Constructing a morphophonological analyzer for Lushootseed*

Joshua Crowgey
University of Washington

Abstract: This describes ongoing work to construct a finite-state morphophonological analyzer for Lushootseed. The language has a large corpus of transcribed text but no electronic lexical resource is currently available. I describe how the analyzer is being incrementally developed and evaluated by looking at coverage rates as the analyzer is applied to new text. The focus is on the methodology and use-cases for the tool under description but some initial results are also presented.

Keywords: Lushootseed, finite-state morphology, implemented linguistics

1 Motivation

This paper describes ongoing work on the construction of an implemented morphophonological analyzer for Lushootseed. The motivation of the project is threefold: (1) to encode knowledge about the morphophonological system of Lushootseed in a way that can be distributed, modified and improved; (2) to provide an interface representation for syntactic/semantic parsing; and (3) to feedback to documentary and descriptive efforts.

The first goal, machine-encoding of linguistic hypotheses, improves the quality of those hypotheses along dimensions of reproducibility, integration[1] and, crucial to any scientific effort, falsifiability. This work therefore aims to not only propose and test an analysis, but to make the resulting resource available to others so that it can be challenged by new data and revised accordingly. For both theoretical and practical reasons (discussed below) the tool takes the form of a finite-state transducer, implemented using the `lexc` and `xfst` languages (Beesley and Karttunen 2003); testing was done using the FOMA library (Hulden 2009), a free and open-source implementation of the Xerox tools.

With respect to the second motivation above, linguistic phenomena are distributed across a collection of subsystems which are taken to interact minimally.

[*] I would like to acknowledge encouragment from Angela Wymer, data from David Beck, open-source software (and usage tips) from Mans Hulden, phonological insights from Sharon Hargus, grammatical and practical advice from Emily Bender, Lushootseed specific discussions with Zalmai Zahir and massively helpful editorial comments from the UBCWPL editors. While these people significantly improved the current work, they are not responsible for remaining flaws contained herein.
Contact info: jcrowgey@u.washington.edu

[1] See the call for linguists to scale-up and integrate analyses found in Bender and Good (2010).

In Papers for the International Conference on Salish and Neighbouring Languages 49,
University of British Columbia Working Papers in Linguistics 37,
Natalie Weber, Emily Sadlier-Brown, and Erin Guntly (eds.), 2014.

In principle, this means that a description of a linguistic phenomenon in one subsystem does not have to make reference to properties from another subdomain. Traditionally, the points at which subsystems interact are referred to as interfaces. Research on the morphophonological analyzer under discussion here has been undertaken in part to provide an interface representation to a syntactic grammar. The plan, then, is to use the morphophonological analyzer to parse Lushootseed orthography[2] to a regularized form which can be input to the syntactic grammar and vice versa, to map output of the syntactic grammar (when it is used in the generation direction) to an orthographic form. To illustrate, consider the form of the stative prefix /ʔas-/, which is spelled varyingly [as], [əs], or [ʔəs] in different morphophonological environments, as illustrated in (1).

(1) a.

xʷiʔ	gʷəsəsaydubs	ʔə	tiʔəʔ	diʔəʔ
xʷiʔ	gʷə=s=ʔas-hay-dxʷ-b=s	ʔə	tiʔəʔ	diʔəʔ
NEG	SUBJ=NMLZ=STAT-known-CAUS-PASS=POSS.3SG	PREP	PROX	here

It is not known by the children. [lut][3]

Basket Ogress—ʔalataɬ Martin Sampson[4]

b.

ʔəshuyəxʷ	tiʔiʔəʔ
ʔas-huyu=axʷ	tiʔ<iʔ>əʔ
STAT-made=now	PROX<PL>

They are ready. [lut] Basket Ogress—ʔalataɬ Martin Sampson

c.

gʷəl	tasɬaɬlil	tiʔiɬ	kikəwič
gʷəl	tu=ʔas-ɬaɬlil	tiʔiɬ	ki-kəwič
SCONJ	PST=STAT-live	DIST	DIM-hunchback

And Little Hunchback dwelled there. [lut]

Basket Ogress—Dewey Mitchell

From the point of view of an linguist working on a syntactico-semantic analysis, the variations in (1)[5] are irrelevant. Encoding them into the grammar not

[2] The motivation for using Lushootseed orthography is twofold: (1) while there are deviations, in general Lushootseed orthography adheres to the phonemic principle; (2) this work is intended to provide a resource for NLP tools for Lushootseed. If Lushootseed readers and writers use the orthography, this tool must also use the orthography.

[3] The ISO:639-3 language code is included in the free translation line of the examples in this paper in order to aid automatic language identification for systems which extract linguistic examples from papers published on the web (Lewis and Xia 2010).

[4] Unless otherwise indicated, all Lushootseed examples in this paper are from Beck and Hess (2014a,b). For each example, the story title and name of raconteur are indicated after the free translation.

[5] Some glosses adapted to provide better compliance with Leipzig Glossing Rules (Comrie et al. 2008).

only complicates the analysis, it also makes the grammar more difficult to edit and maintain (Bender and Good 2005).

With respect to the third motivation, providing feedback to documentary and descriptive efforts, the work described here has already led to an encouraging correspondence with David Beck and a number of discoveries about the analysis of Lushootseed morphophonology in Beck and Hess (2014a,b).[6]

Having outlined the motivations for the project, in the next section I review some formal underpinnings of the finite-state model of morphophonology which is adopted here. After this, in Section 3, I describe the iterative methodology being used to develop the analyzer and the evaluation strategy, along with some initial results. Section 4 briefly discusses the feedback to the documentary efforts to date and the final section point to future work and concludes.

