Towards a robust morphological analyser for Kunwinjku

Lane, William; Bird, Steven

Published in:
Proceedings of the 17th Annual Workshop of the Australasian Language Technology Association

Published: 01/01/2019

Document Version
Publisher's PDF, also known as Version of record

Towards A Robust Morphological Analyzer for Kunwinjku

William Lane and Steven Bird
Charles Darwin University
The Northern Institute

Abstract

Kunwinjku is a polysynthetic language spoken in northern Australia. Members of the community have expressed interest in co-developing language applications which could assist in the production of written language resources for education and language learning. Modelling Kunwinjku morphology is a step towards accomplishing these goals. We discuss some of the modeling challenges presented by Kunwinjku verbal morphology, and in polysynthetic languages more generally. We show that a model using standard features of the *Foma* toolkit can account for much of the verb structure. Our contributions include the first morphological analyzer for Kunwinjku, and a discussion of polysynthetic language features and how they affect modelling decisions. Continuing challenges include robustness in the face of variation and unseen vocabulary, as well as how to handle complex reduplicative processes.

1 Introduction

Kunwinjku is an Aboriginal language of the Gunwinyguan language family (ISO gup), spoken by about 2000 speakers in the West Arnhem region of northern Australia. Several Kunwinjku communities have shown interest in leveraging technology to support the production of literacy materials and language learning applications (Bird, 2018).

A major focus of our research group is to implement language technologies that have positive social impact, such as a morphologically-aware dictionary which lowers the barrier to entry for users who cannot reliably identify or spell citation forms (Hunt et al., 2019; Arppe et al., 2016), or a tool that generates linguistic structures which could help language learners master conjugation and verb structure (Kazantseva et al., 2018).

One thing that these applications have in common is the need to decompose and manipulate text at the level of morphology. In order to accomplish this, we must address polysynthesis, morphophonemic alternations, incorporation, reduplication, and long-distance dependencies. Which aspects of morphosyntax can we model? What are the limitations of computational approaches for modeling polysynthetic languages more generally?

In the sections that follow, we will first give an overview of those features of the language which affect how we approach the modelling task (sec 2). Next, we introduce our data sources and the metrics we use to evaluate performance (sec 3). This is followed in section 4 by a detailed description of our implementation and how we addressed the linguistic features described in section 2. Finally, we report accuracy and coverage on both a development data set and a blind test set, provide an error analysis and discussion, and conclude with some thoughts on future directions. To our knowledge, this is the first morphological analyzer for Kunwinjku.

2 Features of Kunwinjku Verbs

We model and evaluate the morphosyntax of Kunwinjku verbs according to Evans' *Pan-dialectal Grammar* (Evans, 2003). In this section we describe some of these features, and follow-up later with how we account for them in the model.

2.1 Polysynthesis and Agglutination

Kunwinjku is a polysynthetic language, with verb roots having 12 prefix slots including the subject/object/tense pronominal, directional, benefactive, incorporated nominals, and comitative affixes (Figure 1). There are 3 suffix slots for indicating reflexivity, tense/aspect/mood, and case. (In limited cases, embedding one verb in another is allowed between the -1 and 0 slots).

−12	−11	−10	(−9)	(−8)	(−7)	(−6)	(−5)	(−4)	(−3)	(−2)	(−1)	0	+1	+2
Tense	Subject	Object	Directional	Aspect	Misc1	Benefactive	Misc2	Gen.inc.nom	Bod.par.inc.nom	NumeroSpacial	Comitative	Verb Stem	RR	TAM

Figure 1: Verbal affix positions in Kunwinjku. Regions where indices share a cell ([−12,−10], [+1,+2]) indicate potentially fused segments. Slot indices in parentheses indicate optionality. Adapted from (Evans, 2003, Fig 8.1).

The morphology is described as agglutinative, almost "lego-like" (Evans, 2003; Baker and Harvey, 2003), though with some unusual morphophonemic alternations involving glottal stop, long distance dissimilation of peripheral nasals, and complex types of reduplication. Additional complexity can be found in the peripheral "fusion zones" spanning slots [−12, −10] and [0, +2].

