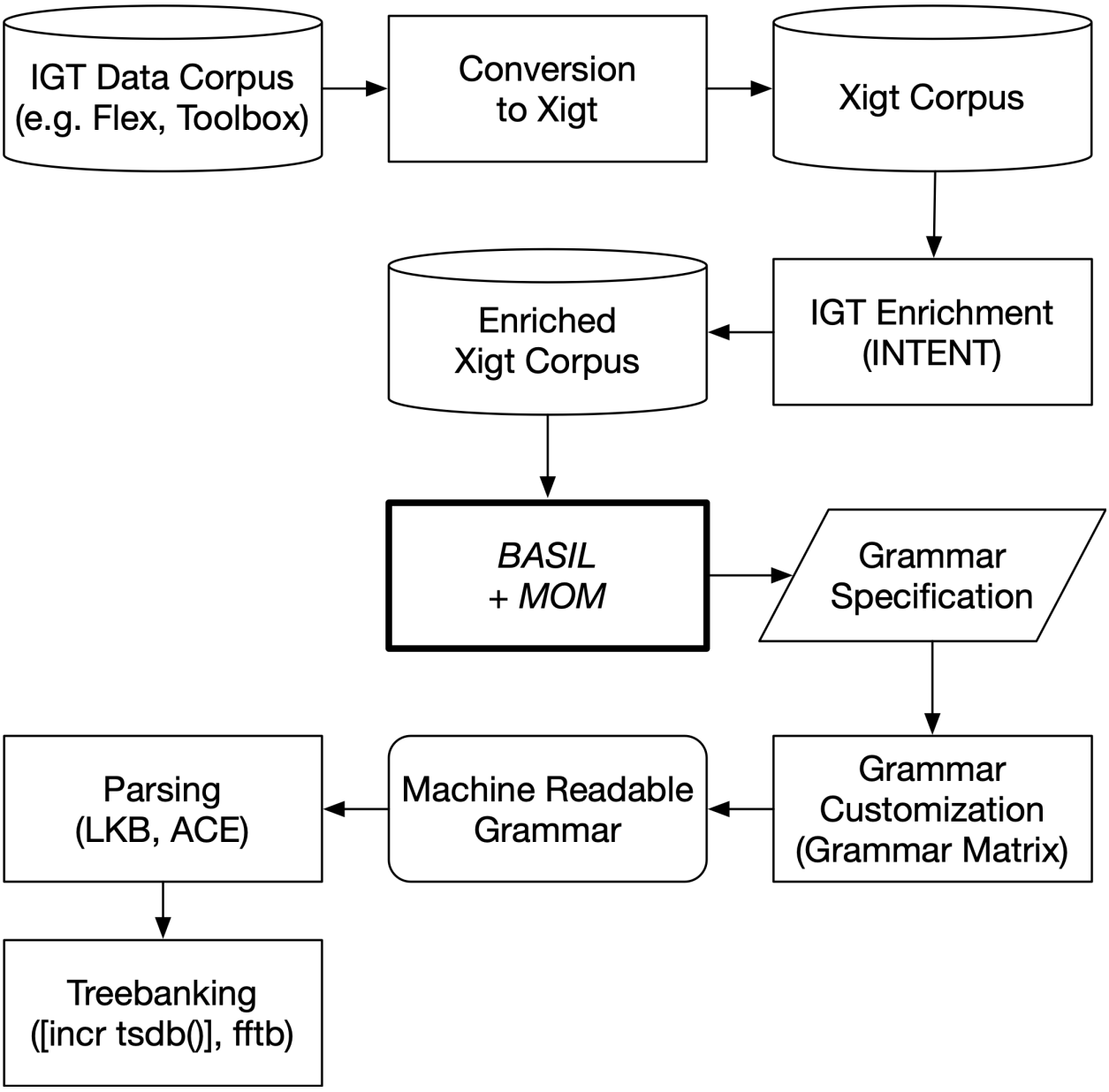


Figure 6.1: AGGREGATION Pipeline

for which the grammar has an analysis for each word in the sentence, meaning that the lemma is in the lexicon and the morphology is accounted for by the grammar's inflectional rules. *Parse coverage* is the number of sentences the grammar parses, meaning that words unify with the necessary syntactic rules to form a sentence. However, each parse does not necessarily have the correct semantic representation, so I consider two types of *correct coverage*. The first, *correct predicate-argument structure* is the number of sentences for which the grammar produces a parse whose semantic representation includes the correct predications, which are linked to the correct arguments. The second is *correct predicate-argument structure and features*, which takes this a step further by including the correct semantic features for each argument. Finally, *ambiguity* is the average number of results per sentence that parses.

After inferring a grammar from the training data, I use the ACE parsing software (Crysmann and Packard, 2012) to parse each sentence in the test dataset.[1] For each sentence, ACE outputs whether the grammar had a lexical analysis for each word in the sentence, from which I calculate *lexical coverage*. If each word has an analysis and the grammar accepts the sentence as grammatical, ACE returns a result which includes the syntactic parse tree and the corresponding semantic representation (illustrated in Figure 6.2), and on this basis, I calculate *parse coverage*. In many cases the grammar contains *ambiguity*, returning multiple parses per sentence, and I report this as the average number of results for sentences that parse.

The process of finding the *correct coverage* is more involved. After parsing the test sentences with ACE, I use the Full Forest Treebanking software (FFTB; Packard, 2015) to analyze the parses and determine if the parse is correct. Using FFTB I examine the lexical and syntactic rules in the parse forest to identify any trees that represent an appropriate syntactic parse for the sentence. I then inspect the corresponding semantic structure by looking at the predicate-argument structure as well as the semantic features on each argument.

[1]For links to ACE and other software used for evaluation, see Appendix A.

Consider the syntactic and semantic representations in Figure 6.2 which were produced by an inferred grammar for the Matsigenka sentence in (21).

(21) Ikamagutakerotyo.

i-kamagu-t-ak-i=ro=tyo

3mS-look-EPC-PERF-REALIS=3fO=AFFECT

'He looked at it.' [mcb] (Michael et al., 2013)

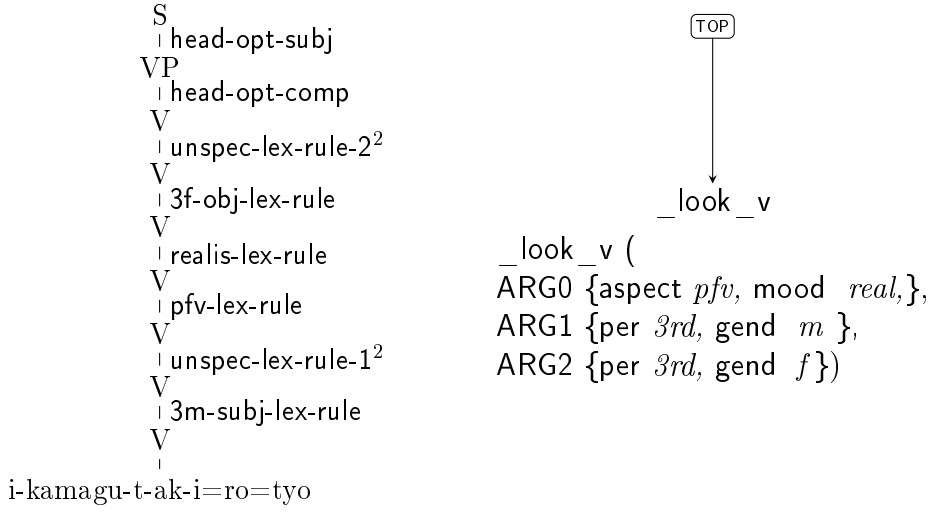


Figure 6.2: The best semantic representation (right) and its corresponding syntax tree (left) produced by the inferred grammar for the sentence in (21)

Sentence (21) has only one word[3] but includes three semantic arguments: an event and two entities. For this reason, the tree in Figure 6.2 contains a series of lexical rules (the

[2]I use 'unspec' as a naming convention for lexical rules that do not add any morpho-syntactic or morphosyntactic features.

[3]Clitics are syntactically free, phonologically bound elements (Zwicky and Pullum, 1983). Although Michael et al. use an = to indicate two clitics (*=ro* and *=tyo*), the inference system analyzes them as affixes, due to the absense of white space between the clitics and the verb they attach to.

nodes labeled as V) and two syntactic rules (object dropping, labeled by VP, and subject dropping, labeled by S).[4] The semantic dependency contains only one predicate, because there is only one predicate-contributing element in the sentence: the verb *kamagu* 'look'. That predicate, however, has three arguments. First is the event argument (ARG0), which is marked with perfective aspect and realis mood. Next there is the semantic argument (ARG1) corresponding to the unexpressed subject, which is marked with third person and masculine gender, and third is the semantic argument (ARG2) corresponding to the unexpressed object, marked with third person and feminine gender.

I consider the semantic representation in Figure 6.2 to have the correct predicate-argument structure because it contains all of the predications that should be in the semantic representation and no additional, incorrect predications, and because the predication has the correct arguments: an event and two entities. I consider the semantic features in Figure 6.2 to be correct because they reflect all of the semantic features that A) are in the IGT and B) my inference system targets: BASIL only targets PNG and TAM features, so those are the only ones I expect. The semantic representation does not reflect the affective meaning because BASIL does not extract stance features.[5] I assume that EPC marks an epenthetic consonant, and does not contribute any semantic feature. If any PNG or TAM features in the IGT were not reflected in the semantic representation or if any features in the semantic representation did not match those in the gloss or implied by the translation,[6] I would classify this parse as having the correct predicate-argument structure, but not as having the correct predicate-argument structure and features.

Although treebanking parses for correctness is an established practice (see inter alia

---

[4]The treatment of these arguments as a dropped subject and object is consistent with the analysis set forth by Inman (2015) of pronoun incorporation in Matsigenka.

[5]This gloss is not explicitly defined by Michael (2008), but from his discussion around such examples, I believe that this refers to stance.

[6]For example if there was a masculine feature for the semantic representation of a noun that is not glossed with a masculine gram, but the corresponding word in the translation was the masculine pronoun 'he', I would not consider this incorrect. Such an occurrence would likely be the result of inconsistent glossing where that pronoun was glossed as masculine elsewhere in the corpus.

	Abui [abz]	Chintang [ctn]
Correct Parse	0.5714	0.7843
Matching Pred-Arg Structure	0.5714	0.7843
Matching Features	0.5714	0.5882
Exact Match MRS	0.5143	0.5882

Table 6.1: F1 scores for inter-annotator agreement on treebanked coverage for the Abui and Chintang datasets

Oepen et al., 2002; Flickinger et al., 2017), assessing the accuracy of semantic representation for languages that one doesn't speak natively and isn't an expert on is a challenging task. For example, it can be hard to know if some locative constructions are core arguments of the verbs or if they are modifiers. Furthermore, glossing conventions vary from linguist to linguist and with limited familiarity with the datasets, one must make guesses as to implications of some grams and the ambiguous cases I might encounter are difficult to anticipate without first engaging with the data. Therefore, we[7] established a practice of consulting both the gloss line and the translation line as the translation line might omit or add some semantic information compared to the gloss line, but the gloss line may be ambiguous with regards to which words are arguments of which and this can be learned from the translation. After developing basic guidelines after discussing some specific examples from the development datasets, Emily M. Bender and I independently treebanked one fold from each of the Abui and Chintang datasets.

Following the methodologies set forth by Dridan and Oepen (2011) for semantic evaluation and Bender et al. (2015) for inter-annotator agreement, with some adaptations to target my task-specific goals, I calculated inter-annotator agreement for the treebanked results of the two development sets, which I present in Table 6.1. Dridan and Oepen (2011) propose an Elementary Dependency Match (EDM) score calculated from multiple parts of the semantic representation. I used their EDM_{na} metric for naming and argument identification,

[7] This is in consultation with Emily M. Bender, based on her previous treebanking work in Bender 2008c, Bender et al. 2014 and Zamaraeva et al. 2019a.

and added a metric for semantic features. Following from Bender et al. (2015) I assess inter-annotator agreement (IAA) for these metrics by calculating the F1 score for these metrics between the two annotators. These F1 scores are shown in Table 6.1 as Matching Pred-Arg Structure and Matching Features. To situate these measures I also present F1 scores for IAA for whether the parse was considered correct at all (Correct Parse) and whether the two semantic representations matched exactly (Exact Match MRS).

The F1 score for correct parse is the same for matching predicate-argument structure, which shows that when we agreed that there was a parse with an acceptable predicate-argument structure, we also agreed on what that predicate-argument structure should be.[8] For example, we agreed on whether certain sentences should be analyzed as having a transitive or intransitive verb. Disagreements were often due to one annotator interpreting something as a modifier instead of an argument (the inferred grammars do not handle modifiers, so these parses would be rejected) or whether sentence fragments should be accepted or rejected, given an otherwise correct semantic representation.[9]

The F1 scores for matching features in Abui matched the F1 for matching predicate-argument structures, showing again that if we both considered a parse acceptable we agreed on what the predicate-argument structure should be. The slightly lower F1 for Exact Match MRS for Abui is due to a slightly different but equally acceptable predication for the verb in one sentence: leave.for_v_rel vs. leave.for-or-step_v_rel, where the second represents two possible meanings of the verb. For Chintang the feature agreement is lower than predicate-argument structure agreement. For this language the grammars have a great deal of ambiguity in the lexical rules. In many cases, it was not possible to find a parse that had all of the correct features, and we chose parses with different subsets of correct and incorrect features.

---

[8]This does not necessarily mean that we chose the same syntactic parse, as spurious ambiguity may result in multiple syntactic structures producing the same semantic representation.

[9]We resolved to accept semantic representations where a locative is treated as an argument only when it could be interpreted as the goal of a verb of motion. In the case of sentence fragments, we decided to accept analyses where the grammar assigned an S to a VP sentence fragment (i.e. with a null subject), but to reject any analysis of an item where the translation in the IGT indicated it was an NP fragment.

After discussing our disagreements, we extended our definitions of correct parses.[10] I use this methodology to evaluate the grammars produced by BASIL as well as three baselines, which I describe in the next section, for each of the evaluation languages.

6.2 Baseline Systems

A core contribution of this dissertation is the inference of syntactic properties from IGT data and the integration of these syntactic properties with the lexical and morphological properties inferred by MOM (Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017). For this reason, I compare the performance of my inference system with three baseline systems that that are morphologically and lexically robust with respect to accounting for the training data, but are syntactically and morpho-semantically naive. Each baseline contains lexical entries and morphological rules from MOM for nouns and verbs. Although MOM extracts morpho-syntactic features for nouns and verbs, including them in the lexicon and morphological rules, inference is required to define them appropriately in the grammar specification. The Grammar Matrix will not produce a working grammar if features in the lexicon and morphology are not defined, so I disabled feature extraction in MOM for the baseline systems.

Table 6.2 enumerates the syntactic features in my baseline grammar specifications. Each baseline has a different way of determining how to specify each feature BASIL accounts for. The first baseline system, BROAD-COV, posits the specifications that I expect to result in a grammar with the broadest coverage. The second baseline, TYP, posits the specifications that are typologically most common, according to the information available in typological resources. If a typologically-most frequent choice could not be made, I select the feature value at random if it is required by the Grammar Matrix customization system, and omit it otherwise. Finally the third baseline, RAND, selects a value for each specification at random. The baseline system makes a different random choice for each feature every time it is run, reducing the individual effect of each random choice on the over-all results.

[10]The final treebanked results for the development languages are presented in Chapter 5.

	BROAD-COV	TYP	RAND
word order	free	sov	RC
has determiners	yes	yes	RC
noun-det order	RC	noun-det	RC
det required	optional	RC	RC
has auxiliaries	no	no	no
verb valence	trans	RC	RC
case frame	none	none	none
s coordination	asyndedeton		RC
vp coordination	asyndedeton		RC
np coordination	asyndedeton		RC
n coordination	asyndedeton		RC
subj-drop	all	all	RC
obj-drop	all		RC

Table 6.2: Grammar ppecifications for syntactic phenomena for three baseline systems. RC indicates a random choice

The Grammar Matrix requires specifications for word order, whether the language has determiners, the order of determiners with respect to the noun, whether each noun class requires a determiner, whether or not the language has auxiliaries and whether each verb class is transitive or intransitive. The specifications for determiners and auxiliaries only have an effect on the grammar's performance if the orthographies are also added to the lexicon. It is possible to extract determiners from the training data naively by adding all words with the POS tag 'det', 'dem', 'quant', 'num' or 'def',[11] so the baselines add the orthographies for each word in the training corpus with these POS tags to the lexicon. Auxiliaries, however, are typically tagged as verbs and are extracted as such by MOM. Without using syntactic inference to identify them, specifying that the language has them and where they attach to the verb won't have any effect on the coverage of the grammar, so all of the baselines assume that the language does not have auxiliaries.