2 Background: Morphophonology as a Finite State Machine

In this section, I discuss some preliminary facts and results from formal language theory which motivate the approach taken in this research. This discussion is necessarily abridged. See Beesley and Karttunen (2003) for a full introduction to the linguistic applications of finite state networks.

Formal language theory begins by defining a language as a possibly nonfinite set of strings. The strings of a particular language are drawn from a collection of symbols called the alphabet of that language. For a given alphabet (Σ) and language (L), formal language theory develops methods and algorithms for deciding whether a particular string of symbols $w \in L$. These notions provide a starting point for discussing and analyzing the properties of nonfinite languages (Hopcroft et al. 2006). Because nonfinite languages can never be directly enumerated, one important aspect of the theory provides an organization of languages into complexity classes based on the types of algorithms, operations and calculations that we can use to explore them. Broadly speaking, these complexity classes are arranged into the familiar Chomsky-Schützenberger hierarchy (2) (Chomsky 1959), wherein the formal devices for a describing a language of type n can be used to describe a language of type $> n$ but not vice-versa.

[6] The possibility for a productive relationship between computational and theoretical efforts has been described in Bender and Langendoen (2010).

(2)

Type	Languages	Devices
0	Recursively Enumerable	Turing Machine
1	Context-Sensitive	Linear-bounded non-deterministic Turing Machine
2	Context-Free	Non-deterministic pushdown automaton
3	Regular	Finite state automaton

The regular languages are those which are least complex on this hierarchy and therefore, most constrained in terms of the formal apparati which define them (rules or automaton). Regular languages are formally defined with respect to an alphabet of symbols Σ as in (3) (Kleene 1956).

(3) a. The empty language $\varnothing$ is a regular language.

b. $\forall a \in \Sigma, \{a\}$ is a regular language.

c. If A is a regular language, A^* is a regular language.

d. If A and B are regular languages:
$A \cup B$ is a regular language.
$A \cdot B$ is a regular language.

e. Nothing else is a regular language.

The crucial takeaway from the definition in (3) is that only three operations serve to generate any regular language: union ($\cup$), concatenation ($\cdot$) and Kleene closure (*). Union is the same operation as is familiar from set theory. Concatenation of languages A and B refers to a language which contains all strings created by appending a string from B onto a string from A. Thus if $A = \{watch, curtain\}$ and $B = \{ed, ing\}$, $A \cdot B = \{watched, watching, curtained, curtaining\}$. Kleene closure, symbolized by *, for a language L indicates the set of strings which contain 0 or more occurrences of any of the strings of L, in any order. These three operations, when applied to regular languages, always define a language which is also regular, thus we say regular languages are *closed* under concatenation, union, and Kleene closure.

Because it is relevant to the model proposed in this paper, it is important to define the notion of a finite-state automaton, a device which can be used to represent a language. Crucially, defining an automaton allows the modelling of a language in terms of acceptance and generation. That is, when we define an automaton for a language, we can use the automaton to implement an algorithm for deciding whether a given string belongs to the language or not. Similarly, we can explore paths through the automaton to begin to enumerate strings of the language. Mathematically, a finite state automaton is conceived of as an n-tuple with 5 components: a set of input symbols (the alphabet of the machine) Σ, a

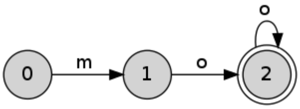

Figure 1: Automaton to model the nonfinite whimsical language of bovines $/moo^*/$.

collection of states S, a distinguished "start" state $s_0 \in S$, a transition function δ, and a set of final, or accepting states F (Hopcroft et al. 2006). Equivalently, we can view an automaton as a graph in which the vertices are states, and labeled arcs which connect vertices constitute the transition function. One vertex is the designated start state and 0 or more states will be designated as accepting states. The graph, then, provides a straightforward way to decide whether an arbitrary string is in the language modeled by the machine. The procedure is to read symbols from the string in question one at a time, moving our attention from vertex to vertex on the graph in correspondence with the symbols found on the arclabels. In more detail, the machine is said to begin in the start state and read the first input symbol. If there is an arc out of the start state which is labeled with that symbol, the machine transitions to the state which the arc points to and reads the next symbol. If the machine reads a symbol for which there is no arc leaving the current state, it is said that the machine "dies" and rejects the input. No further input needs to be read in order to know that the string of symbols which was input is not a string of the language modeled by the machine. If the machine survives to the end of the input, and the final state is an accepting state, then the machine accepts the input, i.e., the input is a string of the language modeled by the automaton. To illustrate, consider Figure 1, which presents an automaton to model the nonfinite language consisting of strings formed by an *m* followed by 1 or more *o*s. In this sort of presentation, accepting states are indicated by the double circle around the vertex.

One more preliminary notion, the *regular relation*, must be defined in order to apply this discussion to our domain of interest, the morphophonological properties of Lushootseed. A regular relation is a relation which is defined to hold between two regular languages A and B wherein for each string in A there is corresponding string in B (and vice-versa). Thus a regular relation is a possibly nonfinite set of ordered pairs (a,b) where $a \in A$ and $b \in B$. Furthermore, regular relations are those which can be modeled by finite state transducers. These are equivalent to the finite state automata discussed above except that (1) the arcs are labeled not with a single symbol, but with symbol pairs and (2) the machine is said to read the first symbol of the pair from the input, and upon taking a transition, to print the second symbol of the pair upon an output tape. In the traditional metaphor, the machine reads from the top side of a tape and prints the

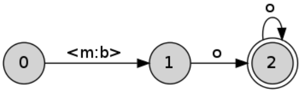

Figure 2: Transducer mapping the whimsical language of bovines $/moo^*/$ to a whimsical language of specters $/boo^*/$.

output symbol on the lower side. In this way, such a machine is said to model the regular relation between two languages, the ‘upper’ and ‘lower’ languages of the machine. Figure 2 presents a finite state transducer in which the upper language is the same as the language of Figure 1, and the lower is made up of strings constructed by *bo* followed by any number of *o*s. This is the nonfinite relation of pairs: $\{(mo,bo),(moo,boo),(mooo,booo)...\}$.