2.2 Noun Incorporation

Figure 1 shows the optional slots −4 and −3, labeled for the general incorporated nominal (GIN), and the body part incorporated nominal (BPIN), respectively. The GIN class represents a closed set of nouns which, after losing their gender/class and case inflections, can be injected into the verb to satisfy valency. Consider the incorporable noun *kunrerrng* "wood", and the phrase *karrimang* "we would go get." To form the phrase "we would go get wood" using noun incorporation, *kunrerrng* loses its noun class inflection *kun-*, and is placed in slot −4:

(1) a. karri-ma-ng kun-rerrng
1pl-get-past IV-wood

b. karri-rerrng-ma-ng
1pl-GIN.wood-get-past
'We would go and get wood' [E.145]

Nouns from the BPIN class perform a similar function, with the characteristic difference of being loosely associated with the human corporal form (an arm, a leg, your shadow, etc). Here we see the noun *kunkanj* "meat," loses its noun class inflection *kun-* and is placed in the verb slot at −3:

(2) a. bi-ngu-neng kun-kanj
3sg.3Hsg.past-eat-past IV-meat

b. bi-kanj-ngu-neng
3sg.3Hsg.past-BPIN.meat-eat-past
'He is eating meat' [E.687]

Additionally, BPIN is an *open* class.

2.3 Valency-affecting prefixes

As can be seen in Figure 1, Kunwinjku allows for 15 morph slots to complete a verb form. Transitivity of the verb is lexically defined, but there are three morph slots which signal valency change and affect the resulting semantic interpretation: the benefactive (BEN), comitative (COM), and reflexive (RR) (Evans, 2003; Ponsonnet, in press).

The following subsections describe the morphemes which affect the valency of the verb.

2.3.1 Benefactive marne-

The benefactive prefix indicates that one of the verb objects is the beneficiary of the action of the verb. For example, the English verb for *say* in Kunwinjku is translated as *yime*, and is designated by the grammar as intransitive. Consider the case where *yime* is paired with the benefactive *marne-* prefix:

(3) ben-marne-yime-ng
3sg.3pl.past-BEN-say-PastPerf
'He told them' [E.637]

We see that this prefix opens up the intransitive verb, in this case *yime*, to the possibility of taking on the 3rd person plural object. That object can be present in the verb itself via the pronominal or an incorporated noun, or it could be located outside of the verb entirely.

2.3.2 Comitative yi-

The comitative slot is located at position −1. Its presence extends verb valency by 1. If the verb root is intransitive, the COM indicates that the additional object is "with" (accompaniment, not instrumental) the subject of the verb. For example:

(4) ben-yi-yibme-ng
3sg.3pl-COM-sink-PastPerf
'He took them down under the water' [E.433]

If the verb root is transitive, then it conveys the meaning that the new argument accompanies the object of the transitive verb. For example:

(5) nga-kole-yi-kurrme-ng
1sg.3sg-GIN.spear-COM-put.down-PastPerf
'I left the spear with him' [E.433]

2.3.3 Reflexive and Reciprocal -rre

Reflexivity and reciprocity are expressed using the morph *-rre* in slot +1. In either case, the result is that the valency of the verb is reduced by 1.

(6) bene-marne-kinjwe-rre-nj
3ua.3sg-BEN-be.jealous-RR-PastPerf
'They were jealous of each other over him.' [E.430]

In this example, *bene-* is the 3rd person dual subject (those two) with a 3rd person singular object. The reflexivity occurring after the verb root directs the action of *being jealous* back onto the subject, with the indirect object (the 3sg "him") remaining unaffected.

2.4 Morphophonemic Considerations

Where morphs combine, there are a few morphophonemic patterns to account for. The most widespread is that of *d-flapping*, where morpheme-initial *d* becomes *rr* after vowel-final syllables. For example, the inflected form *nga-rranginj* has the verb root *dangen*, but we see the *d* has been changed to *rr* because it is preceded by the syllable *nga* which contains a syllable-final vowel. While this rule is fairly regular, Evans' grammar also recognized cases where the pattern doesn't seem to apply and concludes that "a fuller understanding of stress and prosody will be needed before such examples can be accounted for". Take the verb "dirri", "to play" for example:

(7) a. * nga-rrirri-∅
b. * nga-rridi-∅
c. nga-dirri-∅
1sg-play-nonpast
'I play'

Another morphophonemic pattern is the deletion of morpheme-initial *r* following apical consonants *rr*, *l*, and *n*. In careful speech and written Kunwinjku, this pattern is not always obligatory; Evans argues that it is not evident whether these changes should be treated as "fast-speech phenomena" and therefore not shown in the orthography. The most consistent example of this alternation that we have seen is that of $r \rightarrow \emptyset || rr\_$, which manifests itself in the example of *ngarr-re* which becomes *ngarre*, *we two go*.

