Computational Support for Finding Word Classes: A Case Study of Abui

Olga Zamaraeva[*], **František Kratochvíl**[**], **Emily M. Bender**[*], **Fei Xia**[*] **and Kristen Howell**[*]

[*]Department of Linguistics, University of Washington, Seattle, WA, U.S.A.
[**]Division of Linguistics and Multilingual Studies, Nanyang Technological University, Singapore
{olzama, ebender, fxia, kphowell}@uw.edu, fkratochvil@ntu.edu.sg

Abstract

We present a system that automatically groups verb stems into inflection classes, performing a case study of Abui verbs. Starting from a relatively small number of fully glossed Abui sentences, we train a morphological precision grammar and use it to automatically analyze and gloss words from the unglossed portion of our corpus. Then we group stems into classes based on their cooccurrence patterns with several prefix series of interest. We compare our results to a curated collection of elicited examples and illustrate how our approach can be useful for field linguists as it can help them refine their analysis by accounting for more patterns in the data.

1 Introduction

Computational methods can play a major role in endangered language documentation by producing summaries of collected data that identify apparent patterns in the data as well as exceptions to those patterns. On the one hand, this can help identify errors in glossing (where apparent exceptions are merely typos or orthographic idiosyncrasies). On the other hand, it can help the linguist understand and model patterns in the data, especially in cases where the phenomena in question have overlapping distributions. In this paper, we undertake a case study of verb classes in Abui [abz] in light of the morphotactic inference system of the AGGREGATION project (Bender et al., 2014; Wax, 2014; Zamaraeva, 2016).

We begin with an overview of the phenomenon that is the focus of our case study (§2), formulate the problem and describe the steps to solve it (§3). Next we describe the tools and algorithms we apply to compare the output of the system (which summarizes what is found in the accessible corpus data) with a set of elicited judgments (§4). We conclude with a discussion of where the results of our system differ (§5).

2 Abui verb classes

In this section, we provide a brief introduction to Abui verbs, with respect to what prefixes the verbs can take.

2.1 Abui and Undergoer Marking

Abui [abz] is an Alor-Pantar language of Eastern Indonesia. František Kratochvíl and colleagues have collected and transcribed a corpus comprising roughly 18,000 sentences, of which about 4,600 have been glossed (Kratochvíl, 2017).

Abui is notable for its argument realization, which Kratochvíl (2007; 2011) argues is sensitive to semantic rather than syntactic features. A key part of this system is a collection of five prefix series that can attach to verbs which index different undergoer-like arguments. For the most part, each undergoer type has a phonologically distinct paradigm (e.g. PAT prefixes tend to end in *a*); the full paradigm is given in Table 1.[1] The prefixes occur in both first and second (and in some cases, third) position with respect to the stem, though in this paper we will focus on the first position only.

PERSON	PAT	REC	LOC	GOAL	BEN
1S	*na-*	*no-*	*ne-*	*noo-*	*nee-*
2S	*a-*	*o-*	*e-*	*oo-*	*ee-*
1PE	*ni-*	*nu-*	*ni-*	*nuu-*	*nii-*
1PI	*pi-*	*pu-/po-*	*pi-*	*puu-/poo-*	*pii-*
2P	*ri-*	*ro-/ru-*	*ri-*	*ruu-/roo-*	*rii-*
3	*ha-*	*ho-*	*he-*	*hoo-*	*hee-*
3I	*da-*	*do-*	*de-*	*doo-*	*dee-*
DISTR	*ta-*	*to-*	*te-*	*too-*	*tee-*

Table 1: Abui person indexing paradigm

An example of the prefix attachment is given in (1) where the stem *mia* ‘take’ agrees in person and

[1] The 3I mostly create reflexives. The (DISTR) prefixes index reciprocals and distributive.

number with the noun *aloba* ‘thorn’. The subject *na* (1SG.AGT) is not indexed on the verb.[2]

(1) Na aloba he-mia
1SG.AGT [thorn]$_{\text{LOC}}$ 3UND.LOC-take.IPFV
‘I am taking out the thorn.’ [abz; N12.064]

The five undergoer prefix series mark distinctions among different undergoer-like roles. Many verbs can co-occur with different undergoer prefixes (Kratochvíl, 2014; Fedden et al., 2014; Kratochvíl and Delpada, 2015). Accordingly, the prefixes (and the role distinctions they mark) can be analyzed as contributing to the interpretation of the event. This is illustrated in alternations such as *he-komangdi* ‘make it less sharp’ ∼ *ha-komangdi* ‘make it blunt’; *he-bel* ‘pluck it’ ∼ *ha-bel* ‘pull it out’; *he-fanga* ‘say it’ ∼ *ha-fanga* ‘order him’ (Kratochvíl and Delpada, 2015).

In order to better understand the semantic contribution of these prefixes, we would like a rich, detailed description of their distribution in the corpus of naturally occurring speech. In particular, looking at verb classes defined in terms of undergoer prefix compatibility is a promising avenue for better understanding the semantic restrictions on and contributions of the prefixes themselves.