For the BROAD-COV baseline I posit free word order, determiners that are optional for

[11]My work on the development languages suggested that these tags should serve reasonably well for identifying determiners in the data.

each noun class and specify all verbs as transitive (but able to drop their objects). Intuitively these specifications will result in the highest likelihood that a sentence will parse, therefore maximizing parse coverage. They do not, however, ensure the correctness of these parses and are likely to result in incorrect parses and increased ambiguity. The order of the determiner with respect to the noun will result in higher coverage for the language only if the 'correct' choice is made for the language, but without syntactic inference, the baseline has no way of determining what that is. Therefore, it selects a determiner-noun order at random, but because it makes a different random choice for each evaluation fold, it is possible for this random choice to be correct for the language as often as not. For the TYP baseline, I posit SOV word order and that determiners are possible and occur after nouns based on they surveys of Dryer (2013c) and Dryer (2013d), respectively. The typological literature doesn't help to determine whether a particular verb class is more or less likely to be transitive, so TYP makes a random selection for each verb. For each of these specifications, the RAND baseline selects randomly from the possible choices.

Negation, case system, coordination and argument optionality have down-stream dependencies in order to function in the grammar. For any negation strategy I could select, a word or lexical rule must be identified as the negation morpheme, so I leave this phenomenon out of the baseline systems. Similarly, even if I posit an over-arching case system, the grammar can only enforce it if case features are also inferred in the noun morphology. Thus all three baselines posit that the language has no case system, which allows any noun to be a subject or object. Any coordination strategy would require an inferred coordinator, except for asyndedic. For the BROAD-COV baseline, I add an asyndedic coordination strategy for each constituent type, which allows coordination between each of the constituent types supported by the Grammar Matrix without any overt coordination marker. The RAND baseline randomly adds or doesn't add an asyndedic strategy for each constituent type (i.e. a virtual coin toss determines whether to add asyndedic coordination for sentences and another determines whether to add it for nouns and so on for each constituent type). Finally, the most common coordination strategy from the survey of Drellishak (2004) is monosyndedic. However, for

	Language	iso	Family	Corpus
1	Arapaho	arp	Algic	Cowell 2018
2	Hixkaryana	hix	Cariban	Meira 2020
3	South Efate	erk	Austronesian	Thieberger 2006b
4	Titan	ttv	Austronesian	Bowern 2019
5	Wakhi	wbl	Indo-European	Kaufman et al. 2020

Table 6.3: Languages used in held-out evaluation

the grammar to model monosyndedic coordination an orthography for the coordinator would be required. Because identifying the orthography of this coordinator and e.g. whether it is an affix or an independent word would require inference, TYP does not add a coordination strategy to the grammar specification, and the resulting grammar will not support coordination. Finally, argument optionality in the Grammar Matrix can be possible for all verbs, possible for some verbs, or not possible. If it is possible for some verbs, it would be necessary to infer which verbs. Therefore the baselines choose only between all and none. BROAD-COV allows both subject and object dropping for all, RAND makes a choice for each at random, and TYP allows subject dropping for all verbs based on the survey of Dryer (2013b). I was not able to find a typological survey that indicates whether object dropping is allowed in more languages than not, so TYP leaves this unspecified (and therefore not possible in the resulting grammar).

6.3 Held-out Languages

To test how well BASIL generalizes to new languages, I acquired datasets for five additional languages, which I did not consider during development and which are geneologically and geographically varied from the development languages. These languages are listed in Table 6.3 and the locations where they are spoken are shown on the map in Figure 6.3.

I pre-processed each dataset by filtering out ungrammatical examples (examples marked with a *) and removing duplicates. For held-out evaluation, I selected only languages with

	Language [iso]	Source Type	Size	POS tags in source
1	Arapaho [arp]	toolbox	5000	yes
2	Hixkaryana [hix]	toolbox	5749	yes
3	South Efate [erk]	toolbox	1875	yes
4	Titan [ttv]	toolbox	1799	yes
5	Wakhi [wbl]	FLeX	683	yes

Table 6.4: Source, size and presence of POS tags for the held-out datasets

POS tags in the original dataset.[12] This information as well as the type of source dataset and the number of IGT after filtering are summarized in Figure 6.4. In this section, I provide a brief description of each language and dataset.

Arapaho

Arapaho [arp] is an Algonquian language of the Algic language family with only about 250 native speakers in the United States (Cowell and Moss Sr, 2011). The dataset I use is a subset of a larger Arapaho documentation project and contains over 60,000 IGT (Cowell, 2018). In order to work with a more manageable quantity,[13] I randomly selected 5,000 IGT (sampling only from the fully glossed IGT, excluding duplicates) to use for evaluation. The corpus includes elicitations and transcribed conversations, among other genres.

Hixkaryana

With about 1,200 speakers, Hixkaryana [hix] is a Cariban language in the Waiwai subgroup (Eberhard et al., 2019). The dataset is quite large and the glossing is fairly complete (Meira, 2020). After removing some IGT with incomplete glosses as well as ungrammatical examples and duplicates, the corpus contains almost 6,000 IGT.

[12]For Titan, POS tags were not already included in the IGT data, but a lexicon with POS tags was included in the Toolbox project.

[13]Morphological inference is extremely time consuming and memory-expensive for a dataset of this size, especially for a morphologically complex language.

Figure 6.3: Map of the coordinates of languages used in evaluation

South Efate

South Efate [erk] is a Vanuatu language of the Austronesian language family and is spoken by about 6,000 people on the Efate island in the Republic of Vanuatu (Thieberger, 2006a). The corpus contains about 3,000 IGT (Thieberger, 2006b), and after removing IGT with incomplete glossing, ungrammatical examples and duplicates I used a dataset of just under 1,900 IGT.

Titan

Titan [ttv] is also an Austronesian language, and while it and South Efate are both Oceanic, Titan is grouped as a language of the Admiralty Islands while Efate is Central-Eastern Oceanic. The various dialects of Titan are spoken by approximately 3,500-4,500 people (Bowern, 2011). This corpus contains just under 1,800 IGT after filtering for glossing, gramaticality and duplicates (Bowern, 2019). The Toolbox corpus does not contain POS tags, but it is accompanied by a toolbox lexicon. Using the lexicon, I added POS tags to the corpus by looking up each word in the dictionary and selecting the POS for the first entry. Because of lexical ambiguity, this approach introduces some incorrect POS tags to the corpus, but creates a better starting point for grammar inference than using only the POS tags provided by INTENT.

Wakhi

Wakhi [wbl] is an Iranian language of the Indo-European language family and is spoken primarily in Afghanistan and has a growing speaker population of about 17,000 (Eberhard et al., 2019). The dataset is small, containing only about 700 IGT after filtering (Kaufman et al., 2020). However, it is thoroughly glossed and is made up primarily of elicitations targeting specific syntactic phenomena.

6.4 Cross Validation

The use of held-out languages for evaluation is key to testing BASIL's cross-linguistic generalizability. For each of these languages, I also test the inferred grammars on held-out test items to verify the generalizability of the grammars to unseen forms. To accomplish this, I parse every sentence in the corpus using ten-fold cross validation. I ran evaluation ten times, using one fold for evaluation and the remaining nine to infer a grammar. I tested each inferred grammar on the held-out portion of the data by parsing the sentences in the held-out set.

6.5 Summary

In this chapter I presented my methodology for evaluating BASIL on held-out languages. I use five metrics to assess the inferred grammars, which are designed to evaluate the grammars in terms of how many sentences they can analyze, how correctly they can analyze them and how much ambiguity they contain. I compare the inferred grammars with three baseline systems which use the same morphological and lexical inference system, but are syntactically naive, in order to create meaningful comparison for the contribution of BASIL's syntactic inference. Finally, while I consulted a number of languages during development, I evaluate my system on five additional languages which I have not previously considered. The results of this evaluation will be described in the following chapter.

Chapter 7
RESULTS AND ANALYSIS

I applied the evaluation methodology described in Chapter 6[1] to datasets for five held-out languages: Arapaho [arp], Hixkaryana [hix], South Efate [erk], Titan [ttv] and Wakhi [wbl].[2] In this chapter, I present the results of that evaluation, which demonstrate the degree to which BASIL generalizes to languages that were not considered during system development.[3] I analyze the results from three perspectives, first comparing system performance between languages, then comparing the baselines across each evaluation metric and finally situating each metric with respect to the others. I follow my discussion of the results with an analysis of the sources of error, highlighting areas in which the system could be improved.

7.1 Results

Using the methodology described in Chapter 6, I performed ten fold cross-validation on the evaluation languages for the BASIL system as well as three baselines: BROAD-COV, which uses grammar specifications expected to parse as many sentences as possible; TYP, which uses the typologically most likely specifications; and RAND, which selects specifications for each grammar at random.[4] The results for each system's performance on each language are presented in Tables 7.1-7.5 as follows. Lexical coverage, or the percentage of test items (sentences) in which each word was accounted for by the grammar, is in Table 7.1. Parse

[1]Code for this evaluation methodology and instructions for reproducing the results can be found at https://git.ling.washington.edu/agg/repro/basil-2020.

[2]For descriptions of these languages and the datasets I used for evaluation, see Section 6.3.

[3]To test how well BASIL generalizes I froze development of the inference system prior to evaluation. However, some minor bugs in the inference system regarding data handling prevented the pipeline from running. The changes that I made to resolve these are documented in Appendix B.

[4]For more detail on the baseline grammar specifications and how they were constructed, see Section 6.2.

Language	BASIL	BROAD-COV	TYP	RAND
Arapahoe [arp]	3.64	3.52	3.64	3.18
Hixkaryana [hix]	38.09	36.01	35.88	35.92
South Efate [erk]	12.80	13.55	14.29	13.17
Titan [ttv]	13.56	19.40	20.34	19.40
Wakhi [wbl]	39.68	29.72	31.48	31.04

Table 7.1: Lexical coverage for held out languages as a percentage of the total number of test items across ten folds

coverage, or the percentage of test items which parse, is in Table 7.2. Coverage with correct predicate-argument structure, or the percentage of test items that parse and at least one parse produces a semantic representation with the correct predications and relations with respect to their arguments, is in Table 7.3. Coverage with correct predicate-argument structure, where the semantic representation also contains all of the appropriate semantic features, is in Table 7.4. Finally, ambiguity, or the average number of parses per sentence that parsed, is shown in Table 7.5.[5]

Because the process of treebanking the results to determine whether a parse is correct is very time consuming, for each language I treebanked n folds such that the number of parsed sentences in n folds is greater than 100. Thus the results for lexical coverage, parse coverage and ambiguity are averages across ten folds, while the results for coverage with correct predicate-argument structure and coverage with correct predicate-argument structure and features are averages across n folds where n is given in Table 7.6.

The results in Tables 7.1-7.5 show 5 metrics across 4 systems and 5 languages. In order to assess the meaning of these results, I begin with a comparison of the results for the languages with respect to each other (§7.1.1), followed by a comparison of BASIL's performance with respect to the baselines (§7.1.2). With this foundation, I then compare each metric for the

[5]For more detail on how these metrics are calculated, see Section 6.1.

[6]For Titan I report a correct coverage that is higher than the parse coverage for the TYP and RAND baselines. This is possible because there were more parsed items per fold in the 6 folds I treebanked than in the remaining 4.

Language	BASIL	BROAD-COV	TYP	RAND
Arapahoe [arp]	3.04	3.06	0.50	0.26
Hixkaryana [hix]	34.18	31.28	2.80	1.25
South Efate [erk]	6.77	9.81	0.27	0.27
Titan [ttv]	10.34	16.18	0.06	0.17
Wakhi [wbl]	30.31	24.89	10.25	3.22

Table 7.2: Parse coverage for held out languages as a percentage of the total number of test items across ten folds

Language	BASIL	BROAD-COV	TYP	RAND
Arapahoe [arp]	0.17	0.20	0.00	0.03
Hixkaryana [hix]	2.26	2.26	1.57	0.52
South Efate [erk]	0.38	0.31	0.00	0.00
Titan [ttv][6]	0.28	0.65	0.09	0.19
Wakhi [wbl]	14.20	12.75	2.61	0.58

Table 7.3: Coverage with correct predicate-argument structure as a percentage of the total number of test items across n folds

Language	BASIL	BROAD-COV	TYP	RAND
Arapahoe [arp]	0.09	0.06	0.00	0.00
Hixkaryana [hix]	0.00	0.00	0.00	0.00
South Efate [erk]	0.15	0.00	0.00	0.00
Titan [ttv]	0.19	0.00	0.00	0.00
Wakhi [wbl]	5.80	0.58	0.00	0.00

Table 7.4: Coverage with correct predicate-argument structure and semantic features as a percentage of the total number of test items across n folds

Language	BASIL	BROAD-COV	TYP	RAND
Arapahoe [arp]	145	936	4	3
Hixkaryana [hix]	5642	15596	2	6
South Efate [erk]	126379	9759	2	4
Titan [ttv]	595	6201	2	1
Wakhi [wbl]	10	26	1	2.5

Table 7.5: Average number of results per parsed sentence for across ten folds

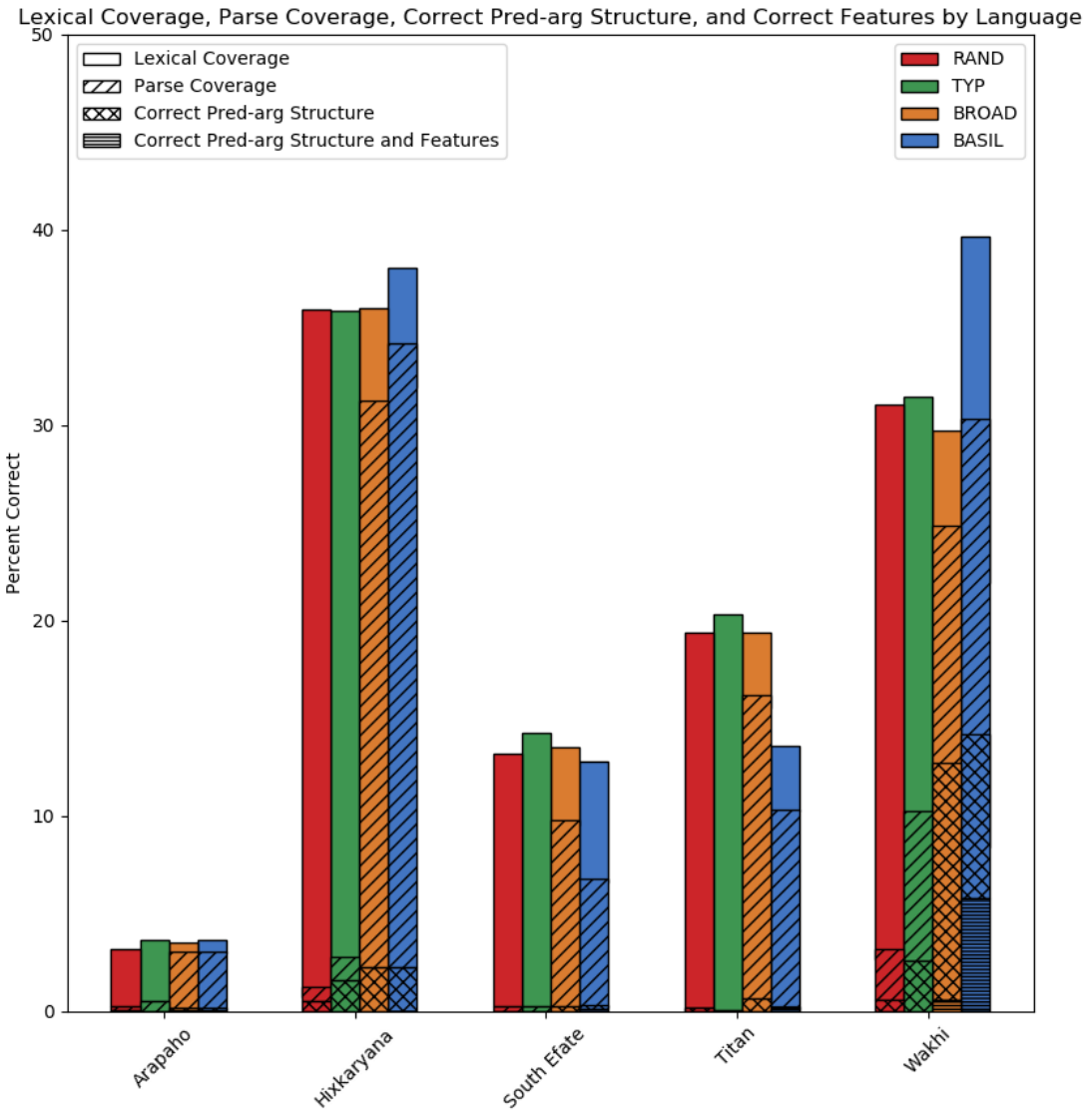


Figure 7.1: Results for Held-out Languages

Language	folds (n)	Parsed sentences in n folds	Total sentences in n folds
Arapahoe [arp]	7	109	3500
Hixkaryana [hix]	1	198	575
South Efate [erk]	7	110	1504
Titan [ttv]	6	110	1080
Wakhi [wbl]	5	115	345

Table 7.6: Number of sentences treebanked across n folds for each held-out language

specific languages and systems with the other metrics to assess how well BASIL meets the goal of increasing correct coverage without increasing spurious parses (§7.1.3).