Regular relations or (equivalently) finite state transducers have another closure property which is not relevant for regular languages: that of composition. The composition of two regular relations R and S, written $R \circ S$, is also a regular relation. $R \circ S$ is the relation which maps a string $a \in$ the upper language of R to $b \in$ the lower language of S just in case R maps a to x and S maps x to b. Informally, composition consumes the inner representations. Because regular relations are closed under composition, the composition of two regular relations is also a regular relation, and can also be represented as a finite state transducer mapping an upper language to a lower language.

Finally, then, we can begin to see the utility of these devices for the construction of a model of morphophonology. I take morphophonology to subsume morphotactics (what morphs can co-occur and in what order) and phonology (a system mapping canonical forms into surface forms based on their environments). Consider, then, that a finite collection of prefixes is a regular language (call it P) and, similarly, a collection of roots is a regular language (call it R). We know from the discussion above that we can construct a finite state automaton to model these languages. The concatenation of $P \cdot R$ is also a regular language (call it S, a putative language of stems), and can be modeled via a finite state automaton. Crucially, a set of tools to facilitate defining such automata, along with implementations of the operations discussed above, has been released for linguists to take advantage of (Beesley and Karttunen 2003). Thus, given computer files containing regular languages modelling proclitics, prefixes, roots, suffixes and enclitics we can begin to use these operations to model a system of morphotactics.

Since Johnson (1972), it has been known that SPE-style[7] phonological descriptions which consist of a series of string-rewriting rules of the form $A \rightarrow B/C\_\_\ D$, despite their superficial similarity to generalized rewriting systems,

[7] *Sound Pattern of English*, Chomsky and Halle (1968).

are actually employed by phonologists in such a way as to make them equivalent to finite state transducers.[8] This is a fortunate result because the closure properties of finite state transducers mean that, similar to the model given above for morphotactics, we can use known operations to derive a single regular relation (finite-state transducer) from a cascade of individual transducers. This is especially appealing because the resulting network performs computations which are equivalent to the stepwise rule-by-rule derivations of linear phonology, but without producing intermediate representations.

To provide an illustration of this notion, consider the spelling of the Lushootseed stative prefix cited in (1c). The form *tasɬaⱡlil* can seen as a result of the application of two phonological rules, one which deletes the glottal stop of an aspect prefix when it is preceded by a proclitic (4),[9] and one which deletes a *u* preceding a vowel (5).

(4) $? \rightarrow \varnothing \;/\; = \_\_ \begin{Bmatrix} \text{as} \\ \text{u} \end{Bmatrix} \text{-}$

(5) $\text{u} \rightarrow \varnothing \;/\; \_\_ \; (=) \; \text{V}$

(6) $\{(\mathrm{a},\mathrm{a}),(\mathrm{aa},\mathrm{aa}),\ldots,(\mathrm{ab},\mathrm{ab}),\ldots,(\mathrm{a}=\mathrm{as}-\mathrm{bb},\mathrm{aasbb}),\ldots\}$

We can model the rule in (4) as a transducer which captures the relation in which the upper language contains any string and most of those strings are mapped to identical strings in the lower language. However, strings in the upper language which contain a substring which matches the structural description of the rule would be mapped to strings in the lower language in which the structural change has been carried out. This infinite relation contains pairs as shown in (6). This relation as a finite state transducer is shown in Figure 3. The rule in (5) similarly can be modeled as a regular relation and a transducer defined, also shown in Figure 3.

Figure 4 shows two automata which model regular languages representing the collection of temporal proclitics and two aspect prefixes. In Figure 5, the Kleene closure of the former automaton (representing optional, cyclic application of proclitics), and 0 or 1 occurrences of the latter (representing optional application of an aspect prefix) are concatenated with an automaton modelling the language of the two stems *ɬaⱡlil* and *ʔux̌ʷ*. In this simplistic example, we have a model of a small fragment of Lushootseed morphotactics. Note that any

[8] The crucial point of expressive power lies in a rule's ability to use its own output as a structural description for successive application. Johnson outlines the few phonological rule types which go beyond the complexity of regular relations—they are notoriously few. See also Kaplan and Kay (1994), who reproduce and augment Johnson's findings using different formal arguments.

[9] All proclitics seem to trigger this, not just ones that end in a vowel.

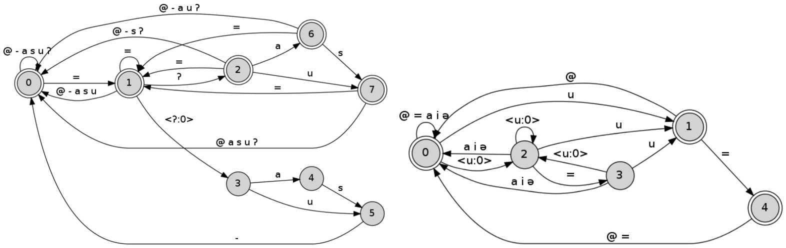

Figure 3: Transducers which implement the rules shown in (4) (left) and (5) (right). In these diagrams, the symbol @ is a metacharacter which stands for any symbol (even one not in Σ for a given machine).