2.2 Abui Undergoer Prefix Glossing

Each prefix series marks one undergoer type, but varies by person and number. The undergoer type of the prefix can in principle be consistently inferred from its phonological form. Specifically, each series ends with a characteristic vowel pattern, at least in the singular, which seems to have a much higher frequency in the corpus. The patterns are shown in Table 2 together with the gloss labels typically used for each series. The *C-* at the start of each prefix form represents the consonants which vary with series. The gloss labels are suggestive of semantic patterns, but the exact semantic contribution of the prefixes is ultimately what we are working towards understanding and thus these labels should be interpreted as preliminary place-holders.

As is typical for active field projects, the glossing is not consistent across the corpus. Most relevantly for our purposes, *Co-* prefixes are sometimes glossed as GOAL (i.e. the same as the *Coo-* prefixes) and *Ce-* prefixes are sometimes glossed as BEN (i.e. the same as the *Cee-* prefixes). For the purposes of our present study, we work around these glossing inconsistencies by merging the prefix classes (treating *Ce-* and *Cee-* as one class and *Co-* and *Coo-* as one), effectively reverting (temporarily) to the older analysis in Kratochvíl (2007). This is indicated in Table 2 by the shared Condition numbers for these series. At this stage we also exclude any forms that do not end with *a-*, *o-*, or *e-*, from the analysis.

Form	Gloss	Condition
Ø-	stem alone	I
Ca-	patient (PAT)	II
Ce-	location (LOC)	III
Cee-	benefactive (BEN)	III
Co-	recipient (REC)	IV
Coo-	goal (GOAL)	IV

Table 2: Prefix forms and glosses; Condition I is stem attested bare.

Stem	I	II	III	IV	Class
fil ‘pull’	+	+	+	+	A (*1111*)
kaanra ‘complete’	+	+	+	+	A (*1111*)
kafia ‘scratch’	+	-	+	+	B (*1011*)
yaa ‘go’	+	-	+	+	B (*1011*)
mpang ‘think’	+	-	-	+	C (*1001*)
bel ‘pull out’	-	+	+	+	D (*0111*)
luk ‘bend’	-	-	+	+	E (*0011*)

Table 3: Examples of Abui verb classes

Together, Conditions I-IV define 16 possible verb classes,[3] where a class is defined by the property of being able to appear in each Condition. In Table 3, we illustrate this with seven verbal stems. We track whether these stems can occur freely (Condition I); whether they are compatible with the prefix *Ca-* (PAT), Condition II; prefix *Co-* (REC) or *Coo-* (GOAL), Condition III; prefix *Ce-* (LOC), or *Cee-* (BEN), Condition IV.

In Table 3, the first five stems (Row 1-5) can occur freely (without affixes—Condition I). Of these five, only the first two stems are also compatible with all other prefix series (Conditions II-IV). The remaining stems form three distinct inflectional classes, labeled with capital letters in the last column of the table and with a binary code which can be used to easily decipher the nature of the class (e.g. class *1111*: all combinations are possible).

[2] According to Siewierska (2013), systems marking undergoers alone (leaving actors unmarked) are rare, constituting only about 7% of her sample. In the Alor-Pantar family, undergoer marking is a common trait.

[3] In practice, we will have 15 classes, since class *‘0000’* (no Condition applies) cannot be described without additional Conditions, such as a the presence of a light verb.

The question we are investigating here is which verbs appear to belong to which of these inflectional classes, according to the collected corpora. We have created a set of elicited judgments for 337[4] verbs regarding their compatibility with the different undergoer prefixes (Kratochvíl and Delpada, 2017). Our goal is to provide a summary of attested verb-prefix combinations, from both glossed and as yet unglossed corpus data, and compare it to the elicited judgments. In the following sections, we describe how the systems developed by the AGGREGATION project can be used to these ends.

3 Methods for computational support

Classifying verbal stems according to the Conditions we have outlined is a cooccurrence problem: given segmented and glossed IGT, we seek to determine which stems co-occur with which types of affixes. Presently, the Abui corpus is managed using the SIL Toolbox (SIL International, 2015), which allows simple concordance functions, but does not support the kind of distributional analysis we are engaging in here.[5] The AGGREGATION project machinery, which is concerned with building precision grammars, finds cooccurrence patterns of affixes as part of the morphological rules inference. Thus we are taking advantage of the existing pipeline and do not have to create an additional piece of software for this task.

In addition to providing the cooccurrence analysis, the AGGREGATION machinery offers a crucial benefit: It is building a full-fledged morphological analyzer, which we can use to automatically analyze words that have not yet been manually glossed. This gives us more data and helps find more instances of the hypothesized verb classes.[6] Inferring a morphological grammar automatically from IGT is one of the AGGREGATION project's principal subtasks (see e.g. Bender et al. (2014)), and in this paper we are taking it one step further by actually using the inferred grammar to help develop resources for Abui. The process is outlined in Figure 1 and explained in the next section.