There are other morphophonemic changes that occur in Kunwinjku speech, but which do not appear to be reflected in the accepted orthography. Evans posits that since the Kunwinjku dialect has a longer tradition of literacy (relative to other dialects), these changes are not usually reflected in the written medium. Some of the phenomena that fall into this camp are specific cases of *nasal assimilation*, and *peripheral dissimilation*. Since the goal of our morphological analyzer is to recognize the inflected forms of written verbs, we avoid giving a more complete description of morphophonemic processes which do not impact the standard written form. It is important to note however that these processes may cause variation in how speakers of the language write. A truly robust analyzer intended for applications like spell-checking would need to consider such processes as they manifest themselves in human input.

2.4.1 Reduplication

Kunwinjku has three main types of partial verbal reduplication signalling iterative, inceptive, and extended meaning. Moreover, each type of reduplication can have more than one consonant (C) and vowel (V) reduplicative pattern, depending on which of the 11 verb form paradigms the verb belongs to. See Figure 2 for details.

Computational modeling of partial reduplication in human language using finite state transducers (FSTs) has been addressed in the past (Culy, 1985; Roark et al., 2007; Dras et al., 2012), with the general consensus being that these kinds of partially reduplicative processes explode the state space of the model, and are therefore highly burdensome to develop. More recent work addresses these challenges using 2-way FSTs (Dolatian and Heinz, 2018, 2019), and offers a promising future avenue of exploration for our work with Kunwinjku. We include reduplication in this paper for the sake of completeness (see Figure 2), but acknowledge that a solution lies beyond the scope of this work.

3 Data and Metrics

As mentioned previously, the grammar implementation is based on (Evans, 2003). The lexicon was subsequently expanded using the resources curated at kunwok.org, a website dedicated to open sharing of content and teaching the Kunwinjku language (Bird and Marley, 2019), as well as the verbs from the online Kunwinjku dictionary at njamed.com (Garde et al., 2019). In terms of written or digital language materials Kunwinjku is firmly in the low-resource camp, though we are in the favorable position of being supported by motivated native speakers who work with us to clarify questions about language data.

Type of reduplication	Pattern(s)	Unreduplicated Verb	Reduplicated Verb	Semantic Effect on the verb (V)
Iterative	CVC	dadjke = cut	dadj-dadjke = cut to pieces	Doing V over and over again
	CV(C)CV(h)	bongu = drink	bongu-bongu = keep drinking	
	CVnV(h)	re = go	rengeh-re = go repeatedly	
Inceptive	CV(n)(h)	yame = spear (something)	yah-yame = try (and fail) to spear (something)	Failed attempt to do V
		durnde = return	durnh-durnde = start returning	Starting to do V
Extended	CVC(C) ‖ _ men	djordmen = grow	djordoh-djordmen = grow all over the place	Doing V all over the place
	CVC(C) ‖ _ me	wirrkme = scratch	wirri-wirrkme = scratch all over	

Figure 2: Reduplication in Kunwinjku has three forms, and each form has its own patterns defining how much of the verb is captured and copied. In the case where we've used the form X ‖ _ Y, we mean that pattern X is the reduplicated segment if found in the context of Y. Figure adapted from (Evans, 2003).

To construct our development corpus of inflected verbs, we extracted all of the Kunwinjku examples from the reference grammar; a total of 567 glossed verbs. We further refined the list to exclude cases of reduplication (cf 2.3.4) which left us with 530 verbs which we used to produce a data set to support the development of the FST.

Additionally, we glossed a small set of 114 verbs randomly sampled from the Kunwinjku translation of the Bible, for the purpose of judging how well the FST generalizes to another domain. The Bible translation was recently completed in 2018, and targets the modern vernacular.

We use accuracy and coverage to measure the effectiveness of the model on the development data set as well as the test set.

4 Implementation

Finite state transducers are viewed as an ideal framework to model morphology (Beesley and Karttunen, 2003; Chen and Schwartz, 2018; Lachler et al., 2018). Our FST was implemented using the *Foma* toolkit (Hulden, 2009) which is a popular framework for building morphological analyzers for polysynthetic languages (Chen and Schwartz, 2018; Moeller et al., 2018; Littell, 2018). The definition of an FST in *Foma* is comprised of a lexicon implemented in the *.lexc* format, and a *.foma* file for defining rules covering regular morphophonemic changes. The final FST is produced by composing the FSTs defined in both files.