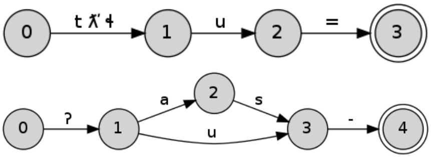

Figure 4: Automata to model the language of three temporal proclitics: $\{\text{tu} =, \text{u} =, \text{u} =\}$ and the language of two aspect prefixes $\{\text{as}-, \text{u}-\}$, respectively.

automaton can be also be thought of as a transducer representing the identity relation for that language. Therefore we can use composition to build a new transducer which models the relation of the lexical, or underlying forms, to surface forms with the deletion rule applied.

This illustration highlights both the utility of the finite state calculus as a model for implementing and testing hypotheses about the morphophonology of a language as well as the need to allow computational tools to perform the actual operations of composition, concatenation, etc. which are used to build up the network. Figures 3 – 6 also highlight the fact that the networks necessarily get unwieldy very quickly if we were to attempt to manipulate them by hand. Yet by defining these networks as a series of concatenated lexicon files and phonological transformations we can take advantage of the perspicuity of the traditional phonological representation.

In this way, finite state transducers have been used as a formal basis for implemented models of morphophonology. At a high level, the idea is to first model lexical classes and their combinatory potential as regular languages and operations upon them, yielding a finite state automaton which recognizes strings of the language. After that, one can define phonological rules in a series of finite state transducers, which models the context dependent transformations that

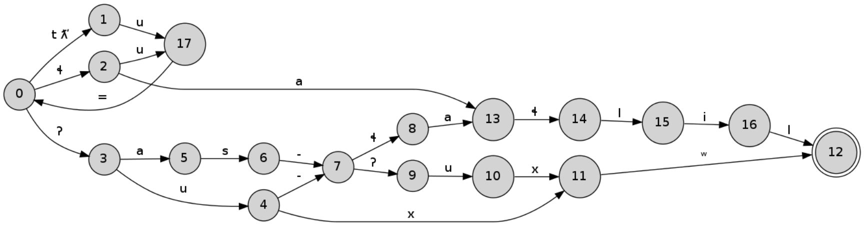

Figure 5: Automaton created by concatenating three regular languages: (a) the Kleene closure of the language of Proclitics, (b) the language of 0 or 1 occurrence of the language of Prefixes (both in Figure 4) and (c) an automaton which models the language of two stems *ɬaɬlil* and *ʔux̌ʷ*.

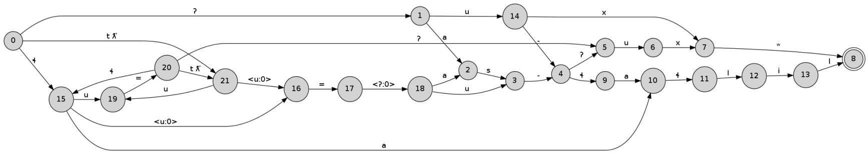

Figure 6: Transducer created by composition of the automaton in Figure 5 with the transducer of Figure 3.

map canonical forms to surface variants. Finally, the operation of composition can be used to build a large network which models these properties in a single computational object.

3 Resources, methodology, initial results

Developing a morphophonological analyzer for Lushootseed necessarily draws on preexisting resources, insofar as they are available. There is a rich body of literature on the grammatical systems of Lushootseed beginning with Hess (1967). Another valuable resource are the pedagogical grammars by Hess and Hilbert (1995a,b) and the grammatical snapshots in the first parts of Hess (1995, 1998). Most drawn upon for this work has been Beck (nd), which provides a sketch of morphotactic position classes and selectional allomorphy.

The ideal starting point for implementing a model of morphophonology is a lexicographical source which provides a listing of the roots, stems, affixes and clitics which populate the continuation classes of the morphotactics. Because this work aims to create an implemented model, a machine-readable lexicon is would be maximally useful. However, the principle lexicographical source on Lushootseed, Bates and Hilbert (1994), is currently only available in paper form.[10] Thus while one can read about Lushootseed morphophonology in the literature and review the lexical items in paper form, the challenge of building up a reasonable-sized collection of machine-readable roots and stems in order to populate the underlying lexical files is a serious one.

3.1 Iterative development

Thanks to a collaboration with David Beck, I was able to answer this challenge by starting with an electronic version of the texts in Beck and Hess (2014a,b). This collection of four-line interlinear texts provides both an entrance point for building a lexical resource as well as a set of gold standard[11] data against which the developing machine can be evaluated. This is because the second line of these texts presents a regularized, morphophonemic line, the ideal input for a syntactic grammar. Because this data was hand-created by experts of the language, it provides a standard against which the output of the automatic system being constructed can be evaluated . Exceptions to this wholesale adoption of the format in the second line of Beck and Hess (2014a,b) as the target output of this tool mainly have to do with the representation of reduplication, which is discussed in the appendix to this article. The methodology undertaken here, then, is summarized in Figure 7 and described in the following paragraphs.

The process starts by selecting a story for preliminary development. “Basket Ogress” as told by ʔalataɬ Martin Sampson was chosen. The initial system is then set up by looking through the lexical items in the story and populating the lexicon files with roots and affixes as they are found in the text. As a starting point, the morphotactic system was adapted from the table in Beck (nd:30). The relationships between the lexical forms and orthographic forms were observed in the first text and phonological rules were posited and tested.[12]

Once the analyzer has accepted 100% of the wordforms in the preliminary story and mapped them with 100% accuracy to the lexical forms in the gold data, evaluation upon unseen text begins. A second story is chosen from the corpus, and the system as developed based on the data in the preliminary text is run against the second story. The following metrics were recorded: (1) coverage rates for types, tokens, (2) ambiguity, (3) accuracy against the gold standard.

[10] Although Bates and Lonsdale (2010) describe the conversion of the dictionary from legacy software to a modern XML representation and Lonsdale and Matsushita (2011) make use of this electronic resource to build a two level model of Lushootseed morphophonology, neither the electronic dictionary nor the two-level model are available to the public or to other researchers at the time of this writing.