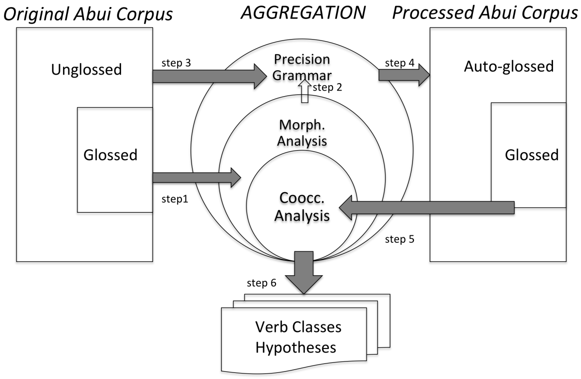

Figure 1: The components of the process. Cooccurrence and morphological analysis are separate processes which both belong to AGGREGATION pipeline. Morphological analysis is converted to a precision morphological grammar, which is used to parse the unglossed parts of the corpus.

4 System overview

In this section, we describe the systems we use to perform distributional analysis and the analysis of the unglossed corpus. We start with a brief description of precision grammars and systems for generating them from language descriptions (§4.1) before turning to software (dubbed 'MOM') for extracting such descriptions for the morphotactic component of precision grammars from interlinear glossed text (IGT; §4.2). We then describe how the system was straightforwardly extended to model verb classes in Abui (§4.3), and finally, how we used the resulting precision grammar to produce hypothesized glosses for parts of the unglossed corpus (§4.4).

4.1 Precision Grammars and the AGGREGATION project

A precision grammar is a machine-readable encoding of linguistic rules that supports the automatic association of analyses (e.g. morphological or syntactic parses or semantic representations) with strings. As argued in Bender et al. (2012), precision grammars can be useful for language documentation. Here we explore how they can

[4] There are actually 347 verbs in the set, but for the purposes of this paper we do not distinguish between homophones; we compare verbs by orthography only. This lets us compare between e.g. *fanga* ('say') in the curated set and *fanga* ('tell') in the corpus.

[5] The SIL Toolbox system also does not have a functioning consistency check function. Migration to other systems such as FLEX (SIL International, 2017) is not ideal, because the glossed part of the IGT (worth hundreds of man-hours) would be lost during the transfer. The FLEX system also does not support linked audio recordings which help to refine the transcription.

[6] There are further potential analyses facilitated by our methodology, including an exploration of cases where multiple prefixes occur together. We leave these to future work.

help provide hypothesized glosses for as-yet unanalyzed text.

Precision grammars are expensive to build, requiring intensive work by highly trained grammar engineers. The Grammar Matrix project (Bender et al., 2002; Bender et al., 2010) aims to reduce the cost of creating precision grammars by producing a starter-kit that automatically creates small precision grammars on the basis of lexical and typological language profiles. The AGGREGATION Project (Bender et al., 2014) is further building on this by applying the methods of Lewis and Xia (2008) and Georgi (2016) to extract language profiles suitable for input into the Grammar Matrix grammar customization system from existing collections of IGT.

In this paper, we focus on the morphotactic component of these systems. The Grammar Matrix's morphotactic system (O'Hara, 2008; Goodman, 2013) allows users to specify position classes and lexical rules. Position classes define the order of affixes with respect to each other (and the stem) while lexical rules pair affix forms with morphosyntactic or morphosemantic features. The grammars created on the basis of this information contain morphological subcomponents that can recognize well-formed strings of stems and affixes and associate them with feature structures that represent the information encoded by the morphemes and are compatible with the rules for syntactic structures. Here we develop only the morphological component, and leave the syntax in an underspecified state. We take advantage of the embedded morphological parser by analyzing words individually. In the following two subsections, we describe the Matrix-ODIN-Morphology (MOM) system and how we use it both for distributional analysis of verb stems (vis à vis prefix classes) and to create a grammar which we then use to parse the unglossed portion of the corpus. Note that it is straightforward to find morpheme cooccurrences in segmented data, and the fact that we use MOM for it is a matter of convenience. However, using MOM to automatically gloss words is a novel approach which we describe in detail.

4.2 Matrix-ODIN Morphology

The goal of the Matrix-ODIN Morphology or 'MOM' system (Wax, 2014; Zamaraeva, 2016) is to extract, from a corpus of IGT, the information required to create a computational morphological grammar of the regularized forms found in the IGT. Specifically, this information includes: (i) a set of affixes, grouped into position classes; (ii) for each affix, the form of the affix (and eventually, the associated morphosyntactic or morphosemantic features, as indicated by the glosses); (iii) for each position class, the inputs it can take (i.e. which other position classes it can attach to).

The MOM system first observes affix instances in the data (relying on the segmentation provided in IGT) and where in the word they are attached. Affix instances with the same form and the same gloss are considered to be examples of the same morpheme. Affixes are then recursively grouped into position classes on the basis of overlap in the sets of morphemes they are observed to take as input (attach to).[7] The degree of overlap is a tuneable parameter of the system.