7.1.1 Comparison across languages

There is a great deal of variation in how well any of the systems did at inferring grammars that can parse held-out sentences for each language, as illustrated by the graph in Figure 7.1. Coverage for Arapaho was very low, at roughly 3% lexical coverage for each system and similar parse coverage for BASIL and BROAD-COV. Across all systems, Hixkaryana and Wakhi had significantly higher lexical and parse coverage, exceeding BASIL's performance on most of the development languages.[7] South Efate and Wakhi fall between these two extremes.

The correct coverage for the various languages is more consistent across languages with Wakhi as an outlier. For Wakhi, BASIL achieves correct predicate-argument structure for 14.20% of the items in the test set and correct predicate-argument structure and features for 5.8% and the BROAD-COV baseline achieves 12.75% correct predicate-argument structure, while the remaining languages have much lower correct coverage across systems.

Finally, the ambiguity (or average number of parses per parsed item) for these languages is quite low for Wakhi, on the order of tens, and extremely high for South Efate, on the order of 100,000.[8] Ambiguity is common in natural languages and is expected in inferred

[7]For BASIL's performance on development languages, see Section 5.4.

[8]For languages with a great deal of ambiguity, the parser (ACE; Crysmann and Packard, 2012) sometimes

grammars which are less carefully constrained than hand-written grammars. I will provide more detail on the causes of ambiguity in the inferred South Efate grammar in Section 7.4.

Overall, the systems performed best on Wakhi across the metrics I considered. Performance for Hixkaryana, South Efate and Titan was somewhat lower, with coverage for Arapaho being the lowest. The variation in performance can be due to both unusual or difficult to infer facts of the language or may be the result of characteristics of the IGT datasets. In the error analysis portion of this chapter (§7.2 and §7.3), I will shed some light on this variation. However, in the mean time, understanding the variation in performance across languages will contextualize the comparison of BASIL with the baseline systems.

7.1.2 Comparison across systems

To understand the impact of *syntactic inference* on automatic grammar generation, I compare BASIL with three baselines that use the same morphotactic and lexical inference system (MOM; Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017) as BASIL, but must specify the syntactic portions of the grammar specification through some other means. The BROAD-COV system uses the specifications that are expected to result in grammars that have the broadest coverage, or can parse the most sentences, whether correctly or incorrectly. TYP uses the typologically most common specification and RAND uses a random choice.[9] Each of these baselines uses a random choice for at least one specification, where no clear determination could be made for broad coverage or typological frequency, so ten-fold cross validation (given that a new random choice is made when specifying the grammar for each fold) is important to reduce the effect of chance on the overall performance of each baseline.

Because the same morphotactic and lexical inference system was used for the baselines as for BASIL, the lexical coverage across systems is roughly comparable. For some languages, the baseline lexical coverage is lower because the baselines can only use POS tags to identify

exceeds its memory limit. For this reason, the true ambiguity for the languages with more ambiguity may be even higher than reported.

[9]Details on the individual specifications can be found in Section 6.2.

lexical items, while BASIL uses additional heuristics. For other languages, it is slightly higher because BASIL strategically excludes ditransitive and clausal complement-taking verbs (which it would not handle correctly) from the lexicon.[10] Additional variation in the lexical coverage across systems can be attributed to variations in the morphological graph, which is different for each baseline due to the random assignment of verb valence for the TYP and RAND baselines which influences which lexical classes are merged.[11]

A larger and more meaningful difference between the systems is seen in parse coverage. Here, the TYP and RAND baselines have significantly lower coverage than BASIL and BROAD-COV. While the TYP baseline has a better chance of using the correct value for each individual specification, it will not necessarily be correct for enough phenomena to produce a grammar that can parse simple sentences: For example, even if the order of verbs with respect to subjects and objects is correct, sentences with determiners won't parse if the determiner-noun order is incorrect. By design, the BROAD-COV system has the highest parse coverage, often outperforming BASIL; however, without syntactic inference this coverage could be spurious, so we must consider correct coverage.

I measure correct coverage in two ways: first using the percentage of test items that parse with the correct predicate-argument structure and second with test items that parse with the correct predicate-argument structure and semantic features.[12] Again, the TYP and RAND baselines under-perform the other two systems, as there is a relatively low chance that their specifications will correctly model any given language. In terms of correct predicate-argument structure, BASIL outperforms BROAD-COV for South Efate and Wakhi, while BROAD-COV does better for Arapaho and Titan. They tie on Hixkaryana. As BROAD-COV is designed

[10]More specifically, MOM adds a '∼' to the orthography of any verb that was not inferred as intransitive or transitive, so that it can be used as an example in the morphological graph, but the resulting grammar will not parse it. Strictly speaking these verbs are in the lexicon, but the change to their orthography prevents them from being used to parse any real words in the language. This comes from Zamaraeva et al. 2019a.

[11]For more information on valence in the baselines, see Section 6.2. For details on how transitivity and morphotactic patterns interact in lexical inference, see Section 4.2 and in Zamaraeva et al. 2019a.

[12]A detailed discussion of how I assess correctness is given in Section 6.1.

to maximize coverage, it allows asyndedic coordination for each language, which allowed it to parse sentences for languages where BASIL failed to infer this strategy.[13] The second way I measure correct coverage is by considering how many sentences not only have the correct predicate-argument structure, but also the correct semantic features. For this metric, BASIL outperforms all of the baselines, because they are not able to posit semantic features at all. For a couple of sentences, BROAD-COV had the 'correct features' because the semantic representation shouldn't include any features at all; however, these cases are relatively rare.

So far, I have shown that BASIL and BROAD-COV out-perform the other two baselines in parse coverage and correct predicate-argument structure, while BASIL out-performs all of the baselines in correct predicate-argument structure and semantic features, as illustrated by Figure 7.1. The last thing to consider is how much ambiguity each of the grammars contain. TYP and RAND produced grammars with very little ambiguity. These grammars only parsed extremely simple sentences, so this is not surprising. BROAD-COV was designed to maximize coverage, but this comes at the cost of increased ambiguity. For example, positing free word order for each language will ensure that all word orders will parse, but will allow parses where the wrong constituents are identified as subjects and objects. As a result, the BROAD-COV baseline has significantly higher ambiguity than BASIL for all languages except for South Efate.

Overall, BASIL outperformed the baselines by having higher coverage than the TYP and RAND systems without the added ambiguity contributed by BROAD-COV. In addition, the semantic representations it produced were richer, in that it had the most parses with correct semantic features. In addition to considering these five metrics, it is important to look at how the results for each metric compare across systems not just as raw scores but relative to the other metrics, as I will show in the next section.

[13]More detail on this error in inference is given in Section 7.3.

7.1.3 Comparison across metrics

Another useful way to look at these metrics is in terms of what they show about how often the grammars parse a test item incorrectly or less completely than possible. I assessed this in part by considering ambiguity, which shows how many spurious parses were produced for a given sentence or test item, but the degree of spurious analyses in the grammar can also be assessed by considering the number of items that fit one evaluation criterion but not another. Figure 7.1 provides a visualization of these differences.

In an ideal scenario, if every word in a sentence could be analyzed by the grammar (shown by lexical coverage), the sentence as a whole could also be analyzed or parsed (shown by parse coverage). Furthermore, if a sentence could be parsed by the grammar, one of those parses should be correct (shown by correct predicate-argument structure and correct predicate-argument structure and features). Thus the best system will not only have the highest lexical coverage, parse coverage, correct predicate-argument structure and correct predicate-argument structure and features scores; it will have the smallest gap between each of these scores with respect to each other.

Figure 7.1 shows that the RAND and TYP baselines have large gaps between the lexical coverage and parse coverage, which indicates that while lexical and morphotactic inference was successful, the syntactic phenomena were not correctly specified and therefore the resulting grammars did not have the appropriate syntactic rules to parse most of the sentences. On the other hand, the BROAD-COV baseline was designed to parse as many sentences as possible, regardless of whether these parses are correct. For this reason, BROAD-COV grammars have a much smaller difference between the lexical and parse coverage than the other baselines. In fact, for Arapaho and Wakhi, the difference between lexical and parse coverage is smaller than it is for BASIL. BASIL has the smallest difference for Hixkaryana and South Efate and it ties with BROAD-COV on Titan.

The RAND and TYP grammars parse very few test items at all and even fewer correctly, so for the remaining metrics, I focus on comparing the BROAD-COV baseline with BASIL.

BROAD-COV has a smaller difference in parse coverage and coverage with correct predicate-argument structure than BASIL for Hixkaryana and Wakhi, while BASIL has a smaller difference for South Efate and Titan. The two tie for Arapaho.[14] However, an important part of the correctness of the semantic representation is the richness of the representation in terms of containing the correct semantic features. To better visualize this difference, Figure 7.2 includes only the scores for correct predicate-argument structure and correct predicate-argument structure and features, and Figure 7.3 zooms in on the bottom of the scale. From these figures, we can see that BASIL has a smaller difference in the scores for correct predicate-argument structure and correct predicate-argument structure and features than BROAD-COV, as BROAD-COV has no correct features for most of the sentences. In fact, BROAD-COV only has the correct features when there should be no features in the semantic representation.

In this subsection I summarized the performance of the different systems in terms of the various metrics in relation to each other. In other words, rather than considering the scores for each metric in isolation, I compared them as portions of each other, where the best grammar would be the one that minimizes errors by parsing test items as correctly or completely as possible. For most of the metrics I compare, BASIL and BROAD-COV perform fairly comparably, however, BASIL outperforms BROAD-COV in terms of the number of test items it parses with the correct predicate-argument structure and features out of those it parses with the correct predicate-argument structure.

7.1.4 Summary

While the results show a great deal of variation across the test languages, BASIL and BROAD-COV outperform the TYP and RAND baselines for most of the metrics. BASIL and BROAD-COV perform fairly comparably for a number of the metrics, but BASIL excels in two areas. First, BASIL generally has fewer spurious parses than BROAD-COV, indicating that there is

[14] There is a negligible difference of .01%, in favor of the BROAD-COV baseline.