[11] In computational linguistics, the term “gold standard” is used to refer to hand-created data against which the output of a computational system can be evaluated.

[12] Much other initial work was done at this stage to facilitate an overall development environment: scripts were written to do the updating of the system network, the running of the analyzer against texts, as well as evaluation of an output against the gold standard.

Coverage refers to whether or not the analyzer returns at least one analysis for a given form. Ambiguity, measured overall in readings per type, refers to the fact that there can be more than one analysis for a given wordform. Accuracy against the gold standard refers to whether or not the output of the analyzer matches the analysis of that wordform in the second line of Beck and Hess (2014a,b). In order to measure the ability of the analyzer to generalize to unseen word forms, the metrics were also recorded for the analyzer after removing any word triples (form, lexical form, gloss) which are found in the preliminary texts.

Next, development continues against the second story: first, any wordforms which failed to be accepted because of missing stems in the lexicon files (either roots, affixes, lexical suffixes or clitics) are counted and those stems are added. Then, another set of measurements are taken. This measurement effectively extracts the error rate given by missing lexical material, leaving an evaluation of the system's combinatorics and transformations.

After this, the remaining forms which fail to be analyzed do so because of some morphotactic or phonological rule which was previously unimplemented. These rules are then added until coverage on the second story is up to 100% with 100% accuracy as measured against the gold standard. The analyzer is now ready to be tested against a third story. The steps of the methodology are to be repeated continually against new texts until the coverage rates begin to approach an upper limit.

This methodology builds in a running evaluation which is intended to show the maturation of the system over time. By measuring against only unseen forms, we can see how well the system generalizes. By measuring against new texts again after adding missing stems, we can see how much of the error rate against a text is due to missing stems and how much is due to missing morphotactic or phonological rules.

In this way we can expect that as the system matures, the error rates should asymptotically approach a low rate—not zero. The value for this rate will be empirically established later in this research program.

3.2 Initial results

Having discussed the methodology and the evaluation strategy for this project, I can now present some initial results. The preliminary text chosen was "Basket Ogress" as told by ʔalataɬ Martin Sampson (MS). This text has 83 sentences, 395 word tokens which fall into 158 unique wordtypes.

After the preliminary development stage the system captured all this data with 100% accuracy finding 14 wordtypes to be ambiguous (average ambiguity per type of 1.12). At this point the system constituted the first iteration of the analyzer to be tested on unseen text. "Basket Ogress" as told by Dewey

0. preliminary:
    (a) Choose an initial story for development
    (b) Populate lexicon files with roots and affixes found in the text
    (c) Set up morphotactic combinations based on preliminary table in Beck (nd)
    (d) Define phonological rules to map lexical forms to surface forms based on the correspondences in the gold data
1. run morphological analyzer against a new story, record evaluation data
2. record evaluation data against unseen forms only
3. add missing stems, record evaluation data
4. modify morphotactics and phonology as needed
5. select a new text, return to step (1)

Figure 7: Enumeration of the methodology

Mitchell (DM) was selected for this.[13] DM's telling consisted of 87 sentences, 428 wordtokens falling into 123 word types. The initial system failed to provide an analysis for 123 of these types a coverage rate of (36.27%). The rejected types consisted of 147 tokens, so only 34.35% of tokens were rejected (coverage rate: 65.65%) . That is, the types which were rejected were not high frequency types. The system found 4 ambiguous types with an average ambiguity of 1.04 per type. While many types were rejected by this initial analyzer, the types which did receive coverage were analyzed accurately: 98.93% of tokens which received coverage matched the gold data analysis.

The next step was to evaluate the performance of the system on unseen wordtypes. This test consisted of wordforms from DM's telling after subtracting wordforms which occurred in MS's telling, leaving 189 wordtokens falling into 159 wordtypes. As expected, the performance of the system on unseen

[13] The collection of texts in Beck and Hess (2014a,b) contains six tellings of "Basket Ogress", five tellings of "Star Child", three tellings of "Mink and Tutyika". It was decided beforehand that during development, tellings of the same text would be added in sequence. Because of the measure on unseen wordforms only, this does not adulterate our ability to view the system's ability to apply to generalize to unseen forms. That is, despite the narrative similarity, extracting the unseen forms shows that the tellings do not present exactly the same collection of words. See in Table 1 that ML's telling of "Basket Ogress" contains 397 wordforms, 347 of which are not found in MS or DM's telling.

words is necessarily lower than the performance of the system on the entire text. Coverage fell to 22.64% percent of types (22.22% of tokens), 2 types were ambiguous giving an average ambiguity of 1.02 per type. While this error rate is rather high on unseen words, overall accuracy was still quite high at 92.86%. Although these initial numbers are low, they provide a baseline upon which we hope to see the system improve over time.

Next, the rejected wordforms from DM's telling were inventoried and categorized as to whether the failure was due to missing lexical material or missing morphotactic/phonological rules or both. Missing lexical material was added and the evaluation metrics were recorded again. Coverage rose to 73.58% of types (87.15% tokens) with ambiguity still relatively low at 9 types (1.07 average per type). Accuracy remained high at 98.39%.

After this, morphotactics and phonological rules were modified until analysis of the DM telling reached 100% coverage with 100% accuracy. A third text was selected: Martha Lamont's (ML) telling of Basket Ogress. This text is significantly larger than first two with 240 sentences which is 987 wordtypes distributed across 397 types. Coverage rose to 55.16% of types, which accounted for 74.77% of tokens. Ambiguity was noticeably higher at 1.15 readings per type. Accuracy remained high at 98.08%.