The relationships between the affixes can then be expressed as a graph where groups of morphemes are nodes and input relations are directed edges. The graph can be used to specify the morphotactic component of a precision grammar.

Suppose our corpus of IGT consists of the one sentence in (2).

(2) he-ha-luol tila bataa ha-tang
3LOC-3PAT-gather rope tree 3PAT-hand
he-tilak-a mai neng nuku di
3LOC-hanging-CONT and.then man one 3A
mi ya ho-pun-a ba
take SEQ 3REC-grab.PFV-CONT SIM
natea.
rise.IPFV

'In the next one, there was a rope hanging on the tree branch when a man came and took it and remained standing there holding it.' [abz]

Initially, MOM will collect affix instances and represent the data as the graph in Figure 2 shows. Then, since nodes *verb-pc3* and *verb-pc1* have 100% overlap in incoming edges, they will be merged into one position class, as Figure 3 shows.

Note that the grammar in Figure 2 cannot generate *he-ho-verb1-a*, but the grammar in Figure 3 can thanks to the merge. In other words, MOM

[7] MOM (Wax, 2014) models linear ordering of affixes as follows. In the string *p2-p1-stem-s1*, *p1-* will be assumed to apply first, then *p2-*, and finally *-s1*. Internally, *s1-* will be modeled to take *p2-* as input rather then the stem. Inputs (rather than outputs) are considered the defining property of the position class, which is consistent with the theoretical model of position class morphology as outlined in e.g. Crysmann and Bonami (2012).

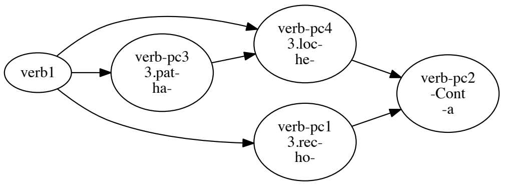

Figure 2: MOM-generated graph which groups affix instances seen in the data into types, and reflects the order in which the affixes were seen with respect to the stems and to each other. Prefixes and suffixes are distinct which is seen in their orthography in the figure.

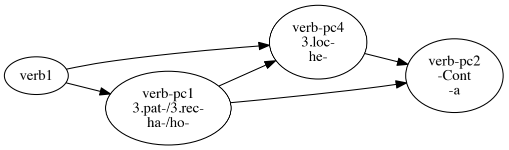

Figure 3: MOM-generated graph which has combined two affix types in Figure 2 into one position class, based on the overlap of their incoming edges.

generalizes patterns seen in the data based on its definition of position class to potentially increase the coverage of the precision grammar.

4.3 Extending MOM to model verb classes

The MOM system as implemented by Wax (2014) and developed further by Zamaraeva (2016) considers all stems of the same part of speech to belong to a single class, and then groups affixes based on what they can attach to (e.g. directly to the stem or to specific other affixes). Setting input overlap to 100% is equivalent to grouping affixes by their cooccurrence pattern (right-hand context for prefixes and left-hand context for suffixes).

We straightforwardly extended the system to model classes of stems (within parts of speech) based on which affixes attach to them. We modify MOM to initially put every new stem into its own node.[8] Then running the edge overlap algorithm described in §4.2 on the outgoing edges of the stem nodes with overlap set to 100% outputs verb classes defined by their cooccurrence with the first attaching prefix. Finally, computing overlap only over the edges that represent the undergoer prefixes yields a set of hypothesized Abui verb classes as they are described in §1.

4.4 Applying a Morphological Grammar to the Unglossed Part of the Corpus

The goal of grouping verbs by their cooccurrence patterns could be achieved by a variety of simple methods and there is no specific benefit in finding these patterns using MOM. However, when we set out to aggregate the cooccurrence information, we found only about 8,000 verbs in the glossed portion of the Abui corpus (see Table 4). In order to extend our analysis to the unglossed portion, we took advantage of the ability to turn MOM output into a precision grammar via the Grammar Matrix customization system. This grammar, like all grammars output by the customization system, is compatible with the ACE parsing engine (Crysmann and Packard, 2012).

The specification for the morphological grammar is created by running MOM on the glossed portion of the corpus, exactly as described in §4.2, treating all stems as one class, with affix input overlap set to 100%. The grammar is customized by the Grammar Matrix customization system and loaded into ACE. Then, the unglossed portion of the corpus is converted into a list of words, and each word token from the unglossed portion is parsed with ACE and the derived grammar.

Of the $12,034$ total unique words extracted from the unglossed part of the corpus, the ACE parser was able to find morphological analyses for $4,642$ words. The remainder are either not verbs, based on verb stems not attested in the glossed portion of the corpus,[9] or affix combinations not anticipated by our derived grammar.

Because Abui has many short stems and affixes (e.g. consisting of one phoneme), there are typically many ways to segment a word, and the parser produces multiple analyses for most words for which there was a successful parse; we pick one parse based on the following heuristic.