4.1 The .lexc file

The *.lexc* file contains definitions of lexicon groups corresponding to morphological units of the language. Lexical entries of the group are listed below the group definition. Each entry in the lexicon is paired with its *continuation class* which defines legal paths through the FST, enforcing valid sequences of morphs. Figure 3 gives a stripped-down example of this by implementing a *.lexc* file capable of mapping the inflected Kunwinjku verb *ngakanjngun*, "I am eating meat", to its analysis: *1sg.nonpast-GIN.meat-eat-nonpast*.

```
LEXICON TSOPreBase
[V][1sg.nonpast]:nga GINPreBase;

LEXICON GINPreBase
[GIN]:0 IncNounBase ;
0 PostNominal ;

LEXICON IncNounBase
0:kanj PostNominal;

LEXICON PostNominal
@R.TYPE.VERB@ IntransVerbs ;

LEXICON IntransVerbs
ngu V3IrrPostBase;

LEXICON V3IrrPostBase
[NonPst]:n #;
```

Figure 3: Lexicon groups are defined in the *.lexc* file using the LEXICON keyword. Valid paths through lexicons are defined on an entry-by-entry basis. Here each lexicon only has one entry, and there is only one path through the graph. The accept state in the graph is signaled by the # character.

In general, slot positions in the grammar mapping to lexicons in the implementation have a one-to-many relationship, that is, one slot can be satisfied by an entry from one of many lexicon groups. In the example of Figure 3, we show only 4 lexicons filling 4 of the available 16 positions: *TSOPrebase* corresponding to the entry which fuses the morph positions spanning indices $[-12,-10]$, *GINPrebase* corresponding to the morph position at index -4, *IntransVerbs* corresponding to the verb root at index 0, and *V3IrrPostBase* corresponding to the suffix at index +2. Our complete implementation contains 63 lexicons, each of which map to one of the 16 slots defined in the grammar.

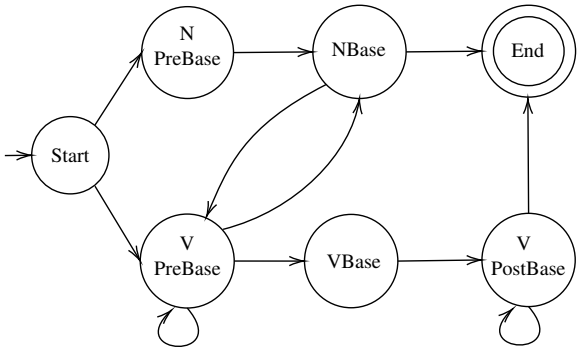

Figure 4: A high-level overview of the morphological analyzer. Verbal prefix lexicons are represented by *VPreBase*, verb root lexicons are represented by *VBase*, and verbal suffixes are represented by *VPostBase*.

4.1.1 Noun Incorporation

We handle noun incorporation similarly to Chen and Schwartz (2018): we allow transitions from states in the verbal pre-base lexicons (specifically, the GIN and BPIN slots) to the noun base lexicon (see Figure 4). We use flag diacritics to enforce constraints on word type: if the word begins with verbal morphology and crosses into the noun base via incorporation, it can only be recognized as a valid string if it also ends with verbal post-base morphology.

As described in Figure 3, we define *GINPreBase* as a lexicon which points to the IncNounBase class, which enumerates the closed class of incorporable nouns. Similarly, we define a lexicon named *BPINPreBase* representing the open class of body part nouns which can be incorporated, and which fill the optional position at index -3.

4.1.2 Tense Agreement

The inflected Kunwinjku verb requires agreement between tense in the pronominal prefix and the TAM suffix. To address this we need to discriminate between ambiguous paths in the FST based on a feature value that is set at some node in the graph and persists to enforce agreement downstream. In Foma, we use flag diacritics to implement this. Flag diacritics take the form of *@FLAGTYPE.FEATURE.VALUE@* where *FLAGTYPE* defines the behavior of the flag, selected from the set of predefined flag types, and *FEATURE* and *VALUE* must be defined by the user (Hulden, 2011). We make use of flag types *P* and *R* in our implementation of tense agreement. *P* defines the action of setting *FEATURE* to *VALUE* and *R* defines the action of requiring that *FEATURE* equal *VALUE* in order to remain a valid path.