As before, the system was also measured after removing from ML's telling any wordform triples (orthography, lexical form, gloss) which occur in either MS or DM's telling. This left 342 types instantiated in 473 tokens. The new iteration of the system provided analyses for 48.15% of types (up from 22.64% against the unseen forms in the first iteration), or 46.57% of tokens (as compared to 22.22% in the first iteration). This is an encouraging result: the system's performance on completely unseen forms has improved by approximately 25%.

Finally, items which were rejected for missing lexical material were inventoried, and that material was added to the lexicon files and the tests were run again. This second iteration of testing after adding missing lexical material provided coverage for 87.15% of the stems in ML's telling of "Basket Ogress" (92.21% of the tokens). The data in Table 1 shows the entire set of measurements to date; in order to better illustrate the trend, the chart in Figure 8 shows presents some of these data graphically.

3.3 Discussion

The preliminary results presented here are encouraging. They show a consistently high accuracy rate, due to the hand built nature of the system.[14] And

[14] Broadly speaking, in computational linguistics, much success has been achieved by constructing systems built on statistical models. There is a known complementary dichotomy whereby hand-built systems tend to be accurate, often at the cost of coverage, and statistical models tend to provide high rates of coverage, often at the cost of accuracy.

	type coverage (%)	token coverage (%)	avg. ambig. (readings per type)	total types	accuracy (% types)
DM	36.27	65.65	1.04	193	98.93
DM.unseen	22.64	22.22	1.02	159	92.86
DM.stems	73.58	87.15	1.07	193	98.39
ML	55.16	74.77	1.15	397	98.08
ML.unseen	48.25	46.57	1.16	342	91.03
ML.stems	87.15	92.21	1.26	397	96.77

Table 1: Table presenting the evaluation data to date.

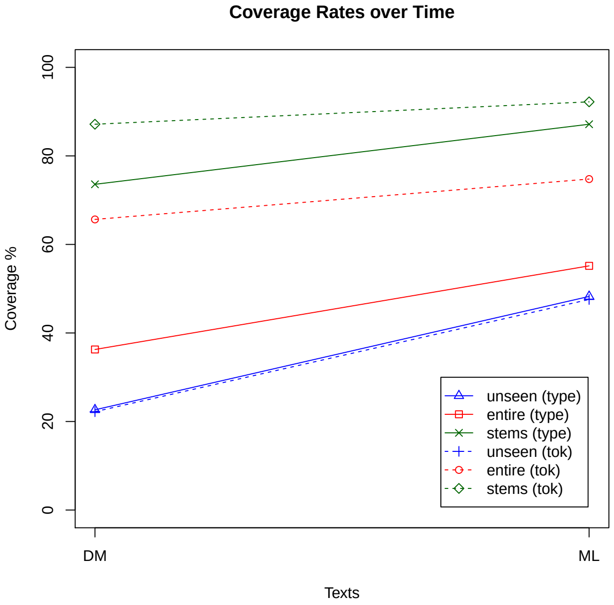

Figure 8: Chart showing the initial performance of the system. Solid lines show error rates against wordtypes, dotted lines against tokens. Red indicates performance on an entire text, blue indicates performance only on unseen wordforms, green shows the reevaluated performance after adding new lexical material.

although coverage on the initial round taken against DM's telling was low, the second set of measurements taken against ML's telling shows us that things are moving in the right direction. Especially encouraging is the 25.61% improvement observed on unseens forms.

In this sort of evaluation scheme we can view each iteration of results as a reference point against which later iterations can be compared. After further iterations of development and evaluation have been performed, we can also begin to measure the rate of improvement over time. As the improvement rate moves towards 0, we will have an empirical measurement of the expected coverage rate and accuracy of the tool on unseen texts. The corollary of this is that upon each iteration of the development and testing cycle, the time per cycle will decrease (there will be less missing stems to add, less phonological or morphotactic rules to add or revise).

4 Structures

In this section, I briefly review some structures and statistics of the system in its current iteration. From a high-level I discuss the linguistic analyses which were implemented in the system to date.

4.1 Lexical classes

This initial treatment of Lushootseed morphophonology has three base lexical classes mnemonically labeled noun, verb and constant. Although verbs, nouns, adverbs, numerals and interrogative words function as the syntactic head of a clause (Beck 2013),[15] here we are concerned about derivational, inflectional and clitic-hosting (morphotactic) potential alone. Van Eijk and Hess (1986) show that nouns and verbs have distinct (but partly overlapping) inflectional and derivational potential: possessive affixes only attach to nouns and causative affixes are restricted to verbal stems. At least some grammatical words (such as the oblique marking preposition *ʔə* or the question marker *ʔu*) do not inflect at all. A caveat: on the whole this subdivision is almost certainly too simplistic to be ideal, yet one should note that this high-level division does not prevent further organization along other dimensions within the network, it merely serves as a basic starting point for morphotactic combinatorics.

4.2 Morphotactics

The current morphotactic system has been adapted from Beck (nd), Essentially, the verbal template was adapted wholesale but with the following changes and

[15] Even the seemingly grammatical preposition *ʔal* (location) is attested in this syntactic position (Bates and Hilbert 1994:5).

extensions: (1) reduplication procedures were implemented cyclically nearest the root with CV- reduplication occurring outside of CVC reduplication; (2) lexical suffixes were allowed to attach just outside reduplication (but inside all other affixation) (3) inflectional affixes were added outside of the derivational material (for verbs: nominalizers and aspect prefixes, object marking and the relational suffix; for nouns: possessive affixes); (4) furthest out, proclitics and enclitics were allowed to attach (the former cyclically). The network graph in Figure 9 presents the current implementation's morphotactic system in a representation in which each position class is a single node in the network. In fact, the actual network explodes each of these nodes into subnets based on the lexical material contained within them.