As explained in §2, for the purposes of this paper there are three prefix series of interest, namely, the ones that end with *a*-, *o*-, and *e*-. Of them, some will end with *oo*- and *ee*-. We rank higher the

[8] A new stem for the system is the orthography, normalized by lowercasing and stripping punctuation, which has not been seen before.

[9] The ACE parser in principle allows unknown word handling which would allow us to extend our analysis to unseen verb stems. This is left for future work.

parses where the prefixes of interest are adjacent to the stem; furthermore, we rank higher the long vowel prefixes ending in *oo-* and *ee-* (Conditions III and IV), because initial experiments showed that they were least represented in the glossed data and led to the majority of errors (see Section 5 and Table 7). Error analysis later shows that this preference leads to higher accuracy for Conditions III and IV.

After this step, we have a mapping of $4,642$ word orthographies from successful parses to a segmented and glossed version, one per word. This mapping is used to automatically segment and gloss these words whenever they are found in the unglossed sentences in the corpus. The result is a new corpus combined from the original sentences that were fully glossed and the previously unanalyzed sentences, for which one or more words is now analyzed and glossed. Table 4 shows the amount of the segmented and glossed data before and after applying the morphological grammar to the unglossed part of the corpus.[10]

4.5 Summary

This section has explained our methodology for distributional analysis over glossed Abui text and for automatically glossing portions of the unglossed text. For the first goal, which is a simple cooccurrence problem, we use a module of the MOM system which, after minor modifications, outputs verb classes as defined in §2. For the second objective, we use the full functionality of the MOM system as well as other tools leveraged in the AGGREGATION pipeline, including the Grammar Matrix customization system and the ACE parser. In the next section, we quantitatively compare the results of this analysis based on attested forms with a data set based on elicited judgments and summarize an Abui expert's qualitative opinion about the results.

5 Results

5.1 Comparison with the curated set

We now turn to a comparison of our summary of the attested data with a curated set of elicited judgments.[11] This curated set contains 337 verbal stems and documents their compatibility with Conditions I-IV discussed in §2, according to native speaker intuitions. Both the curated data set and our automatic output are represented as illustrated in Table 3, where each row represents a verb stem as belonging to a particular class.

To quantify the relationship between these two sources of information, we set the curated data as the target of comparison and then compute precision and recall per class for the automatic data.[12]

Only 12 of the 15 possible verb classes are found in the curated data set, but the system finds all 15 in the corpus. Thus the automatic processing of the forms attested in the corpus hypothesizes both additional verbs for known classes and additional classes.

It is important to emphasize that the curated data set that we compare our extracted data to is a work in progress, and the elicitation mostly relies on a small number of speakers. Even with further work, this curated data set will undoubtedly continue to contain gaps and possible mistakes.[13] Conversely, the corpus alone will probably never be enough: gaps in the corpus data can either be accidental or a result of grammatical incompatibility. This is the familiar tension between corpus and intuition data (see, for example, Fillmore (1992) and Baldwin et al. (2005)). Thus mismatches between the curated and automatically derived data sets do not necessarily indicate system errors (though we have performed error analysis in order to try to identify any). Instead,

[10] At this point, we do not know how accurate the morphological analyzer is; we assess the result by how useful the automatically glossed data ends up being for expert analysis. Evaluating the morphological analyzer by cross-validation will be part of future work.

[11] Code, data and instructions for reproducing the results can be found at `http://depts.washington.edu/uwcl/aggregation/ComputEL-2-Abui.html`

[12] Precision (P) and recall (R) are traditionally defined in terms of different types of mistakes that the system makes. For this paper, we define P and R with respect to the curated set as follows, per class. Let V_1 be the number of verbs in class A in the curated set classified as A by the system, e.g. a verb that was classified as *1111* by the system is also present as *1111* in the curated set. Let V_2 be the number of verbs in class A in system output which belong to a different class in the curated set, e.g. the verb that is *1111* in the system output is actually *1011* in the curated set. Then precision for class A is $P = \frac{V_1}{V_1+V_2}$. Let V_3 be the number of verbs in a class in the curated set which were not put in this class by the system, e.g. the verb is in the curated set as *1111*, but the system put it in *1011*. Then recall $R = \frac{V_1}{V_1+V_3}$. Precision and recall tend to compete, and F1-score is a harmonic mean of the two; the higher the F1-score, the better the system fares in both precision and recall. $F1 = 2 \cdot \frac{precision \cdot recall}{precision+recall}$.

[13] For example, a word form presented to a speaker in isolation or in an invented sentence might sound bad, while being perfectly fine in a naturally occurring context.