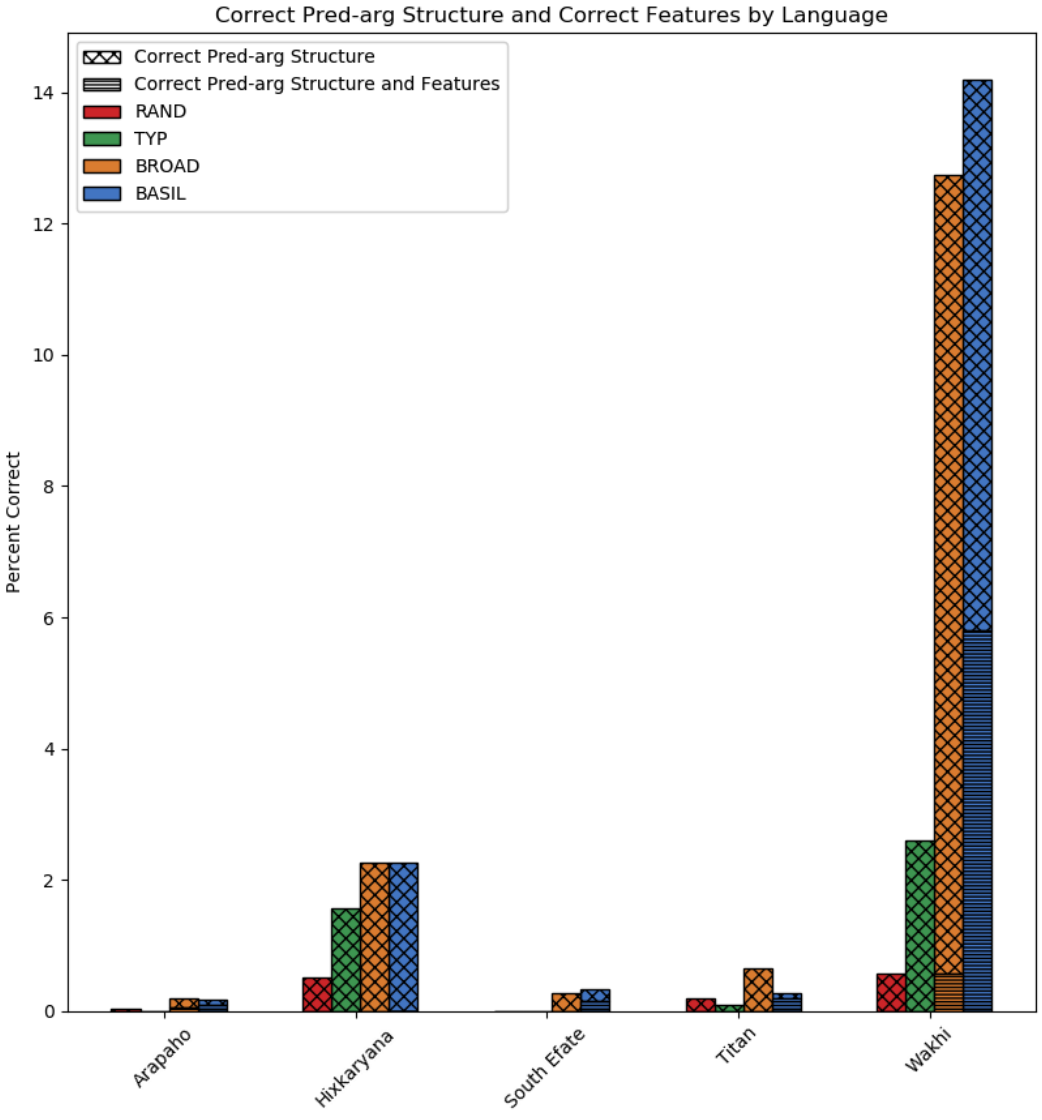


Figure 7.2: Correct Coverage for Held-out Languages

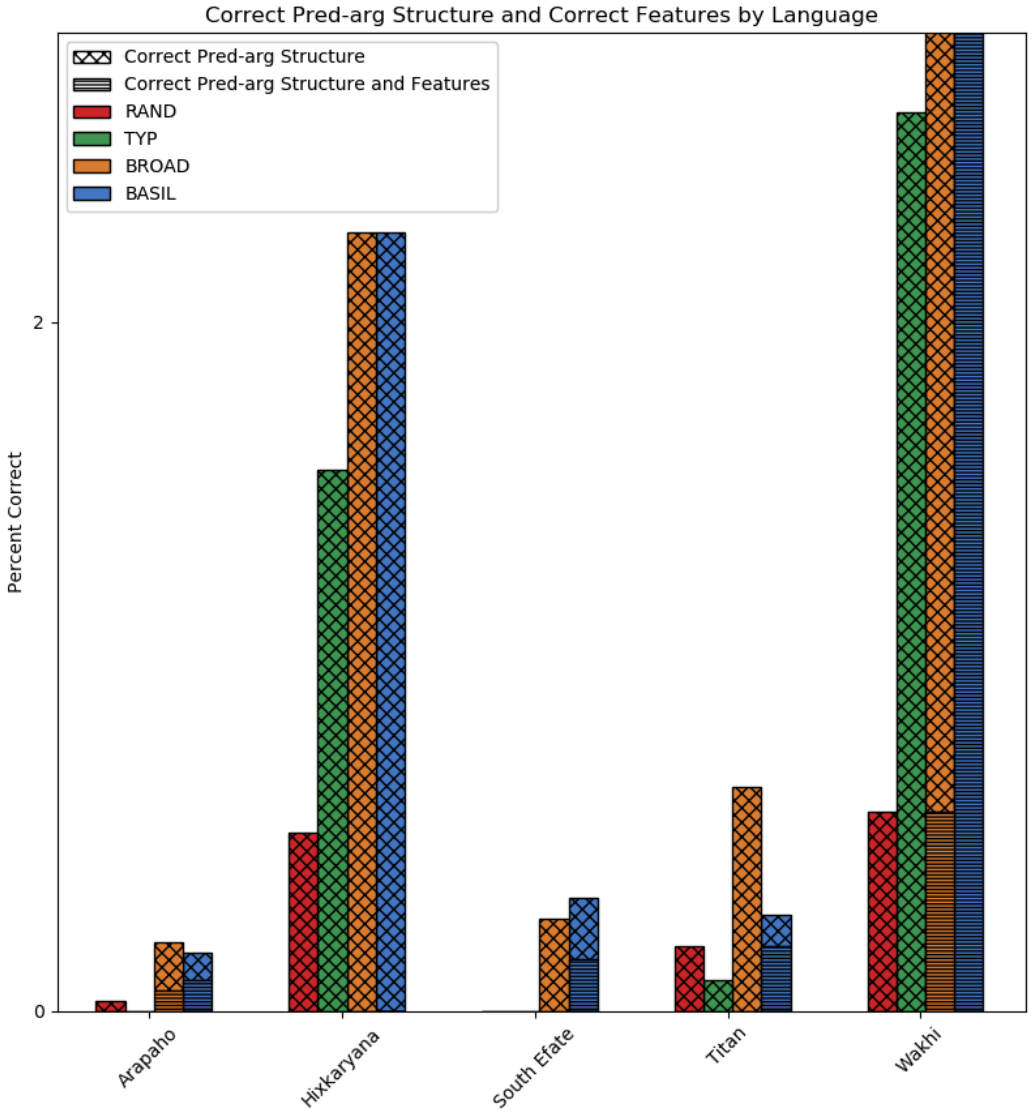


Figure 7.3: Zoomed in: Correct Coverage for Held-out Languages

less ambiguity in the inferred grammars than in that baseline. While TYP and RAND have even lower ambiguity scores, they also have such low coverage that this is not an advantage. Second, the semantic representations produced by BASIL are more correct in that they contain semantic features, resulting in higher scores for the correct predicate-argument structure and features metric.

Having analyzed the performance of BASIL's inferred grammars in terms of coverage and ambiguity, I will use the remainder of this chapter to describe the data and inferred grammars, identifying sources of error. I group errors into three categories: failures to parse or correctly parse out of scope phenomena, which are not targeted by BASIL's methodology as described in Chapter 4 (§7.2); failures to parse or correctly parse in scope phenomena, which are targeted by BASIL's inference system (§7.3); and sources of spurious or incorrect analyses in the grammar, which result in ambiguity (§7.4).

7.2 Error Analysis: Out of Scope Phenomena

I begin my error analysis by establishing what I do not expect BASIL's grammars to parse. Focusing on sentences where lexical coverage was achieved but the sentence did not parse or parsed incorrectly, I describe phenomena that are frequent in the test data but are beyond the scope of this dissertation.

7.2.1 Syntactic phenomena not targeted by BASIL

BASIL currently handles a number of lexical types such as transitive and intransitive verbs, auxiliaries, nouns, determiners and case-marking adpositions, as well as phenomena including word order, case, argument optionality, sentential negation and coordination. However, it does not yet handle a number of very common phenomena such as adjectives, adverbs, ditransitive or clausal complement-taking verbs, wh-questions, possession, etc. Therefore, sentences containing these lexical items will only have lexical coverage if a lexical item was inferred in error. At the same time, sentences that contain these syntactic phenomena will not parse or will not parse correctly.

Verb valence

Some of BASIL's errors can be characterized as errors in lexical inference, where a word was inferred as transitive or intransitive, when it is neither. For example, in South Efate, BASIL defined *pul* 'sling' as a transitive verb. This may be legitimate based on some uses of *pul* in the corpus; however, BASIL could not also define it as ditransitive because ditransitive verbs are not currently supported by BASIL or the Grammar Matrix. As a result, the parse for the sentence in (22) analyses *kusu* 'rat' as the direct object, of *pul* 'sling', which is incorrect, because *kusu* is the indirect object and the direct object is dropped.

(22) Ipulkin pak kusu.
i-pul-ki-n pak kusu
3S.RS1-sling-TR-3S.O to rat
'It slung it at the rat.' [erk] (adapted from Thieberger, 2006b)

Similarly, clausal complement-taking verbs such as 'say' were often inferred as transitive. In example (23) from Hixkaryana, *ka* 'say' is analyzed incorrectly as a transitive verb. As a result, the sentence parses incorrectly as two coordinated sentences, the second of which has a dropped object. This parse effectively means "did you kill the otter, and I said something", which is not the intended meaning according to the English translation.

(23) Wayawaya muwono àkano uro.
wayawaya m-wo-no i-ka-no uro
otter.SP 2A-shoot-IPST 1Sa-say-IPST 1
'Did you kill the otter, I said.' [hix] (adapted from Meira, 2020)

Possession

Another out of scope phenomenon that was missanalyzed in BASIL's grammars was possession. In many languages, possession is expressed morphologically, so there is no possessive

pronoun to analyze. As a result, sentences like (24) parse with the appropriate predicate-argument structure between the noun and verb, but there is no possessive relation in the semantics.

(24) Tosar yohoryaha.
t-esa-ri y-ehori-yaha
3R-place-POS REL-look.for-PRES:CTY
'He looks for his place.' [hix] (adapted from Meira, 2020)

Vocative

A number of sentences addressed the interlocutor, as shown in (25) from Arapaho. In most cases, a vocative such as this is missanalyzed as the subject or object.

(25) Howoto'oo neixoo.
howoto'oo neixoo
wake-up father
'Wake up father!' [arp] (adapted from Cowell, 2018)

Sentence linkers

Another phenomenon that BASIL does not handle is sentence linkers. Particularly in narratives and transcribed speech, sentence linkers such as coordinators are often found at the beginning of sentences, as illustrated in (26). I expected that these sentences would not parse at all, because there is only one coordinand. Incidentally, these sentences do parse, but the phenomenon has not been completely implemented in Grammar Matrix customization system. As a result, this sentence (and others like it) parses but the coordinand is not linked with the coordinator in the semantics, resulting in an ill-formed semantic representation.

(26) Noh hetceenok
noh het-ceenoku
and IMPER.FUT-sit-down
'and take your place' [arp] (adapted from Cowell, 2018)

7.2.2 Disfluencies

In addition to the syntactic phenomena that the inferred grammars do not cover, they also do not handle disfluencies well. In the transcribed speech in the Arapaho corpus, disfluencies take a number of forms, including pauses (27) and repeated words (28).

(27) NiiP niine'eehek nehe' honoh'oe.
nii-P niine'eehek nehe' honoh'oe'
IMPERF-pause/break here-is.3S this boy
'Here is this young man.' [arp] (adapted from Cowell, 2018)

(28) He'ih'iiniibei'i nuhu', nuhu' siisiiko'.
he'ih'ii-niibei nuhu' nuhu' siisiiko'
NARRPAST.IMPERF-sing this this duck
'This one duck was singing.' [arp] (adapted from Cowell, 2018)

In cases like (27), BASIL picks up the *P* character as a verb (it is tagged as a verb in the corpus and has verbal inflection) with the predication _pause/break_v_rel. This predication is spurious in the semantic representation, so I do not consider the semantic representation correct. Sentences with repetition can sometimes be analyzed with asyndedic coordination, but in (28) the repeated word is a determiner, which BASIL does not infer coordination for. Because BASIL does not have a way to handle this type of disfluency, the sentence does not parse.

7.2.3 Summary

In this section I described test sentences with lexical coverage that did not parse or parsed incorrectly because they include phenomena not handled by BASIL. Because the scope of BASIL's inference is limited to a specific set of linguistic phenomena and I tested it on corpora that contain out of scope phenomena, these account for the majority of the parse failures and incorrect parses.

7.3 Error Analysis: In Scope Phenomena

Whereas the previous section described common errors due to out of scope phenomena in the test data, this section focuses on errors due to BASIL failing to correctly infer phenomena that it was designed to handle. I group these errors into two broad categories, although they interact closely: lexical and morphological errors and syntactic errors. Using specific examples, I describe the source of the errors which range from the input data to problems with BASIL's inference algorithms or their implementation.

7.3.1 Lexical and morphological errors

A number of errors can be attributed to BASIL's lexical and morphological inference. In some cases, words were inferred as belonging to the wrong lexical class. In others the lexical category was inferred correctly, but the word has the wrong predication or incorrect or missing semantic features.

Wrong part-of-speech

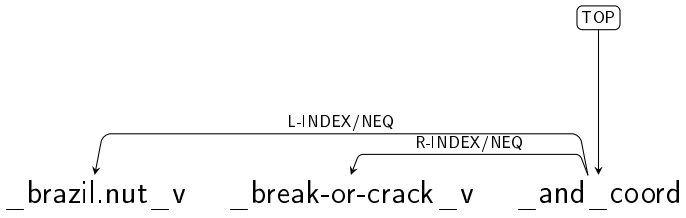
Both BASIL and MOM (the morphological and lexical inference system; Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017), rely on POS tags in the input to identify nouns and verbs. In some cases, the POS tag included in the original corpus may be incorrect. While this can be hard to determine without expertise in a given language, in some cases it is fairly clear. For example, in (29) the word *titko* is glossed as 'brazil.nut' but marked with a verbal

Vt POS tag. Annotation errors like this are not uncommon, as even the most careful human annotation is subject to error.[15]

(29) Tutko yakahetxkoni.
titko y-akaha-yatxkoni
Brazil.nut REL-break-DPST2:COL
Vt prs-Vt-tamn
'They were shelling Brazil nuts.' [hix] (adapted from Meira, 2020)

In this example, and throughout the corpus,[16] *titko* is glossed as a transitive verb (via the gloss Vt). As a result, the inferred grammar treats it semantically as an event instead of as a participant of the breaking/shelling event, resulting in an incorrect semantic representation. Furthermore, because the transitive verb *akaha* requires nominal arguments, the grammar cannot parse the verb *titko* as an argument of *akaha* and instead parses the sentences by coordinating the two verbs. This results in the incorrect MRS, shown in (30).

(30)



Working with unfamiliar corpora, I am also subject to making mistakes in interpreting some POS tags used in the corpora. In the Arapaho corpus, the tag 'vai' is generally used for verbs and is typically followed by something else, such as a feature (eg. vai.IMPERF). MOM requires lists of noun and verb POS tags used in the corpus in its input, and I

[15]Working with corpora before they have been thoroughly cleaned creates an opportunity to provide the linguist or curator with feedback regarding potential annotation errors in their corpus. At the same time, occasional mistakes in annotation have a relatively small impact on grammar performance overall and are not a reason to exclude a dataset.

[16]Because Toolbox (SIL International, 2015) has auto-glossing options, having *titko* tagged as Vt throughout the corpus does not mean that the linguist repeatedly glossed it that way. Instead, it is more likely that a mistake was made once and propagated by the auto-glossing functionality.

included all tags starting with vai as verbal tags for this language. In the case of vai.ITER, either my assumption was not correct or the term was glossed in error, as that tag is used exclusively for the word *he'iiteihi3i* 'someone'. Even if the linguist analyzes *he'iiteihi3i* as a verb syntactically, treating it as a verb in the inferred grammar again results in an incorrect semantic representation for sentences like the one in (31). Similar to the Hixkaryana example in (29) and (30), the parse for this sentence, which treats *he'iiteihi3i* as a verb, treats it as an event coordinated with the follow/track event, rather than as a participant of that event.

(31) he'iiteihi3i nih3ookuheinoo.
he'iiteihi3i nih-3ookuh-einoo
someone PAST-follow/track-3S/1S
vai.ITER prefix-vta-infl
'Someone is following me.' [arp] (adapted from Cowell, 2018)

Finally, whereas some corpora include word-level POS tags, others only include morpheme-level POS tags. When MOM sees morpheme-level POS tags in the input, it expects the part-of-speech tag on the root to be the POS for the word. However, in example (32) from Wakhi, the root *John* is not tagged. At the same time, a tense/aspect/mood clitic is attached to it and tagged with the verbal tag TAM. In the absence of a POS tag on the root, MOM uses the tag on the clitic and incorrectly identifies *John* as a verb.[17]

(32) Johnep pitet jawem.
John=əp puv-t=ət jaw-əm
___=FUT drink-PST=and eat-1SG.NPST
___=TAM v-vPst1=conn v-v:Tns/Agr
'John will drink and I will eat.' [wbl] (adapted from Kaufman et al., 2020)

[17]This error is made possible by MOM's handling clitics as affixes rather than independent words, when they are not separated from the word they attach to with white space. Better handling of clitics by MOM would also resolve this error.

Wrong predication

In many cases, even if the correct lexical class was inferred, the predication was incorrect. MOM defines the predication value for words by identifying the root's gloss and then removing any known grams to identify the lemma. The lemma is then used to build the predication, but if none is found, the grams are used in the predication. For example, a verb glossed as sing.PST would be given the predication _sing_v_rel. However, if the root is not correctly identified, the wrong gloss will be used to construct the predication.

One way that MOM identifies the root of a word is by assuming that hyphens are included on affixes and not on roots.[18] For the IGT in (33), consider the morpheme boundaries in the Xigt data representation in (34).

(33) Nehe' hinen nihneenit.
nehe' hinen nih-neeni-t
this man PAST-itis-3.S
'The man was the one.' [arp] (adapted from Cowell, 2018)

(34)

```
<item id="m1" segmentation="w1">nehe'</item>
<item id="m2" segmentation="w2">hinen</item>
<item id="m3" segmentation="w3">nih-</item>
<item id="m4" segmentation="w3">neeni</item>
<item id="m5" segmentation="w3[8:8]">-</item>
<item id="m6" segmentation="w3">t</item>
```

[18]MOM can be told not to use this heuristic with the configuration setting 'boundaries', in which case it will use a list of user-provided glosses to identify affixes.

For the third word (comprising morphemes m3-m6), MOM assumes that *nih-* is a prefix because it includes a hyphen. However, the hyphen between *neeni* and *t* is not attached to either,[19] so MOM uses a heuristic to guess the root. In this case, MOM guessed that the last morpheme was the root, so the predication was constructed from its gloss. This resulted in the word having the predication _3.S_v_rel, instead of the more meaningful _itis_v_rel.[20]

Missed semantic features

The area where BASIL has the greatest advantage over the baseline systems is its addition of semantic features to the grammars. Even so, it made some errors in feature inference. There is significant variation in the way linguists gloss syntactico-semantic features in their corpora, and BASIL's most straight-forward source of error for semantic features was in not properly identifying all grams in the held-out corpora. BASIL uses a large dictionary of glosses, which it maps to 99 common PNG, TAM and case grams to identify features.[21] Even so, the held-out corpora included grams that were not in this dictionary. In particular, this dictionary does not include any glosses for the pluperfect aspect 'PLPF', which is used in Wakhi, the immediate past 'IPST' or distant past 'DPST' used in Hixkaryana, or the narrative past 'NARRPAST' used in Arapaho. In addition, while the dictionary includes 'D' as a possible gloss for dual number and includes quite a few person and number combinations (e.g. '3DU'), it did not contain '3D' which is used for third person, dual number in the South Efate corpus. This alone led to a number of sentences, which otherwise parsed correctly, not including all of the appropriate semantic features.