We define the following tags to enforce tense agreement:

1. @P.TENSE.PAST@
2. @P.TENSE.NONPAST@
3. @R.TENSE.PAST@
4. @R.TENSE.NONPAST@

To see how this works in the *.lexc* format, we update the example in Figure 3 to reflect the enforcement of tense agreement between the pronominal and the tense inflection on the verb using flag diacritics (Figure 5). Notice how *@P.TENSE.PAST@* is present before the continuation class transitions in the new TSOPreBase lexicon: this effectively labels all paths proceeding from this point as having the attribute *TENSE* set to a value of *PAST*, indicating that this path will agree with any attempt to enforce agreement with a past tense. Indeed, at the bottom of Figure 5 we see the *@R.TENSE.PAST@* diacritic on both the up and down side of the transducer, indicating that in either direction[1] if we match the morphological entry, the path we've taken must also match the *TENSE* flag feature value in order for the analysis to be valid.

4.1.3 Valency Agreement

As described in 2.3, the valency of the verb is affected by the presence of certain prefixes: the benefactive *marne-*, the comitative *yi-* and the reflexive *rre-*. Our initial belief was that in order to analyze an inflected verb in Kunwinjku, it would be necessary to model these valency changes. We saw no reason to allow the FST to generate analyses which seemed to demonstrate valency imbalance in either the direction of over-saturation or under-saturation. For example, the following verbs seem to provide too many or too few arguments:

(8) bi-marne-bong-yo-y
3sg.3Hsg.past-BEN-GIN.string-lie-PP
+2 −1 +1 −1 0
'He had the string lying there for her' [E.429]

[1] If you label a continuation class with a flag diacritic, it has the same effect as if you explicitly label both sides of the FST up/down transitions inside that continuation class with a diacritic flag.

```
LEXICON Root
@P.TYPE.VERB@ TSOPreBase ;

LEXICON TSOPreBase
@P.TENSE.PAST@ SingleIntransPastTSO;

LEXICON SingleIntransPastTSO
[V][1sg.past]:nga GINPreBase;

LEXICON GINPreBase
[GIN]:0 IncNounBase ;
0 PostNominal ;

LEXICON IncNounBase
0:kanj PostNominal;

LEXICON PostNominal
@R.TYPE.VERB@ IntransVerbs ;
@R.TYPE.NOUN@ # ;

LEXICON IntransitiveVerbs
ngu V3IrrPostBase;

LEXICON V3IrrPostBase
@R.TENSE.PAST@[NonPst]:@R.TENSE.PAST@n #;
```

Figure 5: Updating our *.lexc* file to constrain possible paths through the FST based on the value of the TENSE feature. Shown also is a diacritic enforcement of word type in the PostNominal lexicon.

(9) ∅-djare-ni
3sg.past-want-PastImperf
+1 −2 0
'He was wanting _' [E.229]

However, both of the above examples are valid. In these glossed examples, we've added a row under the morpheme analysis to mark which morphemes satisfy valency (+1 if it represents 1 object, +2 if it represents 2 objects), morphs which increase the valency of the verb (−1 if it demands 1 object, etc), and morphs which have no effect on valency (0). This allows us to do the simple arithmetic to convince ourselves that (9) is over-saturated, while (10) is under-saturated.

In (9), we have a subject ("he") and two object candidates ("her" and "the string"). The verb root "yo" ("to lie") is inherently intransitive, and the presence of *marne-* opens up room for the verb to take an additional argument as the benefactor: "her". In this instance, the verb appears to be saturated, leaving "the string" to act rather as specification of the verb "yo". As this case shows, ambiguity around when an incorporated nominal is acting as an object rather than providing referential specification impedes the attempt to model valency based on surface form morphology alone.

In (10), the object of the transitive verb for *want* is not incorporated, but exists in the wider sentential context. This implies that valency cannot be disambiguated without reference to syntactic context. Indeed, a syntactic concept can be expected to have syntactic scope, and it is not uncommon for languages containing valence-altering morphs to provide valence-satisfying objects outside of the verb (Haspelmath and Müller-Bardey, 2004). In light of this, we decided to take a permissive stance, allowing valence-imbalanced analyses at the level of individual verbs.