4.3 Phonological rules

The current system implements 24 phonological rules, with 11 applying at the word level (before clitics are attached) and 13 applying "post lexically". These rules vary widely in their domain of applicability. For instance, there is a rule which removes the initial "h" of *hay* (to make) when preceded by the stative marker (7a), which does not effect other words with similar phonological shape (7b).

(7) a. xʷiʔ gʷəsəsaydubs ʔə tiʔəʔ diʔəʔ
xʷiʔ gʷə=s=ʔas-hay-dxʷ-b=s ʔə tiʔəʔ diʔəʔ
NEG SBJ=NM=STAT-known-DC-PASS=3PO PR PROX here

It is not known by the children. [lut] Basket Ogress (MS)

b. ʔəshuyəxʷ tiʔiʔəʔ
ʔas-huyu=axʷ tiʔ-iʔəʔ
STAT-made=now PL-PROX

They are ready. [lut] Basket Ogress (MS)

Other rules apply very broadly, such as a rule which allows any vowel to be lengthen for dramatic emphasis. Other rules were implemented but allowed to apply optionally. For example, the verb *pusu* sometimes appears with, sometimes without its final vowel, even in a similar environment, as illustrated in (8).

(8) a. xʷi·ʔ gʷəspusdubs tiʔəʔ diʔəʔ kikəwič
xʷiʔ gʷə=s=pusu-dxʷ-b=s tiʔəʔ diʔəʔ ki-kəwič
NEG SBJ=thrown-DC-PASS=3PO PROX here ATTN-HUNCHBACK

Little Hunchback isn't hit. [lut] Basket Ogress (DM)

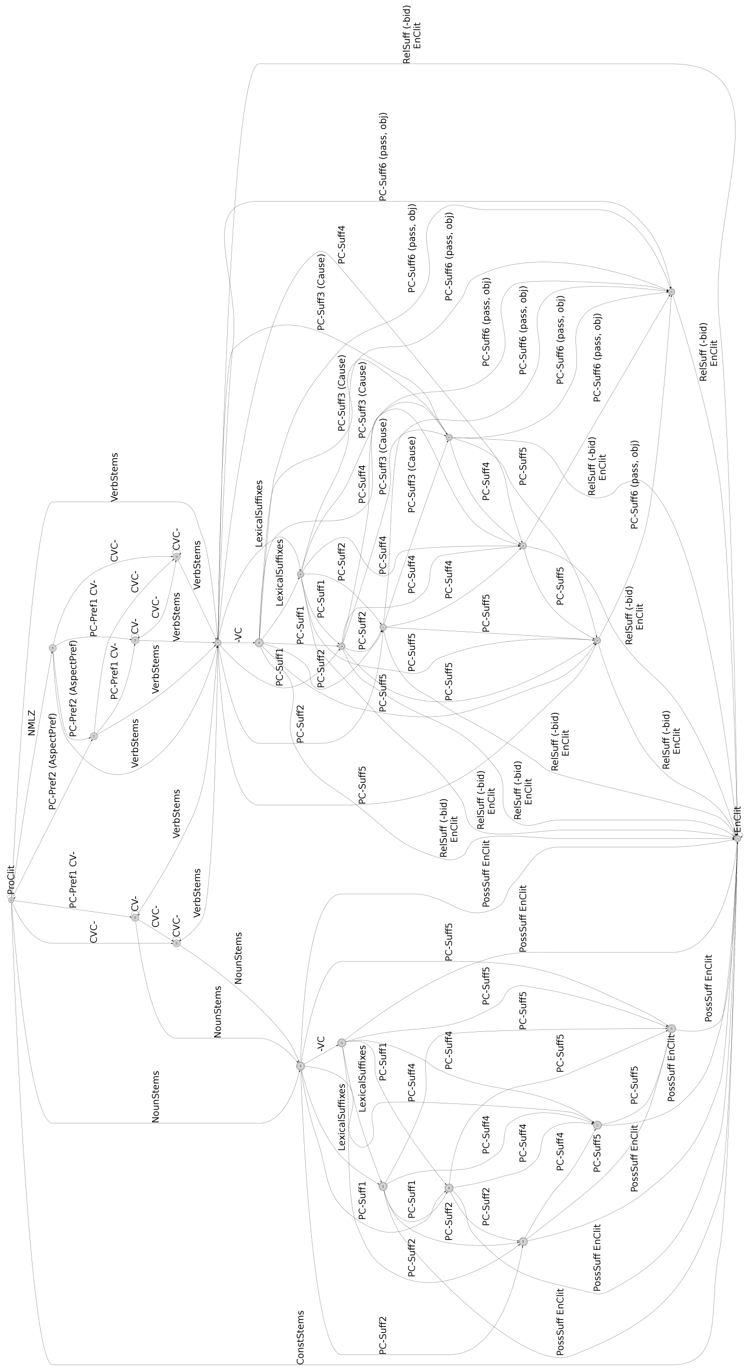

Figure 9: Morphotactic system of position classes

b. ləpuspusutəb
lə=pus-pusu-t-b
PROG=DSTR-thrown-ICS-PASS
He was being thrown at.[lut] Basket Ogress (DM)

Other rules are theoretically uninteresting but important for the practical nature of an implementation of the system. One such rule cleans up boundary symbols after all other rules have applied. Most rules fall between these extremes, applying generally but in specific phonological environments.