For the Wakhi data, a number of TAM features were missed by the inference system because a TAM clitic leaned on a non-verbal word, as in (35). Here the progressive aspect

[19]In the original toolbox data, the hyphen was separated from both morphemes with white space: *nehe' hinen nih- neeni - t*. The Xigt importer (Goodman et al., 2015) identifies morphemes based on white-space, so the hyphen is kept separate.

[20]This predication is not ideal either, but at least it contains meaning that corresponds with the word *be*, as it is represented in the gloss.

[21]A description of how these grams were collected can be found in Section 4.1.

clitic leans on the pronoun and the future tense clitic leans on the conjunction. Because BASIL does not collect TAM features from words that are not verbs or auxiliaries, these features do not end up in the semantic representation for this and similar sentences.

(35)	Wuzʂ	jawem	wozep	jawem.
	wuz=ʂ	jaw-əm	woz=əp	jaw-əm
	1SG.NOM=PROG	eat-1SG.NPST	and=FUT	eat-1SG.NPST
	pronom=TAM	v-v:Tns/Agr	conn=TAM	v-v:Tns/Agr

'I am eating and I will eat.' [wbl] (adapted from Kaufman et al., 2020)

In addition to the problem picking up TAM features from clitics that lean on non-verbal categories, this example also illustrates another area of future work: The second `_eat_v_rel` predication should actually have include the future tense, which is a subtype of the non-past tense that inflects on the verb. However, because BASIL defines tenses individually and not as part or a hierarchy, if the future tense was picked up by the grammar, it would fail to unify with the non-past. For this reason, work should be done to infer TAM hierarchies so that the non-past and future features can both be modeled a separate features, but will also unify to the more specific of the two (in this case future).

Summary

Errors in lexical inference were largely due to limitations in information BASIL can access in the input data, such as incorrect or missing annotations or spurious white-space between affixes and hyphens. At the same time, BASIL's collection of expected or known glosses, while gathered from a variety of different sources, still doesn't anticipate every possible gloss used by linguists. Adding to the dictionary as new glosses are discovered is trivial, but better handling of missing annotations in the input data would improve the performance of the inferred grammars.

7.3.2 Incorrect syntactic specifications

Sources of error due to syntactic specifications are a bit more difficult to identify than lexical errors. Identifying failed parses based on incorrect lexical specifications is the first step when examining parse trees, and only when the lexical items are correct, is it possible to see that the words failed to combine into the expected structures for syntactic reasons. From sentences that had correct lexical specifications, I found three primary sources of error from syntactic specifications: auxiliary order, coordination and case.

Auxiliaries

As discussed in Section 4.5.1, BASIL treats words that have only TAM and/or PNG agreement features as auxiliaries. The abundance of such TAM auxiliaries in the held-out languages, such as the future tense auxiliary in (36), revealed an unfortunate bug in my implementation of auxiliary inference. The clause in the inference code that checks whether the auxiliary occurs before or after its complement assigns the opposite value that it should. For this reason some inferred grammars required auxiliaries to occur after their verbal complements, where in fact they should occur before. While there were auxiliaries in the development languages, my focus on a language with V2 word order (Wambaya) allowed me to overlook this bug, as the Grammar Matrix's analysis for V2 languages does not worry about the order of the auxiliary with respect to its complement, as long as the auxiliary is in the second position.

(36) Tumrə maz̧ jittu.
tumrə maz̧ jaw-tu
FUT 1SG.OBL eat-PLPF
'I will have had eaten.' [wbl] (adapted from Kaufman et al., 2020)

This bug also extended to agreement clitics, which BASIL analyzes as auxiliaries. Titan

has an agreement particle, illustrated in (37) as *i*,[22] which occurs before the verb. Because BASIL incorrectly posited that auxiliaries occur after the verb they attach to instead of before, many sentences in the Titan corpus which include this agreement particle do not parse.

(37) Cako i tokai nat i tokai.
cako i tokai nat i tokai
hermit_crab 3SG go boy 3SG go
'The hermit crab and the boy left.' [ttv] (adapted from Bowern, 2019)

Coordination

Coordination inference, described in Section 4.5.5, errs on the side of positing VP coordination unless it finds explicit evidence of S coordination in the form of a projected subject dependency that intervenes between the coordinator and a verb in the coordinand. This algorithm may be too aggressive because dependency tag projection is not always successful. In addition to that, the algorithm does not consider cases where the subject is dropped or cases where there is no coordinator, because an asyndedic strategy is employed, i.e. clear cases of S coordination that nonetheless lack the mark the system was looking for: a subject between a coordinator and a verbal head of a coordinand. Because the inference of S coordination relies on an overt coordinator, sentences like the one in (38) from Titan, which shows asyndedic coordination are taken by BASIL as evidence of VP coordination instead of S even though each coordinand has an overt subject. Because examples like this do not result in BASIL adding asyndedic S coordination to the grammar specifications, they cannot be parsed by the inferred grammar.

[22]This agreement particle is homophonous with the third singular pronoun, but Bowern (2011) analyzes it as separate word category than that pronoun.

(38) I ani pou i ani ma.
i ani pou i ani ma
3SG eat pig 3SG eat taro
'He ate the pig and he ate the taro.' [ttv] (adapted from Bowern, 2019)

Furthermore, the examples of monosyndedic S coordination in Wakhi were missclassified as VP coordination because failure to align the subjects between the English translation and the sentence. This prevented BASIL from inferring S coordination strategies and adding them to the grammar specifications. Because the BROAD-COV baseline posits asyndedic S coordination for all languages, that baseline was able to correctly parse the Titan sentence in (38) as well as other sentences with asyndedic S coordination in Titan and Wakhi, giving it a boost in coverage over BASIL.

Case frame

Finally, BASIL infers case frame based on the overt case markings on the subject and object (according to projected dependencies), in order to account for quirky case, as described in Section 4.5.2. However, if no overt argument is found, the verb's case frame remains under-specified until it is merged with another verb that does has overt arguments with case marking. Even though BASIL inferred the over-arching nominative-accusative pattern for Wakhi, it found verbs in the training data with oblique subjects which were merged with verbs that did not have overt case marking on their subjects. Because of this, the inferred grammars for some of the Wakhi folds included a rather large transitive verb class with oblique case on the subject. This resulted in a number of IGT with overtly marked nominative subjects in the test data that did not parse.

Summary

The majority of errors discussed in this section come from lexical inference, but where lexical inference was successful, I identified three main sources of error in the syntactic specifications.

The first is a bug that resulted in auxiliaries having the wrong order with respect to their complements. Resolving this bug is trivial, while the errors in S coordination and case-frame inference require some re-designing of the algorithms. In the case of S coordination, the algorithm requires too much evidence to infer S coordination. As future work, I propose modifying the algorithm to rely less on projected dependencies and instead to leverage the dependency parse of the English translation to distinguish between VP and S coordination in the translation. The same redesign could be applied to N and NP coordination as well. The case frame inference algorithm may assign quirky case too readily and rather than merging lexical items with no case frame with those that have quirky case, should assign default case to those verbs unless a verb with the same orthography is found with quirky case in the corpus. Alternatively, better verb classes could be inferred with some re-tooling of the interaction between BASIL and MOM so that case frame inference happens after morphotactic inference, similar to the pronoun and auxiliary inference methodologies in Section 4.2.2.

7.4 Error Analysis: Ambiguity

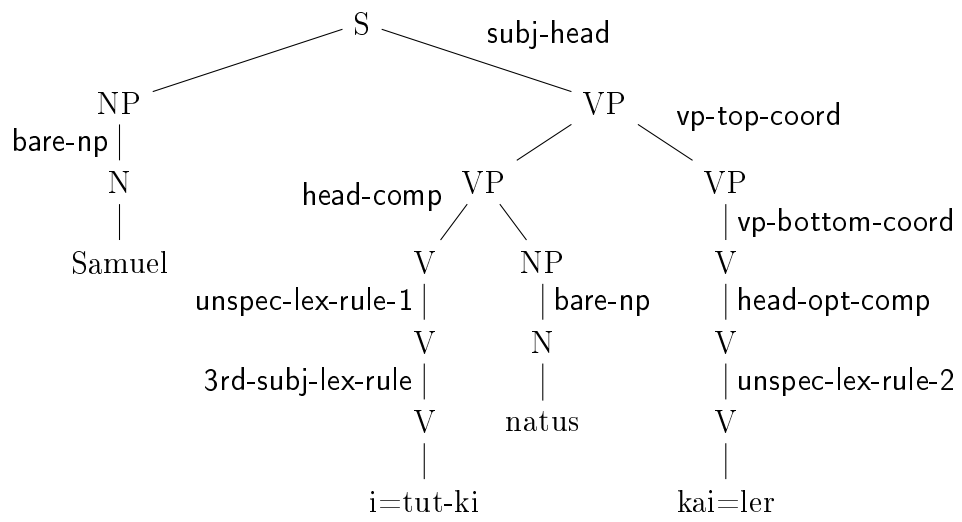
BASIL's inferred grammars generally had less ambiguity than the BROAD-COV baseline for two intuitive reasons. First, the free word order, argument optionality and coordination specifications in BROAD-COV introduce a lot of ambiguity in the number of ways nouns and verbs can combine. Second, BASIL's specifications for case frame and agreement further constrain which arguments can be subjects and objects, even in freer word order languages. In spite of this, BASIL's grammars for South Efate have significantly more ambiguity than BROAD-COV's.

First of all, BASIL infers free word order, subject and object dropping and asyndedic coordination for VPs and NPs for this fold. Because of this, BASIL's inferred grammar is not less ambiguous than BROAD-COV in those areas. In order to understand why BASIL's grammar is even more ambiguous than BROAD-COV's, I use as an example the sentence in (39) which has asyndedic coordination, lexical ambiguity, morphological ambiguity and no overt case marking.

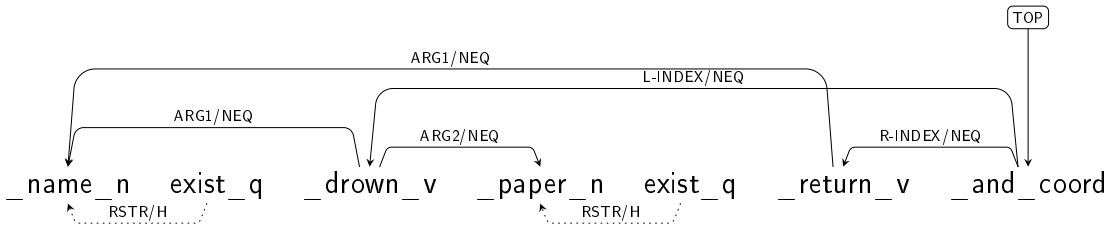
For this sentence, BASIL's grammar produces 2448 trees, while BROAD-COV's produces 19.[23] The best reading, produced by both grammars, is shown in the parse tree in (40) and semantic representation in (41).

(39) Samuel itutki natus kailer.
Samuel i=tut-ki natus kai=ler
Samuel 3S.RS1-drown-TR paper ES1-return
'Samuel threw in the paper and went back.' [erk] (Thieberger, 2006b)

(40)



(41)



[23]These numbers are based on the number of trees estimated for the sentence by the Full Forrest Treebanking software (FFTB; Packard, 2015). The number of parses produced by the ACE (Crysmann and Packard, 2012) parsing software which I used to generate the report in Table 7.5 is slightly smaller for BASIL's grammar at 1908 parses. This inconsistency is because FFTB does not verify that all trees are valid during unpacking (Packard, pc), but rather calculates an estimate from the packed forest.

One way to find the correct tree in a forest of 2448 and to understand which rules contributed to the ambiguity is to use discriminant-based tree selection as proposed by Carter (1997). I do this using the Full Forrest Treebanking software (FFTB; Packard, 2015), as described in Section 6.1. Figure 7.4 shows the choices among discriminants that single out the tree in (40) from the other 2447 trees in the parse forest.

The discriminants in Figure 7.4 are not ordered, and represent one of many paths in the decision space. The bottom 4 choices in the decision tree result in no difference in the semantic representation, yet combined they increase the ambiguity by a factor of 16. The no-drop-lex-rule is added by the argument optionality library of the Grammar Matrix (Saleem, 2010; Saleem and Bender, 2010). This rule is intended to be further constrained by agreement restrictions for dropped arguments, but because BASIL does not add this information to the grammar, these optional, non-inflecting lexical rules have no impact on the parse except in adding ambiguity for both verbs in the sentence. The at-lex-rule is added by the case library (Drellishak, 2009) for languages with case-marking adpositions. This rule optionally applies to both nouns in the sentence. Because they apply optionally, each of these lexical rules and each of the words they can apply to doubles the number of trees in the forest.[24]

In addition to these sources of ambiguity, there is an under-constrained noun coordination rule that applies optionally to each noun and can apply either before or after the bare-np rule, tripling the number of parse trees for each noun it can apply to. Because neither noun has an adjacent noun to attach to, these parses should not succeed, but they do as the result of a bug in the Grammar Matrix customization system.

All together the spurious case, coordination and argument optionality rules increase the number of possible trees by a factor of 144. Setting those aside, the number of possible trees looks much more reasonable. Additional ambiguity is added by two homophonous lexical rules for the *kai-* prefix: one adds first person agreement to the subject and the other (which produces the correct tree) does not add any features.[25]

---

[24]The optionality of a non-inflecting lexical rule suggests a bug in the Grammar Matrix.

[25]The morpheme is glossed by the linguist as ES1. Thieberger (2006a) defines the ES abbreviation as "echo

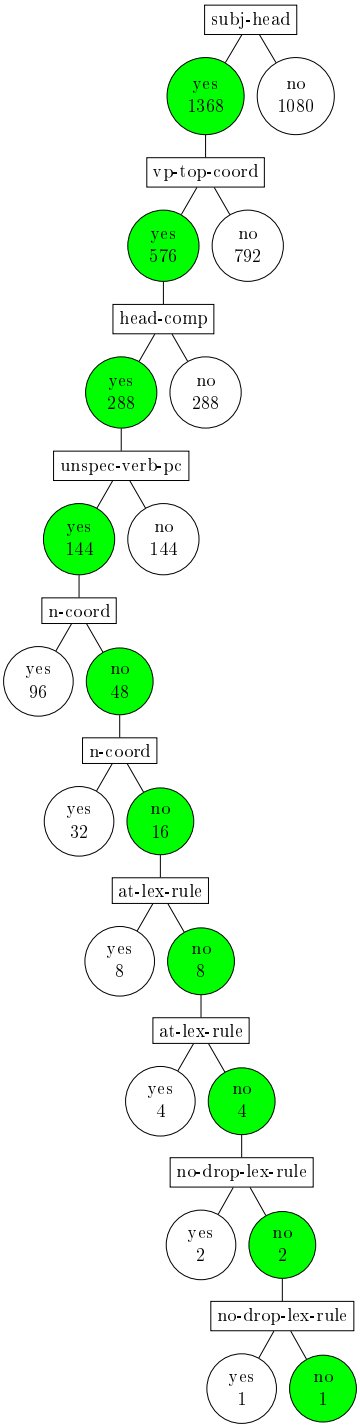


Figure 7.4: A decision tree illustrating the syntactic and lexical rules that discriminate between different parse trees produced by BASIL's grammar for the sentence in (39). The path in green shows the rules that I selected or excluded in order to identify the correct parse tree, which is shown in (40)

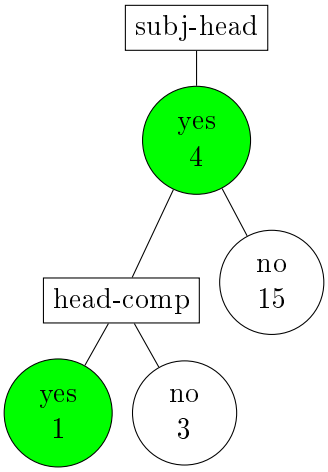


Figure 7.5: A decision tree illustrating the syntactic and lexical rules that discriminate between different parse trees produced by the BROAD-COV grammar for the sentence in (39). The path in green shows the rules that I selected or excluded in order to identify the correct parse tree, which is shown in (40)

The three choices at the top of the decision tree discriminate between trees in which *natus* is the object of *i=tut-ki* or *kai=ler* and indirectly, prevent *kai=ler* from being analyzed as a noun[26] and coordinated with *natus*.