4.2 The .Foma File

The *.Foma* file is the place to encode morphophonological rules in the form of $A \rightarrow B \| \Gamma \_ \Delta$, ie "A changes to B in the context $\Gamma \_ \Delta$", where A and B are orthographic symbols representing phonemes which alternate in the given context. These rules define an FST which can then be composed with the lexicon FST defined in the *.lexc* file, resulting in a final FST representing the complete grammar. Since Kunwinjku is largely agglutinative, with relatively little morphophonemic change to account for. This makes the content of our *.Foma* file a relatively simple composition of three parts: A list of special symbols we define to make our rules more compact, an enumeration of allophonic rules, and finally the composition of the lexicon with the rules to produce the final grammar.

In Figure 6 we give a Foma file that maps the intermediate from *karriˆbimˆbuˆ~om* to the correct final surface form, *karribimbom*. First, the *DeletePrecedingVowel* rule is activated by the observation of a vowel "u" followed by the morpheme boundary marker "ˆ" and the "~", which is an arbitrary symbol we encode in the lexicon to indicate that the TAM inflection *om* tends to override any final vowel in the preceding morpheme. The context is recognized, and the vowel is deleted (changes to 0) followed by deletion of the "~" itself. Application of this first rule now yields *karriˆbimˆbˆom*. But we aren't done yet: the cleanup step occurs with the *CleanMorphBoundaries* rule, which recognizes the "ˆ" symbol in any context, and deletes it. We now have the final form *karribimbom*; "we painted".

```
read lexc kunwok.lexc
define Lexicon;

define V [ a | e | i | o | u ];

define CleanMorphBoundaries "^" -> 0;

define DeletePrecedingV V -> 0 || _ "^" "~" .o.
                          "~" -> 0 ;
.
.
.

define FlapChange "(rr)" -> r r || V "^"_ .o.
                  "(rr)" -> d ;

define Grammar Lexicon                .o.
               DeletePrecedingV       .o.
                  .                   .o.
                  .                   .o.
                  .                   .o.
               FlapChange             .o.
               CleanMorphBoundaries;

regex Grammar;
```

Figure 6: An example of our *.Foma* file. We define phonemic rules which are applied to the *Lexicon* FST by composition, which produces a new and final FST named *Grammar*

5 Evaluation

The final FST implements the rules required to produce verbs in Kunwinjku. This includes 157 pronominal entries (including variations reflecting combinations of tense and transitivity), 23 adverbial/aspective/quantitative modifiers of the verb, 77 general incorporable nouns (a closed class), 31 body part incorporable nouns (an open class), 541 verb roots, and 124 TAM inflection possibilities. As mentioned in section 3, we extracted 530 inflected verb forms from the Evans' grammar which we used to optimize coverage and accuracy. Accuracy in this context refers to the number of correct analyses out of the set of analysis the FST attempted. We calculate it this way to avoid double-counting information already captured in the coverage metric. Those numbers are shown in Figure 8, along with the reported performance over the test set.

6 Discussion

In order to better understand the performance of the FST, we analyzed the coverage and accuracy on the Bible dataset and identified four classes of error: missing verb root, missing incorporated nominal, irregular inflection patterns, and reduplication (see Fig 7).

The most common error type is *missing verb root*, which represents 47% of errors. Similarly, *missing incorporated noun*, which accounts for another 29%, for a total of 76% of errors due to missing lexical entries. We posit that while it may be possible to infer unseen roots by recognizing the surrounding inflection and stripping it away, the presence of unseen incorporated nouns which attach directly to the verb root have the potential to complicate the matter. However, stemming like this would be sufficient for automatically discovering potential verb roots from unannotated text, which can then be verified by language experts prior to adding them to the lexicon.

Reduplication represents 18% of errors. As we discussed in 2.3.4, there are potential solutions including 2-way FSTs (Dolatian and Heinz, 2018, 2019), and the possibility that neural approaches to morphological analysis could learn to recognize reduplication through supervised learning (Micher, 2017; Moeller et al., 2018; Schwartz et al., 2019). It would be interesting to observe a similar error analysis on a much larger sample size to see if this rate of reduplicative structure holds, and to get an idea for the relative distribution of reduplicative structure in Kunwinjku.

The least common class of error contained a single instance: *Irregular inflection pattern*. Here, a path through the FST could not be found because we come across irregular variation of the TAM inflection. Whether this represents an entire class of error or is caused by simple orthographic variation is unknown: the question requires a larger sample size and consultation with language experts.