Code	IGT collection	IGT	IGT with verbs	Total word tokens	Total word types	Verb tokens	Verb types
GL	*Manually glossed*	4,654	2,120	33,293	4,487	8,609	2,712
AG	*Autoglossed*	13,316	11,538*	130,218	12,034	96,876*	4,642*
CB	*Combined*	17,970	13,406	163,511	13,948	105,485	5,939

Table 4: The corpus statistics. GL and AG refer to the glossed and unglossed portions of the corpus. The asterisks in the AG row indicate numbers that are based on the output of the morphological analyzer. The CB row shows the union of the GL and AG portions.

Class	Example	C	GL (in C)	CB (in C)
0001	*tatuk* 'have fever'	0	29 (1)	11 (0)
0010	*tahai* 'look for'	0	60 (4)	24 (1)
0011	*luk* 'bend'	25	4 (2)	4 (2)
0100	*buk* 'tie'	9	112 (10)	50 (4)
0101	*tok* 'drop'	0	3 (1)	8 (1)
0110	*weel* 'bathe'	1	2 (1)	6 (0)
0111	*bel* 'pull out'	10	1 (1)	1 (0)
1000	*king* 'long'	4	430 (38)	247 (19)
1001	*mpang* think	1	33 (8)	35 (6)
1010	*aai* 'add'	2	69 (17)	97 (15)
1011	*kafia* 'scratch'	184	25 (12)	50 (6)
1100	*toq* 'demolish'	19	59 (13)	121 (18)
1101	*kolra* 'cheat'	2	7 (1)	35 (6)
1110	*momang* 'clean'	3	13 (2)	62 (8)
1111	*buuk* 'drink'	75	7 (2)	103 (27)
	Total	337	854 (113)	854 (113)

Table 5: Class sizes. When there is no example in the curated set (C), an example is taken from the hypothesized set output by the system.

they represent cases in which corpus analysis can further our understanding of the language at hand.

5.1.1 Class sizes

Table 5 gives an overview of the class sizes. The size of a class is the number of unique verb stems that belong to that class. For each class (the 1st column), we provide an example of a stem in that class (the 2nd column), and show the number of unique stems in that class according to (i) the curated set (the 3rd column), (ii) the manually glossed (GL) portion of the corpus (the 4th column), and (iii) the combined (CB) data set of the corpus (the 5th column). For the last two columns, the numbers in the parentheses are the number of unique stems that appear in both the corpus (belonging to that class) and the curated set (belonging to any class).

The general trend is that adding more data leads to reclassifying some verbs from classes whose bit-vector definitions contain more 0s to classes whose bit-vector names contain more 1s. This is expected: the unglossed portion of the data may contain patterns that the glossed portion does not contain.

The total number of unique stems found in the originally glossed portion of the data is 854. The intersection between this set and the curated set, however, is only 113. We do not gain any new stems by adding previously unglossed data, since the ACE parser will not presently analyze a word with an unknown stem.[14]

The per class intersection between the corpus derived set and the curated set is often very small; only a few classes defined by the system have more than a few stems which also belong to any of the curated set classes. This means we can only compare the curated set to the system output with respect to a few verbs. Nonetheless, we report the results in §5.1.2 below.

5.1.2 Comparing class labels of stems

For the 113 stems appearing in both the curated set and the corpus, we compare their class labels in the two data sets. In Table 6, we pretend that the curated table is the gold standard, and report precision (P) and recall (R) of the system output. The system output is either based on the glossed portion only (GL) or the combined corpus (CB). The three rows with dash only correspond to the three classes which have zero members in the curated set in Table 5. In other rows, the precision or recall is zero when there is no match between system output and the curated set,[15] which is not surprising given that many verb classes contain only very few stems.

At a general level, Table 6 shows that adding more data by glossing it automatically helps discover more patterns. Specifically, the system now puts at least one verb into class *1001* that exactly matches one of the verbs in that class in the curated set. Since class *1001* contains only 2 verbs in the curated set but the corpus contains over 30, it is possible that further inspecting the system output can contribute to a fuller description of this class.

[14] See note 9.

[15] For the clarity of presentation, we define F1 score to be zero when P and R are equal to zero.

	Precision		Recall		F1 score	
Class	GL	CB	GL	CB	GL	CB
0001	-	-	-	-	-	-
0010	-	-	-	-	-	-
0011	0	0	0	0	0	0
0100	0	0	0	0	0	0
0101	-	-	-	-	-	-
0110	0	0	0	0	0	0
0111	0	0	0	0	0	0
1000	0	0	0	0	0	0
1001	0	**0.17**	0	**0.50**	0	**0.25**
1010	0	0	0	0	0	0
1011	0.58	**1.0**	0.13	0.12	0.21	**0.22**
1100	0.23	0.22	0.50	**0.67**	0.32	**0.33**
1101	0	0	0	0	0	0
1110	0	0	0	0	0	0
1111	0.50	0.35	0.04	**0.35**	0.07	**0.35**
micro-avg	0.10	0.19	0.10	0.19	0.10	0.19

Table 6: Precision, Recall, and F1-scores when comparing verb classes in the curated set and in the corpus (GL is for the glossed portion, CB is for the combined corpus). When the CB result is higher than the GL one, the former is in bold.