In contrast with the forest produced by BASIL, the decision tree for BROAD-COV to produce the parse shown in (40) is shown in Figure 7.5. The lexical rules in the last four nodes in the decision tree in Figure 7.4 are not in the BROAD-COV grammar and therefore do not apply. Because ambiguity is a matter of combinatorics, the spurious lexical rules in BASIL's grammar inflate the ambiguity significantly. The same could be said for the sources of ambiguity in the BROAD-COV grammars for the other languages, where BASIL had less ambiguity.

Many of the sources of ambiguity in the South Efate grammars trace back to bugs in

---

subject", and I assume that the 1 is particular echo subject marker, but does not indicate first person, as there is no first person noun in the translation.

[26]The grammar also contains a noun lexical entry for *ler*, due to an incorrectly projected POS tag in the training data where a POS tag was not included in the original corpus.

the Grammar Matrix customization system, rather than BASIL's inference. Furthermore, the high ambiguity for South Efate grammars was an outlier among the ambiguity in BASIL's grammars for the evaluation languages. This suggests that these sources of ambiguity, both from Matrix bugs and otherwise, are not particularly pervasive.

7.5 Summary of Results and Analysis

In this chapter, I evaluated the quality of my grammar inference system through a number of different lenses. First, I evaluated the system on held out languages and datasets which tested how well the system generalizes beyond the development languages and data. The inference system (as well as the baseline) performed much better on some languages and datasets than others and error analysis revealed a number of out of scope phenomena and unexpected inputs that were the cause of this variation in performance.

BASIL outperforms the baselines by having better coverage and lower ambiguity overall. BASIL has higher coverage than the TYP and RAND baselines, and its coverage is on par with BROAD-COV. At the same time it has lower ambiguity than BROAD-COV, except in the case of South Efate, so it succeeds in the goal of increasing coverage while limiting the ambiguity trade-off. Finally BASIL's semantic representations are richer than the baselines, as they include semantic features.

Chapter 8
CONCLUSION

In this dissertation, I introduced BASIL — Building Analyses from Syntactic Inference in Low-resource languages — a system for the automatic inference and generation of machine-readable grammars from IGT data. Leveraging the rich annotation in interlinear glossed text together with syntactic information projected from parses of the English translation onto sentences in a low-resource language (Xia and Lewis, 2007; Georgi, 2016), BASIL infers syntactic information about the language in order to produce an HPSG (Pollard and Sag, 1994) grammar. In this chapter, I conclude with a summary of the contributions of this work (§8.1, §8.2), considerations for improving the system (§8.3), and BASIL's potential uses in grammar engineering and language documentation and revitalization (§8.4).

8.1 System Summary

BASIL utilizes an end-to-end pipeline that begins with an IGT corpus of a language and produces an HPSG grammar which can be loaded into parsing software to produce syntactic and semantic representations for strings in that language. Drawing on the linguistic information encoded in IGT text and generalizations about language from the typological literature, I designed algorithms that infer lexical and syntactic properties about a language and define these properties in a grammar specification. This grammar specification, can be input into a grammar customization toolkit (the Grammar Matrix; Bender et al., 2002, 2010) to produce a machine-readable HPSG grammar for that language.

I built on existing work in grammar inference that produced both morphological (Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017) and syntactic (Bender et al., 2013, 2014; Howell et al., 2017; Zamaraeva et al., 2019a), specifications for a language. This work focused

on lexical and morphotactic specifications for nouns and verbs, word order, case system and case frame for verbs. I integrated the existing modules into a single system which I scaled up by adding inference for determiners, auxiliaries, case-marking adpositions, PNG and TAM features, argument optionality, negation and coordination.

The result is an inference system that identifies the over-arching typological patterns for each of these phenomena and encodes that information in a grammar specification, which is then used to produce a grammar. As one of the goals of this work is to automatically infer grammars for a broad range of low-resource and endangered languages, I developed inference algorithms using a data-driven process, testing my system on a genealogically and geographically diverse set of languages. During development, I consulted 27 languages from 19 language families, spread over six continents. I did end-to-end system testing on 9 of those 27 development languages.

8.2 Summary of Results

In order to test the cross-linguistic generalizability of my inference system, I evaluated it using 5 languages from 4 language families that were not considered during development and did not come from any of the language families that I used in previous end-to-end testing. These languages were: Arapaho, Hixkaryana, South Efate, Titan and Wakhi. Doing ten-fold cross validation for each language, I compared the performance of BASIL's inferred grammars with three baselines. The TYP baseline used the cross-linguistically most common specifications for each phenomenon (based on typological surveys), while RAND used random specifications. The low coverage of these baselines demonstrated that in order to produce a useful grammar, it is not sufficient to guess the right specifications for just some phenomena, but the specifications for a variety of interacting phenomena must be correct. The third baseline, BROAD-COV, was designed to parse as many sentences as possible in a language, and in spite of this, BASIL's overall coverage was comparable to BROAD-COV, while its grammars had less ambiguity for four of the five languages.

In addition to BASIL's parse coverage being higher than the TYP and RAND baselines and

comparable with BROAD-COV, the semantic representations produced by BASIL's grammars were richer. I used evaluation metrics in which I assessed not only the number of sentences that parsed, but the correctness of those parses in terms of the meaningfulness of their predications and the correctness of the argument relations for those predications. In this respect, BASIL and BROAD-COV performed comparably, outperforming the other two baselines by a large margin. However, BASIL's grammars also added semantic features for person, number, gender, tense, aspect and mood on the semantic predicates, resulting in even more detailed representations than those produced by the BROAD-COV grammars.

The results of this evaluation demonstrated that my inference algorithms generalize beyond the development languages to new languages and language families. At the same time, evaluating on previously unconsidered languages as well as datasets from different linguists who use different annotation conventions, revealed areas in which the system could be improved, as I will describe in the next section.

8.3 Limitations and Future Work

8.3.1 Extending grammar inference

I evaluated the inference system on IGT corpora that contained text from transcribed speech, narratives and elicitation, all of which include phenomena that are outside of BASIL's current scope. The advantage of this approach is that it measured the utility of the inferred grammars over a variety of naturally occuring speech genres and highlighted the range of linguistic phenomena that must be captured in order to model human language. In particular, a number of sentences did not parse because they contained phenomena not modeled by the inferred grammars such as adjectives, adverbs, posessives, subordinate clauses and constituent questions.

A key dependency of my inference pipeline is the Grammar Matrix customization system (Bender et al., 2002, 2010), which takes BASIL's inferred grammar specifications as input and generates a grammar. Because BASIL relies on the Matrix's typologically robust syn-

tactic analyses to produce the grammars, BASIL can in principle be extended to account for phenomena as they are added to the Grammar Matrix. Recent work has added libraries for clausal complements (Zamaraeva et al., 2019b), adverbial clausal modifiers (Howell and Zamaraeva, 2018), nominalized clauses (Howell et al., 2018), adnominal posession (Nielsen, 2018; Nielsen and Bender, 2018) and constituent questions (Zamaraeva, in prep). There is also a library for adjectives (Trimble, 2014), although adverbs are not currently handled.[1] Leveraging the analyses for these phenomena in the Grammar Matrix, modules can be added to BASIL to account for these phenomena as well.

In addition to highlighting phenomena that BASIL does not handle but are nonetheless common, evaluation on held out languages and datasets helped me to identify short-comings in the current inference modules. In particular, the algorithm for inferring coordination relies too heavily on projected dependencies and is prone to missidentifying sentence coordination as verb phrase coordination. Similarly, case-frame inference should be made more robust to missing alignments by positing the case governed by the language's over-arching case system when overt case-marking is not found, rather than allowing verbs with under-specified case frames to be merged into the same lexical classes as verbs with quirky case. Finally, the current handling of verbs that can be interpreted as both transitive and intransitive defines lexical classes that always treat these verbs as transitive. While this usually produces a correct parse, in which the verb is interpreted as having a dropped argument, it does not also allow a parse to be produced which represents the verb as not having an object that is recoverable from discourse. Future work should consider inferring transitivity more carefully and in some cases positing multiple lexical entries for such verbs.

[1]Some types of adverbs are modeled by the libraries for negation (Crowgey, 2012), subordination (Howell and Zamaraeva, 2018) and constituent questions (Zamaraeva, in prep), but there is no comprehensive analysis for adverbs.

8.3.2 Evaluating dataset characteristics

For development and evaluation, I used varied corpora in terms of both size and content. My smallest dataset, Matsigenka, had 350 IGT, while my largest, Arapaho and Chintang, had 5,000 and 10,000.[2] Some datasets, such as Chintang, were thoroughly glossed, while others, like Haiki, were only partially glossed. Still others, Wambaya and Nuuchahnulth, had thorough interlinear glossing but no part-of-speech tags. Furthermore, the datasets included different proportions of elicited, conversational and narrative data, all of which vary in the degree to which they contain disfluencies, sentence linkers and other linguistic phenomena not handled by BASIL.

This variation in the characteristics of the data make it impossible to say with any certainty that a specific language is more difficult to infer a grammar for than others. At the same time, the variation in languages makes it difficult to reason about how much data with what degree of annotation is necessary for the inference system to work well. The answers to these questions, however, would be interesting. BASIL's highest coverage was on Abui and Wakhi, which were not the largest datasets, but that doesn't necessarily mean that the inference system is biased towards the Trans-New Guinea or Indo-European language families these languages belong to. The lowest coverage was on Arapaho, which was by far the largest dataset, so further investigation into what factors hurt coverage for this language beyond those discussed in Chapter 7 would be informative.

Accounting for the characteristics of languages or datasets that have the most impact on system performance would enable better assessment of the system's weaknesses and ways to improve it. For this reason, I propose future work that systematically tests these factors by testing with different subsets of a single dataset with different sizes, genres, completeness of glossing or presence of part of speech tags. Upon identifying a threshold for these factors

[2]In fact the source datasets for each of these languages are much larger. I down-sampled the Matsigenka corpus to include only IGT with English translations, reduced the Arapaho corpus to 5,000 IGT in order to run the inference and evaluation pipeline in a reasonable amount of time, and worked with a only a sample of the full Chintang corpus.

above which system performance stabilizes, it would then be possible to do more rigorous cross-linguistic testing to find language families or typological properties that BASIL struggles with.

8.4 Applications and Downstream Uses

Fully acknowledging that BASIL's grammars are currently limited to a certain number of phenomena and are subject to some degree of error, in this section I discuss possible uses for these grammars both now and after additional inference modules are added.

8.4.1 Grammar engineering

As noted by Bender et al. (2002, 2010), the process of developing a machine-readable, syntactically grounded grammar by hand is very time consuming. The Grammar Matrix automates this process to some degree by storing customizable analyses and adding them to a grammar based on grammar specifications. Creating these specifications (either by hand or using the Grammar Matrix web questionnaire), is still somewhat time-consuming though, particularly for defining lexical and morphological classes as well as specifying multiple strategies for coordination and subordination. For this reason, if a linguist has a corpus of IGT data available, BASIL can do some of the heavy lifting by automatically producing a grammar specification which a linguist can add to or modify.

8.4.2 Language documentation and revitalization

Machine readable grammars that are somewhat larger than those produced by BASIL have been used for a broad range of applications such as data exploration (Letcher and Baldwin, 2013; Bouma et al., 2015), grammar checkers (da Costa et al., 2016) and automatic tutors (Hellan et al., 2013). Accelerating the process of developing this type of grammar, increases the number of grammars that can be used for these applications.

At the current stage, inferred grammars could still be useful for data exploration as they can be used to search corpora for the phenomena they model. This type of data exploration

could assist linguists in finding relevant examples of specific phenomena they wish to analyze, or it could be used to help teachers find varied examples to use in lessons. Once a sufficient number of phenomena are handled by grammar inference, machine-readable grammars inferred from descriptive grammars could accompany those descriptive resources as a tool for further investigating the language's syntax, as described by Bender et al. (2012) and Bouma et al. (2015). My inferred grammars for Wambaya, which were based on IGT extracted from Nordlinger 1998, serve as proof of concept for this possibility. Finally, as inferred grammars help to streamline the process of grammar engineering, ultimately grammars that started with BASIL and were extended by hand could be used to produce grammar checkers along the lines of da Costa et al. 2016 and other educational tools in order to assist in the effort of language revitalization.

8.4.3 Linguistic typology

Finally, there is potential for a symbiotic relationship between BASIL and typological resources, such as WALS (Dryer and Haspelmath, 2013), SAILS (Muysken et al., 2016) and others. In particular, previous work has found that a number of the Grammar Matrix's specifications map directly to WALS features (de Almeida et al., 2019). For languages where these features are encoded in WALS, this information can be incorporated into the grammar inference pipeline to improve the accuracy of inference for some phenomena (Zhang, in prep). On the other hand, for languages whose features have not been added to databases like WALS, BASIL could be used to automatically infer those features, if an IGT corpus (or a descriptive grammar from which IGT can be extracted) is available.

8.5 Conclusion

The primary contribution of this work is a grammar inference system that takes an IGT corpus as input and produces a machine-readable, HPSG grammar that can be used for parsing and generation. Although previous work has automatically generated grammars for English and other high-resource languages, BASIL focuses on producing language technology

in the form of syntactically precise grammars for low-resource and endangered languages. In light of this, I tested the system on a large number of genealogically and geographically diverse languages and verified its cross-linguistic generalizability. Although the grammars produced by BASIL are still relatively low-coverage over corpora containing the complexity and variety inherent to human language, they provide a valuable starting point for producing broader coverage grammars which can be used to assist data exploration and language documentation and revitalization.

BIBLIOGRAPHY

Peter Ackema, Patrick Brandt, Maaike Schoorlemmer, and Fred Weerman, editors. 2006. *Arguments and Agreement*. Oxford University Press, Oxford.

Andrea Acri. 2018. Draft of an in-progress critical edition of chapter 3 of the Bhuvanakośa prepared for the 4th international intensive course in old javanese, yogyakarta. 15–29 July 2018 (used with the author's permission).

Željko Agić, Anders Johannsen, Barbara Plank, Héctor Martínez Alonso, Natalie Schluter, and Anders Søgaard. 2016. Multilingual projection for parsing truly low-resource languages. *Transactions of the Association for Computational Linguistics*, 4:301–312.

Gregory DS Anderson. 2006. *Auxiliary Verb Constructions*. OUP Oxford.

Emily M. Bender. 2008a. Grammar engineering for linguistic hypothesis testing. In *Proceedings of the Texas Linguistics Society X conference: Computational linguistics for less-studied languages*, pages 16–36.

Emily M. Bender. 2008b. Radical non-configurationality without shuffle operators: An analysis of Wambaya. In *Proceedings of the 15th International Conference on Head-Driven Phrase Structure Grammar, NICT, Keihanna, Japan*, pages 6–24.

Emily M. Bender. June 2008c. Evaluating a crosslinguistic grammar resource: A case study of Wambaya. In *Proceedings of ACL-08: HLT*, pages 977–985, Columbus, Ohio, June 2008c. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/P08-1111`.

Emily M. Bender. 2010. Reweaving a grammar for Wambaya. *Linguistic Issues in Language Technology*, 3(1).

Emily M. Bender and Jeff Good. 2005. Implementation for discovery: A bipartite lexicon to support morphological and syntactic analysis. In *Proceedings from the Panels of the Forty-First Meeting of the Chicago Linguistic Society: Volume 41-2.*

Emily M. Bender, Dan Flickinger, and Stephan Oepen. 2002. The Grammar Matrix: An open-source starter-kit for the rapid development of cross-linguistically consistent broad-coverage precision grammars. In John Carroll, Nelleke Oostdijk, and Richard Sutcliffe, editors, *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics*, pages 8–14, Taipei, Taiwan, 2002.