7 Conclusion

Kunwinjku is low-resource Australian language for which we would like to develop useful language learning applications. Being able to model the rich verbal morphology is an important step towards that goal. In this work, we identified several areas of Kunwinjku morphology which fit well within the framework of finite state transduction, and some for which a different approach may be better suited to the task. FSTs do well at handling the templatic structure of polysynthetic morphology. For languages which exhibit high rates of allomorphy and morphophonemic change, the ability to compose multiple FSTs into a final grammar has been shown to be quite effective (Chen and Schwartz, 2018; Littell, 2018).

The most significant shortcomings of our FST are expanding the lexicon, accounting for redupli-

	Verb Form	Meaning/Problem
Missing Verb Root - 47%	ngurrimirnde**mornname**rren	bear/place on the shoulders
	wobekkang	variation of *bekkan*; to hear about
	nga**kohbanjminj**	become an old person
	nga**rrukkendi**	variation of *dukkan*; tie up; put in handcuffs
	nga**djareni**wirrinj	variation of djare; to want
	yi**djareni**wirrinj	variation of djare; to want
	ka**menyime**	variation of menmenyime; to mean
	yiwernh**marnedjareni**n	variation of marnedjare; to love somebody
Missing Inc. Noun - 29%	yiben**kange**marnbom	heart
	kan**kange**murrngrayekwong	heart
	kan**njilng**marnbom	feelings
	yi**malng**darrkiddi	soul
	kan**kange**marnbom	heart
Reduplication - 18%	**burrbuh**burrbun	keep thinking
	djawahdjawan	keep asking; plead
	djawahdjawani	keep asking; plead
Irreg. TAM Inflection - 8%	ngayimerr**anj**	expected past perfect *-inj* TAM suffix

Figure 7: The inflected verbs from the Bible test set for which the FST had no analysis, sorted into one of the four buckets for error analysis. The **bold substrings** are the morphs which the FST could not account for.

	Coverage	Accuracy
Kunwinjku Bible-Test	85.09	97.94
Evans Grammar-Dev	97.17	95.28

Figure 8: Coverage and accuracy of the FST model of verbs in Kunwinjku. The Evans Grammar represents the data we optimised our FST against. The Kunwinjku Bible data is a blind test set.

cation, and being robust in the face of variation in form and orthography.

Additionally, we could have benefitted from a much larger annotated test set. While the Bible set was sufficient to point out the issue of lexicon coverage in our FST, more data could help solidify the relative importance of the other much smaller error classes. It could also give us more insight into the distribution of other constructions in Kunwinjku, which may inform the pedagogical aspect of designing language learning applications in a low-resource setting.

In future work we hope to expand the lexicon of this tool in parallel with developing other approaches to morphosyntactic analysis. Specifically, recent work in bootstrapping recurrent neural models using an FST to generate training examples has showed significant increase in coverage and accuracy in other polysynthetic environments (Micher, 2017; Moeller et al., 2018; Schwartz et al., 2019).

Acknowledgments

We are grateful for the support of the Warddeken Rangers of West Arnhem. We thank Maïa Ponsonnet for her valuable insights into Gunwinyguan morphology, syntax and semantics. We also thank Alexandra Marley for contributing her expertise on Kunwinjku morphosyntax. This research has been supported by grants from the Australian Research Council and the Indigenous Languages and Arts Program of the Federal Department of Communication and the Arts. Finally, we would like to thank the three anonymous reviewers for their constructive feedback.

References

Antti Arppe, Jordan Lachler, Trond Trosterud, Lene Antonsen, and Sjur N Moshagen. 2016. Basic language resource kits for endangered languages: A case study of Plains Cree. *Collaboration and Computing for Under-Resourced Languages: Towards an Alliance for Digital Language Diversity*, pages 1–8.

Brett Baker and Mark Harvey. 2003. Word Structure in Australian Languages. *Australian Journal of Linguistics*, 23:3–33.

Kenneth R Beesley and Lauri Karttunen. 2003. Finite-state morphology: Xerox tools and techniques. *CSLI, Stanford*.

Steven Bird. 2018. Designing mobile applications for

endangered languages. In *The Oxford Handbook of Endangered Languages*. Oxford University Press.

Steven Bird and Alex Marley. 2019. Kunwok.org. https://kunwok.org/. Accessed: 2019-08-30.

Emily Chen and Lane Schwartz. 2018. A morphological analyzer for St. Lawrence Island / Central Siberian Yupik. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*, Miyazaki, Japan. European Language Resources Association.