5.1.3 Comparing (stem, condition) pairs

In addition to comparing the class labels of stems, we can compare the (stem, condition) pairs in the curated set and in the corpus. The results are in Table 7. There are 113 stems that appear in both sets and four conditions; therefore, there are 452 possible (stem, condition) pairs. If a (stem, condition) pair appears in both data sets, it is considered a match. There are two types of mismatches: a pair appears in the curated set but not in the system output (type 1) or vice versa (type 2).

Condition	Match	Mismatch	
		Type 1	Type 2
Curated v. GL			
I	72	20	21
II	90	17	6
III	50	61	2
IV	36	75	2
Total	248	173	31
Curated v. CB			
I	**84**	8	21
II	83	4	26
III	**66**	43	4
IV	**54**	56	3
Total	**287**	111	54

Table 7: Numbers of matches and mismatches when comparing 452 (stem, condition) pairs in the curated and in the corpus (GL for glossed portion, and CB for the combined corpus). When the number of matches in CB is higher than in GL, we put it in bold.

Table 7 shows higher proportions of matches (e.g., 285 out of 452 pairs in CB) than what is shown in Table 6 because it is making more fine-grained distinctions. For instance, if a verb belongs to class *1100* in the curated set and the system puts it in class *1110*, such a verb will get zero points towards precision and recall in Table 6, but it will get three matches (while getting one mismatch) in Table 7. Where comparison at the class level directly targets our research question, comparison at the (stem, condition)-level offers insights into which prefixes are more consistently glossed in the corpus, and conversely, which may need cleanup.

5.1.4 Mismatch Analysis

Our results show relatively low agreement between the curated data set and the automated distributional analysis, which is reflected by low F1-scores in Table 6. There are several sources for this disagreement: gaps in the collected corpus, gaps in the manually constructed analysis, and system-related issues.

Gaps in the collected corpus As discussed in §5.1.3, there are two types of mismatches between the curated set and the system output: type 1 (occurring in the curated set but not in the corpus) and type 2 (the opposite). Gaps in the collected corpus is what causes the type 1 mismatch. It is not a very informative kind of mismatch, because when something is not attested in a relatively small field corpus,[16] it does not mean it is not possible in the language. For this type of mismatch, the most likely explanation is that an example which would account for it has not yet been added to the corpus.

Gaps in the curated set In contrast, the type 2 mismatch means that there are examples in the corpus that indicate the combination is possible, but the curated set states otherwise. In this case, either the curated set needs to be refined or there are errors in the IGT corpus.[17] This is the more interesting type of mismatch. The counts in Table 7 show that, after adding more data, the type 2 count increases. Specifically, it gives rise to more mismatches with respect to Condition II (the PAT prefix). Most of these counterexamples are in fact spurious and are discussed below with respect to system-related issues, but there is at least one genuine discovery. The verb *kaang* ('to be good') was

[16] As opposed to huge raw-text corpora available for some high resource languages.

[17] Barring bugs in our system.

deemed to not be able to combine with the PAT prefix according to the curated set, but this type 2 mismatch led us to confirm that the (stem, condition) pair is in fact possible (with the meaning 'recover from disease'). Further analyzing this type of mismatches may lead to more discoveries.

System-related issues There are two main sources of noise in our system output: the morphological analyzer (i) mislabels nouns as verbs, and (ii) does not normalize words with respect to phonological variation.

Possessive prefixes in Abui often look like the PAT prefixes, and lexical categories are fairly fluid. Thus some word tokens that are automatically analyzed as verbs by the grammar might actually be nouns in context (and thus more appropriately analyzed as representing other sets of affixes). This affects the precision scores for classes *1100* and *1111*; analysis on the stem-prefix level (Table 7) shows that it is indeed the PAT prefixes that are "to blame" here, as the number of mismatches with respect to other conditions lowers when automatically glossed data is added.

Finally, there are stems from the curated set that are attested in the corpus but which the system was unable to find. This is most often due to lack of phonological normalization in the corpus. This problem is more prominent in the unanalyzed part of the data; by virtue of it having not been analyzed, there is no normalization with respect to phonological variation of forms such as 'tell', written as *anangra* and *ananra*. This indicates an important direction for future work: in addition to segmenting and autoglossing, it will be valuable to train a phonological analyzer which would map stems to a single canonical representation.

5.2 Expert analysis

The second author, who is an expert in Abui, reviewed the verb classes output by our system from the combined dataset, e.g. the 11 verbs in class *0001*, the 24 verbs in class *0010*, etc. In general, the classes were found to contain noise, the sources for which include homophones, mistakes in the corpus,[18] and lack of phonological normalization. At the same time, *0001* appears to be a potentially true–previously unknown–class of verbs whose semantics may have something in common. Many of these verbs usually occur with an experiencer argument. Classes *0100* and *1011* were already known, but the system output helped find more verbs which truly belong to them. These refinements to the classes will help inform linguistic hypotheses about the inflection system as a whole and its interaction with the lexical meaning of the verbal stem. Further analysis of the results will provide a methodology for significantly improving and extending the treatment in Kratochvíl (2007).