Emily M. Bender, Scott Drellishak, Antske Fokkens, Laurie Poulson, and Safiyyah Saleem. 2010. Grammar customization. *Research on Language & Computation*, 8:1–50. ISSN 1570-7075.

Emily M. Bender, Sumukh Ghodke, Timothy Baldwin, and Rebecca Dridan. 2012. From database to treebank: Enhancing hypertext grammars with grammar engineering and treebank search. In Sebastian Nordhoff and Karl-Ludwig G. Poggeman, editors, *Electronic Grammaticography*, pages 179–206. University of Hawaii Press, Honolulu.

Emily M. Bender, Michael Wayne Goodman, Joshua Crowgey, and Fei Xia. August 2013. Towards creating precision grammars from interlinear glossed text: Inferring large-scale typological properties. In *Proceedings of the 7th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 74–83, Sofia, Bulgaria, August 2013. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/W13-2710`.

Emily M. Bender, Joshua Crowgey, Michael Wayne Goodman, and Fei Xia. June 2014. Learning grammar specifications from IGT: A case study of Chintang. In *Proceedings of the 2014 Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 43–53, Baltimore, Maryland, USA, June 2014. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/W14-2206`.

Emily M. Bender, Dan Flickinger, Stephan Oepen, Woodley Packard, and Ann Copestake. April 2015. Layers of interpretation: On grammar and compositionality. In *Proceedings of the 11th International Conference on Computational Semantics*, pages 239–249, London, UK, April 2015. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W15-0128.

Balthasar Bickel, Bernard Comrie, and Martin Haspelmath. The Leipzig glossing rules: Conventions for interlinear morpheme-by-morpheme glosses. Max Planck Institute for Evolutionary Anthropology and Department of Linguistics, University of Leipzig, 2008. URL http://www.eva.mpg.de/lingua/resources/glossing-rules.php.

Balthasar Bickel, Martin Gaenszle, Novel Kishore Rai, Vishnu Singh Rai, Elena Lieven, Sabine Stoll, G. Banjade, T. N. Bhatta, N Paudyal, J Pettigrew, and M Rai, I. P.and Rai. 2013a. Durga. URL https://corpus1.mpi.nl/qfs1/media-archive/dobes_data/ChintangPuma/Chintang/Narratives/Annotations/durga_exp.tbt. Accessed: 2013.

Balthasar Bickel, Martin Gaenszle, Novel Kishore Rai, Vishnu Singh Rai, Elena Lieven, Sabine Stoll, G. Banjade, T. N. Bhatta, N Paudyal, J Pettigrew, and M Rai, I. P.and Rai. 2013b. Mechacha talk. URL https://corpus1.mpi.nl/qfs1/media-archive/dobes_data/ChintangPuma/Chintang/Narratives/Annotations/mechacha_talk.tbt. Accessed: 2013.

Balthasar Bickel, Martin Gaenszle, Novel Kishore Rai, Vishnu Singh Rai, Elena Lieven, Sabine Stoll, G. Banjade, T. N. Bhatta, N Paudyal, J Pettigrew, and M Rai, I. P.and Rai. 2013c. Tangkera. URL https://corpus1.mpi.nl/qfs1/media-archive/dobes_data/ChintangPuma/Chintang/Narratives/Annotations/tangkera_05.tbt. Accessed: 2013.

Balthasar Bickel, Sabine Stoll Stoll, Martin Gaenszle, Novel Kishor Rai, Elena Lieven, Goma Banjade, Toya Nath Bhatta, Netra Prasad Paudyal, Judith Pettigrew, Ichchha Purna Rai, Manoj Rai, Taras Zakharko, and Robert Schikowski. Audiovisual corpus of the chintang

language, including a longitudinal corpus of language acquisition by six children, paradigm sets, grammar sketches, ethnographic descriptions, and photographs. 2013d.

Manfred Bierwisch. 1963. *Grammatik des deutschen Verbs*, volume II of *Studia Grammatica*. Akademie Verlag.

Geert Evert Booij. 2002. *The morphology of Dutch*. Oxford University Press on Demand.

Gosse Bouma, JM van Koppen, Frank Landsbergen, JEJM Odijk, Ton van der Wouden, and Matje van de Camp. 2015. Enriching a descriptive grammar with treebank queries. In *Proceedings of the Fourteenth International Workshop on Treebanks and Linguistic Theories (TLT14)*, volume 14, pages 13–25.

Claire Bowern. 2011. Sivisa Titan: sketch grammar, texts, vocabulary based on material collected by p. josef meier and po minis. *Oceanic Linguistics Special Publications*, (38): iii–466.

Claire Bowern. 2012. *A grammar of Bardi*, volume 57. Walter de Gruyter.

Claire Bowern. 2019. Titan materials. *Digital collection managed by PARADISEC [Open Access]*. (Accessed January 2019).

David Carter. 1997. The treebanker: A tool for supervised training of parsed corpora. *arXiv preprint cmp-lg/9705008*.

Shobhana Lakshmi Chelliah. 2011. *A grammar of Meithei*, volume 17. Walter de Gruyter.

Shobhana Lakshmi Chelliah. 2019. Meithei texts. Manipur Digital Resources in UNT Digital Library. University of North Texas Libraries. (Accessed August 2019).

Keh-Jiann Chen and Yu-Ming Hsieh. 2004. Chinese treebanks and grammar extraction. In *International Conference on Natural Language Processing*, pages 655–663. Springer.

Noam Chomsky. 1995. *The Minimalist Program.* Number 28 in Current Studies in Linguistics. MIT Press, Cambridge, MA.

Bernard Comrie. 1989. *Language Universals & Linguistic Typology.* University of Chicago, Chicago, second edition.

Ann Copestake. 2002a. Definitions of typed feature structures. In Stephan Oepen, Dan Flickinger, Jun-ichi Tsujii, and Hans Uszkoreit, editors, *Collaborative Language Engineering*, pages 227–230. CSLI Publications, Stanford, CA.

Ann Copestake. 2002b. *Implementing typed feature structure grammars*, volume 110. CSLI publications Stanford.

Ann Copestake, Dan Flickinger, Carl Pollard, and Ivan A Sag. 2005. Minimal Recursion Semantics: An introduction. *Research on Language and Computation*, 3(2-3):281–332.

Greville G Corbett. 1991. Gender. *Cambridge: CUP.*

Greville G Corbett. 2000. Number. *Cambridge: CUP.*

Andrew Cowell. 2018. Arapaho text database. Version 1, 2018. University of Colorado, Department of Linguistics (Accessed at https://github.com/Adamits/arapaho_library/tree/master/data February 2020).

Andrew Cowell and Alonzo Moss Sr. 2011. *The Arapaho language.* University Press of Colorado.

Joshua Crowgey. 2019. *Braiding Language (by Computer): Lushootseed Grammar Engineering.* PhD thesis, University of Washington.

Joshua David Crowgey. 2012. The syntactic exponence of sentential negation: A model for the LinGO Grammar Matrix. Master's thesis, University of Washington.

Berthold Crysmann and Woodley Packard. 2012. Towards efficient HPSG generation for German, a non-configurational language. In *Proceedings of the 24th International Conference on Computational Linguistics*, pages 695–710.

Michael Cysouw. 2013. Inclusive/exclusive distinction in independent pronouns. In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig. URL `https://wals.info/chapter/39`.

Luis Morgado da Costa, Francis Bond, and Xiaoling He. 2016. Syntactic well-formedness diagnosis and error-based coaching in computer assisted language learning using machine translation. In *Proceedings of the 3rd Workshop on Natural Language Processing Techniques for Educational Applications (NLPTEA2016)*, pages 107–116.

Östen Dahl. 1979. Typology of sentence negation. *Linguistics*, 17:79–106.

Tifa de Almeida, Youyun Zhang, Kristen Howell, and Emily M Bender. 2019. Feature comparison across typological resources. *Unpublished abstract, presented at TypNLP.*

Jon Ortiz de Urbina. 1989. *Parameters in the grammar of Basque: A GB approach to Basque syntax*. Foris Pubns USA.

John M Dedrick and Eugene H Casad. 1999. *Sonora Yaqui Language Structures*. University of Arizona Press.

R. M. W. Dixon. 1994. *Ergativity*. Cambridge University Press, Cambridge.

Charles Donet. 2014a. The importance of verb salience in the followability of Lezgi oral narratives. Master's thesis, Dallas International University.

Charles Donet. 2014b. Lezgi oral narratives. Dallas International University. Unpublished FieldWorks (FLEx) project. (Accessed August 2019).

Scott Drellishak. 2004. A survey of coordination strategies in the world's languages. Master's thesis, University of Washington.

Scott Drellishak. 2009. *Widespread but not universal: Improving the typological coverage of the Grammar Matrix*. PhD thesis, University of Washington.

Scott Drellishak and Emily Bender. 2005. A coordination module for a crosslinguistic grammar resource. In *International Conference on Head-Driven Phrase Structure Grammar*, volume 12, pages 108–128. URL `http://web.stanford.edu/group/cslipublications/cslipublications/HPSG/2005/drellishak-bender.pdf`.

Rebecca Dridan and Stephan Oepen. 2011. Parser evaluation using elementary dependency matching. In *Proceedings of the 12th International Conference on Parsing Technologies*, pages 225–230.

Matthew S Dryer. 2005. Negative morphemes. In Martin Haspelmath, Matthew S. Dryer, David Gil, and Bernard Comrie, editors, *The World Atlas of Linguistic Structures (WALS)*, pages 454–457. Oxford University Press, Oxford.

Matthew S. Dryer. 2008. Expression of pronominal subjects. In Martin Haspelmath, Matthew Dryer, David Gil, and Bernard Comrie, editors, *The World Atlas of Language Structures Online*, chapter 101. Max Planck Digital Library. URL `http://wals.info/feature/101`.

Matthew S. Dryer. 2011. Order of subject, object and verb. In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Digital Library, Munich. URL `http://wals.info/feature/81A`.

Matthew S. Dryer. 2013a. Negative morphemes. In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig. URL `https://wals.info/chapter/112`.

Matthew S. Dryer. 2013b. Expression of pronominal subjects. In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck

Institute for Evolutionary Anthropology, Leipzig. URL https://wals.info/chapter/101.

Matthew S. Dryer. 2013c. Order of subject, object and verb. In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig. URL https://wals.info/chapter/81.

Matthew S. Dryer. 2013d. Order of demonstrative and noun. In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig. URL https://wals.info/chapter/88.

Matthew S. Dryer and Martin Haspelmath, editors. 2013. *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig. URL https://wals.info/.

David M. Eberhard, Gary F. Simons, and Charles D. Fennig (eds.). 2019. Ethnologue: Languages of the World. Twenty-second edition. URL http://www.ethnologue.com.

Dan Flickinger. 2000. On building a more efficient grammar by exploiting types. *Natural Language Engineering*, 6 (1) (Special Issue on Efficient Processing with HPSG):15 – 28.

Dan Flickinger. 2011. Accuracy v. robustness in grammar engineering. In Emily M. Bender and Jennifer E. Arnold, editors, *Language from a Cognitive Perspective: Grammar, Usage and Processing*, pages 31–50. CSLI Publications, Stanford, CA.

Dan Flickinger, Emily M. Bender, and Stephan Oepen. May 2014. Towards an encyclopedia of compositional semantics: Documenting the interface of the English resource grammar. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 875–881, Reykjavik, Iceland, May 2014. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2014/pdf/562_Paper.pdf.

Dan Flickinger, Stephan Oepen, and Emily M. Bender. 2017. *Sustainable Development and Refinement of Complex Linguistic Annotations at Scale*, pages 353–377. Springer Netherlands, Dordrecht. URL `http://dx.doi.org/10.1007/978-94-024-0881-2\_14`.

Antske Fokkens. 2010. Documentation for the Grammar Matrix word order library.

Antske Fokkens. 2014. *Enhancing Empirical Research for Linguistically Motivated Precision Grammars*. PhD thesis, Department of Computational Linguistics, Universität des Saarlandes.

Ryan Georgi. 2016. *From Aari to Zulu: Massively Multilingual Creation of Language Tools Using Interlinear Glossed Text*. PhD thesis, University of Washington.

Michael W Goodman and Emily M Bender. 2010. What's in a word? Refining the morphotactic infrastructure in the LinGO Grammar Matrix customization system. In *Workshop on morphology and formal grammar, Paris*.

Michael Wayne Goodman, Joshua Crowgey, Fei Xia, and Emily M. Bender. 2015. Xigt: Extensible interlinear gloss text for natural language processing. *Language Resources and Evaluation*, 49 (2):455–485.

Kenneth Locke Hale and Kenneth Locke Hale Samuel Jay Keyser. 2002. *Prolegomenon to a theory of argument structure*, volume 39. MIT press.

Kenneth Locke Hale and Samuel Jay Keyser. 1993. On argument structure and the lexical representation of semantic relations. In Kenneth Locke Hale and Samuel Jay Keyser, editors, *The view from Building 20: Essays in linguistics in honor of Sylvain Bromberger*. Mit Press.

Wenjuan Han, Ge Wang, Yong Jiang, and Kewei Tu. 2019. Multilingual grammar induction with continuous language identification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5732–5737.

Heidi Harley. 2011. A minimalist approach to argument structure. *The Oxford handbook of linguistic minimalism*, pages 426–447.

Heidi Harley. 2019. Haiki text corpus. University of Arizona. Unpublished FieldWorks (FLEx) project. (Accessed August 2019).

Martin Haspelmath. 2007. In Timothy Shopen, editor, *Language typology and syntactic description*, volume 2. Cambridge University Press, Cambridge, UK.

Bryn Hauk. 2016–2019. Tsova-tush lexicon and texts. University of Hawai'i at Mānoa. Unpublished FieldWorks (FLEx) project. V2019.08.20. 2016–2019 (collection date).

Bryn Hauk. 2020. *Deixis and reference tracking in Tsova-Tush*. PhD thesis, University of Hawai'i at Mānoa.

Bryn Hauk and Alice C. Harris. forthcoming. Batsbi. *The Caucasian languages: An international handbook*.

Lars Hellan. July 2010. From descriptive annotation to grammar specification. In *Proceedings of the Fourth Linguistic Annotation Workshop*, pages 172–176, Uppsala, Sweden, July 2010. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/W10-1826`.

Lars Hellan and Dorothee Beermann. 2011. Inducing grammars from IGT. In Mariani J. Vetulani Z., editor, *Human Language Technology Challenges for Computer Science and Linguistics.*, volume 8287 of *LTC 2011. Lecture Notes in Computer Science*. Springer.

Lars Hellan, Tore Bruland, Elias Aamot, and Mads H Sandøy. 2013. A grammar sparrer for norwegian. In *Proceedings of the 19th Nordic Conference of Computational Linguistics (NODALIDA 2013); May 22-24; 2013; Oslo University; Norway. NEALT Proceedings Series 16*, number 085, pages 435–439. Linköping University Electronic Press.

John Hinds. 1986. *Japanese: descriptive grammar*. Routledge.

David Holton, Peter Mackridge, Irene Philippaki-Warburton, and Vassilios Spyropoulos. 2012. *Greek: A comprehensive grammar of the modern language*. Routledge.

Paul J Hopper. 1982. *Tense-aspect: between semantics & pragmatics: containing the contributions to a Symposium on Tense and Aspect, held at UCLA, May 1979*, volume 1. John Benjamins Publishing.

Kristen Howell and Olga Zamaraeva. 2018. Clausal modifiers in the Grammar Matrix. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2939–2952.

Kristen Howell, Emily M. Bender, Michael Lockwood, Fei Xia, and Olga Zamaraeva. 2017. Inferring case systems from IGT: Impacts and detection of variable glossing practices. *ComputEL-2*, pages 67–75.

Kristen Howell, Olga Zamaraeva, and Emily M. Bender. 2018. Nominalized clauses in the Grammar Matrix. In *The 25th International Conference on Head-Driven Phrase Structure Grammar*.

Sagar Indurkhya. 2020. Inferring minimalist grammars with an smt-solver. In *Proceedings of the Society for Computation in Linguistics*, volume 3.

David Inman. 2015. Pronoun incorporation in Matsigenka. URL `http://compling.hss.ntu.edu.sg/events/2015-hpsg/pdf/Inman.pdf`.