Christopher Culy. 1985. The complexity of the vocabulary of Bambara. In *The Formal Complexity of Natural Language*, pages 349–357. Springer.

Hossep Dolatian and Jeffrey Heinz. 2018. Modeling reduplication with 2-way finite-state transducers. In *Proceedings of the Fifteenth Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 66–77, Brussels, Belgium. Association for Computational Linguistics.

Hossep Dolatian and Jeffrey Heinz. 2019. Learning reduplication with 2-way finite-state transducers. In *International Conference on Grammatical Inference*, pages 67–80. Proceedings of Machine Learning Research.

Mark Dras, François Lareau, Benjamin Börschinger, Robert Dale, Yasaman Motazedi, Owen Rambow, Myfany Turpin, and Morgan Ulinski. 2012. Complex predicates in Arrernte. In *Proceedings of the LFG12 Conference*, pages 177–197. CSLI Publications.

Nicholas Evans. 2003. A Pan-dialectal Grammar of Bininj Gun-Wok (Arnhem Land): Mayali, Kunwinjku and Kune. *Canberra: Pacific Linguistics*.

Murray Garde, Jill Nganjmirra, and Dan Kennedy. 2019. Bininj Kunwok Dictionary. njamed.com. Accessed: 2019-07-19.

Martin Haspelmath and Thomas Müller-Bardey. 2004. Valency change. In Geert Booij, Christian Lehmann, and Joachim Mugdan, editors, *Morphology: A Handbook on Inflection and Word Formation*, pages 1130–45. de Gruyter Berlin; New York.

Mans Hulden. 2009. Foma: a finite-state compiler and library. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 29–32. Association for Computational Linguistics.

Mans Hulden. 2011. Morphological analysis tutorial: A self-contained tutorial for building morphological analyzers. https://fomafst.github.io/morphtut.html. Accessed: 2019-09-30.

Benjamin Hunt, Emily Chen, Sylvia Schreiner, and Lane Schwartz. 2019. Community lexical access for an endangered polysynthetic language: An electronic dictionary for St. Lawrence Island Yupik. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 122–126, Minneapolis, Minnesota. Association for Computational Linguistics.

Anna Kazantseva, Owennatekha Brian Maracle, Ronkwe'tiyóhstha Josiah Maracle, and Aidan Pine. 2018. Kawennón:nis: the wordmaker for Kanyen'kéha. In *Proceedings of the Workshop on Computational Modeling of Polysynthetic Languages*, pages 53–64, Santa Fe, USA. Association for Computational Linguistics.

Jordan Lachler, Lene Antonsen, Trond Trosterud, Sjur Moshagen, and Antti Arppe. 2018. Modeling Northern Haida verb morphology. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*, Miyazaki, Japan. European Language Resources Association.

Patrick Littell. 2018. Finite-state morphology for Kwak'wala: A phonological approach. In *Proceedings of the Workshop on Computational Modeling of Polysynthetic Languages*, pages 21–30, Santa Fe, USA. Association for Computational Linguistics.

Jeffrey Micher. 2017. Improving coverage of an Inuktitut morphological analyzer using a segmental recurrent neural network. In *Proceedings of the 2nd Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 101–106, Honolulu. Association for Computational Linguistics.

Sarah Moeller, Ghazaleh Kazeminejad, Andrew Cowell, and Mans Hulden. 2018. A neural morphological analyzer for Arapaho verbs learned from a finite state transducer. In *Proceedings of the Workshop on Computational Modeling of Polysynthetic Languages*, pages 12–20, Santa Fe, USA. Association for Computational Linguistics.

Maïa Ponsonnet. in press. Comitative applicative constructions and their "transfer" extensions in Dalabon and other Gunwinyguan languages (non-Pamanyungan, Australia). In Myriam Bouveret, editor, *A Contrastive study of Give Constructionalization*. John Benjamins Publishing Company, Amsterdam.

Brian Roark, Richard Sproat, and Richard William Sproat. 2007. *Computational approaches to morphology and syntax*, volume 4. Oxford University Press.

Lane Schwartz, Emily Chen, Benjamin Hunt, and Sylvia Schreiner. 2019. Bootstrapping a neural morphological analyzer for St. Lawrence Island Yupik from a finite-state transducer. In *Proceedings of the 3rd Workshop on the Use of Computational Methods in the Study of Endangered Languages Volume 1*, pages 87–96, Honolulu. Association for Computational Linguistics.