6 Conclusion and Future Work

We performed a computational overview of a corpus to hypothesize inflectional verb classes in Abui. As a part of this process, we used precision grammar machinery to automatically gloss part of the previously unanalyzed corpus of Abui and thus obtained more data.

We compared two different types of analyses–manual, based on elicitation, and automatic, produced by our system–and found that the mismatches between the two, especially the type where a pattern is found in the corpus but not in the elicited set, help refine the understanding of the classes.

For future work, we can add a phonological analyzer to the automatic glossing procedure and refine the parse ranking for the automatic glossing. In addition, adding unknown stem handling to the morphological grammar may help further refine the understanding of the patterns of verb-prefix cooccurrence. Finally, the methods which we used here can be extended to perform a computational overview of the Abui verbs with respect to not only first, but also second and third position prefixes. While looking at verbs with first position prefixes only required no more than a simple cooccurrence table, looking at prefixes across all three positions increases the complexity of the problem and further highlights the value of being able to automatically derive and apply a full-scale morphological grammar to the task.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. BCS-1561833 (PI Bender) and by the Singapore Ministry of Education under Tier 2 Grant MOE2013-T2-1-016 (PI Kratochvíl).

[18] The identification of these mistakes is also useful.

References

Timothy Baldwin, John Beavers, Emily M. Bender, Dan Flickinger, Ara Kim, and Stephan Oepen. 2005. Beauty and the beast: What running a broad-coverage precision grammar over the BNC taught us about the grammar — and the corpus. In Stephan Kepser and Marga Reis, editors, *Linguistic Evidence: Empirical, Theoretical, and Computational Perspectives*, pages 49–69. Mouton de Gruyter, Berlin.

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.

Emily M. Bender, Scott Drellishak, Antske Fokkens, Laurie Poulson, and Safiyyah Saleem. 2010. Grammar Customization. *Research on Language & Computation*, pages 1–50. 10.1007/s11168-010-9070-1.

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, Joshua Crowgey, Michael Wayne Goodman, and Fei Xia. 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. Association for Computational Linguistics.

Berthold Crysmann and Olivier Bonami. 2012. Establishing order in type-based realisational morphology. In *Proceedings of HPSG*, pages 123–143.

Berthold Crysmann and Woodley Packard. 2012. Towards Efficient HPSG Generation for German, a Non-Configurational Language. In *COLING*, pages 695–710.

Sebastian Fedden, Dunstan Brown, František Kratochvíl, Laura C Robinson, and Antoinette Schapper. 2014. Variation in Pronominal Indexing: Lexical Stipulation vs. Referential Properties in Alor-Pantar Languages. *Studies in Language*, 38(1):44–79.

Charles J. Fillmore. 1992. “Corpus linguistics” or “computer-aided armchair linguistics”. In Jan Svartvik, editor, *Directions in Corpus Linguistics: Proceedings of Nobel Symposium 82, Stockholm, 4-8 August, 1991*, pages 35–60. Mouton de Gruyter, Berlin, Germany.

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

Michael Wayne Goodman. 2013. Generation of machine-readable morphological rules with human readable input. *UW Working Papers in Linguistics*, 30.

František Kratochvíl and Benidiktus Delpada. 2015. Degrees of affectedness and verbal prefixation in Abui (Papuan). In Stefan Müller, editor, *Proceedings of the 22nd International Conference on Head-Driven Phrase Structure Grammar, Nanyang Technological University (NTU), Singapore*, pages 216–233, Stanford, CA. CSLI Publications.

František Kratochvíl and Benidiktus Delpada. 2017. Abui Inflectional Paradigms. Electronic Database. Nanyang Technological University, February.

František Kratochvíl. 2007. *A grammar of Abui: a Papuan language of Alor*. LOT, Utrecht.

František Kratochvíl. 2011. Transitivity in Abui. *Studies in Language*, 35(3):588–635.

František Kratochvíl. 2014. Differential argument realization in Abui. *Linguistics*, 52(2):543–602.

František Kratochvíl. 2017. Abui Corpus. Electronic Database: 162,000 words of natural speech, and 37,500 words of elicited material (February 2017). Nanyang Technological University, Singapore.

William D. Lewis and Fei Xia. 2008. Automatically identifying computationally relevant typological features. In *Proceedings of the Third International Joint Conference on Natural Language Processing*, pages 685–690, Hyderabad, India.

Kelly O'Hara. 2008. A morphotactic infrastructure for a grammar customization system. Master's thesis, University of Washington.

Anna Siewierska. 2013. Verbal person marking. In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

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

SIL International. 2017. Sil Fieldworks. Lexicon and corpus management system with a parser and concordancer, URL: http://software.sil.org/fieldworks/.

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

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, Berlin, Germany, August. Association for Computational Linguistics.