David Inman. 2019a. *Multi-predicate Constructions in Nuuchahnulth*. PhD thesis, University of Washington.

David Inman. 2019b. Nuuchahnulth texts. University of Washington. Unpublished FieldWorks (FLEx) project. (Accessed March 2019).

Bevan Keeley Jones, Sharon Goldwater, and Mark Johnson. 2013. Modeling graph languages

with grammars extracted via tree decompositions. In *Proceedings of the 11th International Conference on Finite State Methods and Natural Language Processing*, pages 54–62.

Ronald M Kaplan and Joan Bresnan. 1982. Lexical-functional grammar: A formal system for grammatical representation. *Formal Issues in Lexical-Functional Grammar*, (47):29–130.

Rohit J Kate and Raymond J Mooney. 2006. Using string-kernels for learning semantic parsers. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 913–920. Association for Computational Linguistics.

Rohit J Kate, Yuk Wah Wong, and Raymond J Mooney. 2005. Learning to transform natural to formal languages. In *AAAI*, pages 1062–1068.

Daniel Kaufman, Husniya Khujamyorova, and Ross Perlin. 2020. Wakhi texts. *Digital collection managed by KRATYLOS*. Uploaded from www.elalliance.org, Wakhi. In Finkel, R. and Kaufman, D., Kratylos: Unified Linguistic Corpora from Diverse Data Sources. Uploaded April 28, 2020 and retrieved from https://www.cs.uky.edu/ raphael/ela/ on May 20 2020.

István Kenesei, Robert M Vago, and Anna Fenyvesi. 2002. *Hungarian*. Routledge.

Dan Klein and Christopher Manning. 2002. A general constituent context model for improved grammar induction. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002)*, Philadelphia, PA, 2002.

Dan Klein and Christopher Manning. 2004. Corpus-based induction of syntactic structure: models of dependency and constituency. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-2004)*, Barcelona, Spain, 2004.

Dan Klein and Christopher D Manning. 2001. Natural language grammar induction using a constituent-context model. In *Advances in neural information processing systems*, pages 35–42.

Jaklin Kornfilt. 1997. *Turkish*. Routledge, London.

František Kratochvíl. 2007. *A grammar of Abui*. Utrecht: LOT.

František Kratochvíl. 2019. Abui Corpus. Electronic Database: Unpublished toolbox project (accessed March 2019). Nanyang Technological University, Singapore.

Alexander Krotov, Robert Gaizauskas, and Yorick Wilks. 1994. Acquiring a stochastic context-free grammar from the penn treebank.

Alexander Krotov, Mark Hepple, Robert Gaizauskas, and Yorick Wilks. 1998. Compacting the Penn Treebank Grammar. In *Proc. of the 36th Annual Meeting of the Association for Computational Linguistics (ACL-1998)*, Montreal, Quebec, Canada, 1998.

Tom Kwiatkowski, Luke Zettlemoyer, Sharon Goldwater, and Mark Steedman. 2011. Lexical generalization in ccg grammar induction for semantic parsing. In *Proceedings of the conference on empirical methods in natural language processing*, pages 1512–1523. Association for Computational Linguistics.

Ned Letcher and Timothy Baldwin. 2013. Constructing a phenomenal corpus: Towards detecting linguistic phenomena in precision grammars. In *Workshop on High-level Methodologies for Grammar Engineering@ ESSLLI 2013*, page 25.

Charles N Li and Sandra A Thompson. 1989. *Mandarin Chinese: A functional reference grammar*. University of California Press.

Michael Lockwood. 2016. Automated gloss mapping for inferring grammatical properties. Master's thesis, University of Washington.

Mitchell Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The penn treebank: annotating predicate argument structure. In *Proceedings of the workshop on Human Language Technology*, pages 114–119. Association for Computational Linguistics.

Alfred Master and I General. 1946. The zero negative in Dravidian. *Transactions of the Philological Society*, 45(1):137–155.

Sérgio Meira. 2020. Hixkaryana lexicon and texts. Unpublished Toolbox project. (Accessed March 2020).

Lev Michael, Christine Beier, Zachary O'Hagan, (compilers), Haroldo Vargas, José Vargas, and (authors). 2013. Matsigenka text corpus (version june 2013; flex database and latex interlinear output).

Lev David Michael. 2008. *Nanti evidential practice: Language, knowledge, and social action in an Amazonian society*. PhD thesis, Univ. of Texas Austin.

Matti Miestamo. 2008. *Standard negation: The negation of declarative verbal main clauses in a typological perspective*, volume 31. Walter de Gruyter.

Osahito Miyaoka. 2012. *A Grammar of Central Alaskan Yupik (CAY)*, volume 58. Walter de Gruyter.

Paola Monachesi. 1996. *A grammar of Italian clitics*. ITK Dissertations Series 1996-1.

Stefan Müller. 1999. *Deutsche Syntax deklarativ. Head-Driven Phrase Structure Grammar für das Deutsche*. Max Niemeyer Verlag, Tübingen, Germany.

Pieter Muysken, Harald Hammarström, Olga Krasnoukhova, Neele Müller, Joshua Birchall, Simon van de Kerke, Loretta O'Connor, Swintha Danielsen, Rik van Gijn, and George Saad, editors. 2016. *South American Indigenous Language Structures (SAILS) Online*. Max Planck Institute for the Science of Human History. URL `https://sails.clld.org`.

Paul Newman. 2000. *The Hausa language: An encyclopedic reference grammar*. Yale University Press.

Elizabeth Nielsen and Emily M. Bender. 2018. Modeling adnominal possession in multilingual grammar engineering. In Stefan Müller and Frank Richter, editors, *Proceedings*

of the 25th International Conference on Head-Driven Phrase Structure Grammar, University of Tokyo, pages 140–153, Stanford, CA, 2018. CSLI Publications. URL `http://cslipublications.stanford.edu/HPSG/2018/hpsg2018-nielsen-bender.pdf`.

Elizabeth K Nielsen. 2018. Modeling adnominal possession in the LinGO Grammar Matrix. Master's thesis, University of Washington.

Joakim Nivre, Marie-Catherine De Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, et al. 2016. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666.

Hiroshi Noji, Yusuke Miyao, and Mark Johnson. 2016. Using left-corner parsing to encode universal structural constraints in grammar induction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 33–43.

Rachel Nordlinger. 1998. *A Grammar of Wambaya, Northern Australia*. Pacific Linguistics.

Stephan Oepen. [incr tsdb()] — Competence and performance laboratory. User manual. Technical report, Computational Linguistics — Saarland University, Saarbrücken, Germany, 2001.

Stephan Oepen, Kristina Toutanova, Stuart Shieber, Chris Manning, Dan Flickinger, and Thorsten Brants. 2002. The LinGO Redwoods treebank. Motivation and preliminary applications. Taipei, Taiwan, 2002.

Stephan Oepen, Daniel Flickinger, Kristina Toutanova, and Christopher D. Manning. 2004. LinGO Redwoods. A rich and dynamic treebank for HPSG. *Journal of Research on Language and Computation*, 2(4):575 – 596.

Zachary O'Hagan. 2018. The syntax of Matsigenka object-marking. *Berkeley Papers in Formal Linguistics*, 1(1).

Woodley Packard. 2015. Full Forest Treebanking. Master's thesis, University of Washington.

Carl Pollard and Ivan A Sag. 1994. *Head-Driven Phrase Structure Grammar*. University of Chicago Press.

Laurie Poulson. 2011. Meta-modeling of tense and aspect in a cross-linguistic grammar engineering platform. *University of Washington Working Papers in Linguistics (UWWPL)*, 28.

Chris Rogers. 2010. Fieldworks language explorer (flex) 3.0. *Language Documentation & Conservation*, 4.

Safiyyah Saleem. 2010. Argument optionality: A new library for the Grammar Matrix customization system. Master's thesis, University of Washington.

Safiyyah Saleem and Emily M Bender. 2010. Argument optionality in the LinGO Grammar Matrix. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 1068–1076. Association for Computational Linguistics.

Jose Sanchez, Alex Trueman, Maria Florez Leyva, Santos Leyva Alvarez, Mercedes Tubino Blanco, Hyun-Kyoung Jung, Louise St. Amour, and Heidi Harley. 2015. *An Introduction to Hiaki Grammar*. University of Arizona Press.

Vee Satayamas and Asanee Kawtrakul. 2004. Wide-coverage grammar extraction from thai treebank. In *Proceedings of Papillon 2004 Workshops on Multilingual Lexical Databases*.

Robert Schikowski. 2013. *Object-conditioned differential marking in Chintang and Nepali*. PhD thesis, University of Zurich.

Terrill B Schrock et al. 2014. *A grammar of Ik (Icé-tód): Northeast Uganda's last thriving Kuliak language*. LOT: Utrecht.

Kiyoaki Shirai, Takenobu Tokunaga, and Hozumi Tanaka. 1995. Automatic extraction of Japanese grammar from a bracketed corpus. In *Proceedings of Natural Language Processing Pacific Rim Symposium*, pages 211–216.

Melanie Siegel, Emily M Bender, and Francis Bond. 2016. *Jacy: An implemented grammar of Japanese*. CSLI Publications.

Siewierska. 2004. *Person (Cambridge textbooks in linguistics)*. Cambridge University Press.

SIL International. Field Linguist's Toolbox. Lexicon and corpus management system with a parser and concordancer; URL: http://www-01.sil.org/computing/toolbox/documentation.htm, 2015.

Kiril Simov. 2002. Grammar extraction and refinement from an hpsg corpus. In *Proc. of the ESSLLI Workshop on Machine Learning Approaches in Computational Linguistics*, pages 38–55.

Noah A. Smith and Jason Eisner. July 2006. Annealing structural bias in multilingual weighted grammar induction. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (ACL/COLING 2006)*, pages 569–576, Sydney, Australia, July 2006. Association for Computational Linguistics.

James Neil Sneddon, K Alexander Adelaar, Dwi N Djenar, and Michael Ewing. 2012. *Indonesian: A comprehensive grammar*. Routledge.

Ho-Min Sohn. 1994. Korean: A descriptive grammar. *London and New York: Routledge*.

Edward Stabler. 1996. Derivational minimalism. In *International Conference on Logical Aspects of Computational Linguistics*, pages 68–95. Springer.

Mark Steedman. 2000. *The syntactic process*, volume 24. MIT press Cambridge, MA.

Helena Sulkala and Merja Karjalainen. 1992. Finnish. *London & New York NY: Routledge*.

John Sylak-Glassman, Christo Kirov, David Yarowsky, and Roger Que. 2015. A language-independent feature schema for inflectional morphology. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 674–680.

Nick Thieberger. 2006a. *A grammar of South Efate: an Oceanic language of Vanuatu*, volume 33. University of Hawaii Press.

Nick Thieberger. 2006b. Dictionary and texts in South Efate. *Digital collection managed by PARADISEC [Open Access]*. (Accessed March 2019).

Thomas James Trimble. 2014. Adjectives in the LinGO Grammar Matrix. Master's thesis, University of Washington.

David Wax. 2014. Automated grammar engineering for verbal morphology. Master's thesis, University of Washington.

Fei Xia. 1999. Extracting tree adjoining grammars from bracketed corpora. In *Proc. of 5th Natural Language Processing Pacific Rim Symposium (NLPRS-1999)*, Beijing, China, 1999.

Fei Xia and William D. Lewis. 2007. Multilingual structural projection across interlinear text. In *Proc. of the Conference on Human Language Technologies (HLT/NAACL 2007)*, pages 452–459, Rochester, New York, 2007.

Fei Xia, William D. Lewis, Michael Wayne Goodman, Glenn Slayden, Ryan Georgi, Joshua Crowgey, and Emily M. Bender. 2016. Enriching a massively multilingual database of interlinear glossed text. *Language Resources and Evaluation*, 50:321–349.

Olga Zamaraeva. 2016. Inferring morphotactics from interlinear glossed text: combining clustering and precision grammars. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 141–150.

Olga Zamaraeva. in prep. *A cross-linguistic account of constituent questions for a grammar engineering resource*. PhD thesis, University of Washington.

Olga Zamaraeva, František Kratochvíl, Emily M Bender, Fei Xia, and Kristen Howell. 2017. Computational support for finding word classes: A case study of Abui. In *Proceedings of the 2nd Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 130–140.

Olga Zamaraeva, Kristen Howell, and Emily M Bender. 2019a. Handling cross-cutting properties in automatic inference of lexical classes: A case study of Chintang. In *Proceedings of the 3rd Workshop on the Use of Computational Methods in the Study of Endangered Languages*.

Olga Zamaraeva, Kristen Howell, and Emily M. Bender. 2019b. Modeling clausal complementation for a grammar engineering resource. In *Proceedings of the 2nd meeting of the Society for Computation in Linguistics*.

Luke Zettlemoyer and Michael Collins. 2007. Online learning of relaxed CCG grammars for parsing to logical form. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 678–687.

Luke S Zettlemoyer and Michael Collins. 2005. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*.

Youyun Zhang. in prep. Using typological databases to augment grammar inference. Master's thesis, University of Washington.

Arnold Zwicky, Joyce Friedman, Barbara C. Hall, and D.E. Walker. 1965. The MITRE syntactic analysis procedure for transformational grammars. In *Proc. Fall Joint Computer Conference*, volume 67, Pt 1, pages 317–326.

Arnold M Zwicky and Geoffrey K Pullum. 1983. Cliticization vs. inflection: English n't. *Language*, pages 502–513.

Appendix A

DATA AND CODE REPOSITORIES

A.1 Data Repositories

Alaskan Native Languages Archive (ANLA)

`https://www.uaf.edu/anla/`

Archive of Indigenous Languages in Latin America (AILLA)

`http://www.ailla.utexas.org/site/welcome.html`

Endangered Languages Archive (ELAR)

`http://elar.soas.ac.uk/`

Kaipuleohone

`https://scholarspace.manoa.hawaii.edu/handle/10125/4250`

Kratylos

`https://www.kratylos.org/~kratylos/home.cgi`

Multi-CAST

`https://multicast.aspra.uni-bamberg.de/`

ODIN

`http://depts.washington.edu/uwcl/odin/`

Pacific and Regional Archive for Digital Sources (PARADISEC)

`http://www.paradisec.org.au/`

A.2 Code and Project Repositories

ACE

`http://sweaglesw.org/linguistics/ace/`

AGGREGATION, BASIL

`https://git.ling.washington.edu/agg`

DELPH-IN

`www.delph-in.net`

INTENT

`https://github.com/rgeorgi/INTENT2`

FFTB

`http://moin.delph-in.net/FftbTop`

Grammar Matrix

`http://matrix.ling.washington.edu/index.html`

MOM

`https://git.ling.washington.edu/agg/mom`

Xigt

`https://github.com/xigt/xigt`

Appendix B

CHANGES TO BASIL FOR HELD-OUT EVALUATION

Before evaluation on held-out languages, I froze system development in order to test how well BASIL would generalize to new languages and language families. However, the held-out datasets had some differences from the development datasets, which the inference code did not properly handle. In order to allow BASIL to run completely and produce a grammar specification from which the Grammar Matrix could produce a grammar, I made the following minor changes to the inference code.

B.1 Handling Missing Tiers

BASIL expected the corpora to contain at least three line IGT, comprising a language line, a gloss line and a translation line. However, some of the held out corpora contained IGT that did not include either a gloss or translation line.

The auxiliary inference module (infer_aux.py) assumes a "translation words" tier, which is constructed from the translation line. The absence of this tier in some IGT resulted in the inference script crashing. I resolved this by adding an exception to skip IGT without this tier.

Similarly, coordination inference (infer_coordination.py) assumes a "glosses tier", which is constructed from the gloss line. Again the absence of this tier in the held-out data caused the script to break, so I add an exception to skip IGT without this tier.

B.2 Removing Punctuation from Orthography

Part of constructing the lexicon section of the grammar specification involves removing punctuation from the orthographic representations. The coordination inference module adds the

orthographic representations for coordinators to the grammar specification, but as an oversight, BASIL did not remove punctuation from these orthographies, as this did not cause an issue in the development datasets. Coordinators with quotation marks in the orthography caused an error in the Grammar Matrix customization system. To avoid this, I corrected the coordinator inference module (infer_coordination.py) to remove quotations from the orthographic representations for coordinators.