Another aspect which constrains the phonological analysis is the representation in the gold standard which was chosen for the project. In analyzing phonological alternations, linguists are often presented with a choice about the directionality of the rule to be written. That is, which of the observed forms will be "underlying". For decision criteria, one traditionally turns to theoretical motivations such as whether one option leads to a system with fewer or simpler rules, or whether one option presents rules which are known to be cross-linguistically more common. However, in the case of this project, it is often the case that the choice has already been made because the second line of Beck and Hess (2014a,b) is the underlying representation. To provide an acute illustration, Lushootseed has a phenomenon in which two of the causative suffixes show a [u $\sim$ x^w] alternation, in which the x^w forms appear word finally and the *u* forms appear when the suffix is followed by another suffix. Hess (1967) uses /u/ and derives [x^w] in word-final position, perhaps motivated by a cross-linguistic phenomenon of word-final devoicing (Blevins 2004). However, Lushootseed also presents an alternation whereby another causative suffix alternates between [d] and [t], with the [d] form appearing word-finally, potentially leading to a system which contains both word-final voicing and devoicing rules. Nevertheless, the option is already decided for the present purposes because Beck and Hess (2014a,b) use x^w. In this way, the phonological analysis of the present implementation is constrained by the choice of gold standard.

5 Relationship to descriptive/documentary efforts

Thus far, the analysis of the first three texts in Beck and Hess (2014a,b) has led to the repair of a number of typographical errors,[16] some discovery of ambiguity patterns henceforth unnoticed as well as a potentially controversial analysis which was subtly implemented in the relationship between the orthographic an morphophonemic lines.

[16] This is reported in order to highlight the utility of machine-implemented analysis, not to belittle the careful work of the researchers who created the data.

To briefly illustrate ambiguity discovery, the system (correctly) produces two readings for the form *ʔuʔux̌ʷ*, in one reading a perfective prefix /ʔu-/ is affixed to the stem *ʔux̌ʷ* ("go"), in the other diminutive reduplication is prefixed. In personal communication, fluent speaker Zalmai Zahir commented that although he could not recall ever hearing the diminutive reduplication attached to the verb *ʔux̌ʷ*, he considered it to be a legitimate wordform, and added that the two forms would be differentiated through stress.[17] This example serves to illustrate some of the ways that an implemented analysis can be of aid to a theoretical linguist—it can highlight consequences of the analysis which are easily overlooked because they are less salient to the linguist who is familiar with the language.

Another, similar example of such an interaction was found in analyzing forms in which a perfective prefix is completely deleted in the context of a temporal procltic, as in (9).

(9) gʷəl tu∅səgʷqtagʷəl tiʔəʔ cədił tasčəbaʔtəb,
gʷəl tu=ʔu-səgʷq-t-agʷəl tiʔəʔ cədił tu=ʔas-čəbaʔ-t-b
SCONJ PAST=PFV-whisper-ICS-RCP PROX s/he PAST=STAT-backpack-ICS-PASS

ʔəsqiq̓tub
ʔas-qiq̓-txʷ-b
STAT-confined-ECS-PASS

Those that had been carried and confined whispered to each other. [lut]
Basket Ogress (DM)

Note that this deletion fits with other observed alternations, it is already captured by the rules in (4) and (5). The two rules conspire to first remove the glottal stop and then one of the *u*s, so there is no remnant of the aspect prefix left in the surface form. The prediction, then, is that anywhere that a temporal proclitic attaches to a verb without an intervening stative prefix (the stative prefix is in complementary distribution with the perfective), we have to posit two analyses: one with and one without the perfective prefix. David Beck suggests (personal correspondence, 2013) that this ambiguity is spurious, that although there are many examples in the corpus of this deletion, he does not find a semantic motivation to posit the perfective in these examples (and there is no surface phonological evidence on which to rely). This anecdote, then, again illustrates the ways in which machine implementation provides the linguist with insights into both the consequences of their analyses, and the phenomena which can sometimes be "hidden in plain sight" in their own data.

[17] However, as stress is not represented in the standard orthography, the system's analysis is taken to be correct.

6 Conclusions and future work

The principle lines of future work in this project concern further development of the morphological analyzer until the coverage rate converges on an upper limit when tested against new texts. After this, the analyzer will be used to preprocess orthographic text for input in a syntactico-semantic grammar, yielding an entire resource toolkit for Lushootseed linguistics. In order to maintain openness, scientific accountability, and reproducibility the system described here is available for download and experimentation at http://students.washington.edu/jcrowgey/lushootseed/ and the version available will continue to be updated as it is developed.

Another point of future interest is the automatic generation of the gloss line of the 4 line IGT format. Theoretically, items on the second line (be they roots, stems, affixes, or clitics) are in one-to-one correspondence with items on the gloss line. Therefore in mapping in the analysis direction (from orthography to morphophonemic representation) it should be possible to also generate the glosses for the morphemes upon user request. The challenges here are practical, then, rather than theoretical. Note as well that generating a gloss line would also disambiguate forms which are currently conflated by merely targeting a morphophonemic form. For example, the question marker *ʔu* has the same form as the vocative interjection *ʔu*, but the gloss line distinguishes them, so implementing gloss generation would capture this.

In conclusion, this paper presents some initial good news about forthcoming computational tools for Lushootseed and discusses an interactive development methodology and evaluation which can be used even when preexisting electronic resources are somewhat scarce. The work has provided some initial (albeit modest) insights into the data it is being developed on, leading to a useful collaboration between the computational linguist developing the tool and the theoretical and descriptive linguist(s) who generated the primary data. This project forms a first step in a larger research program intending to develop an entire suite of open-source, freely usable and redistributable computational resources for Lushootseed, with the hope that these can in turn be drawn upon, not only by linguists in the context of theory, but also by application developers and others who can leverage them to create applications for language-users such as spelling and grammar checkers, machine-translation, games and programs for language pedagogy and perhaps others.

References

Bates, D. and Lonsdale, D. (2010). Recovering and updating legacy dictionary data. In *Proceedings of the 44th Annual International Conference* *on Salish and Neighboring Languages* (*ICSNL*), volume 27 of *University of British Columbia Working Papers in Linguistics*, pages 1–12.

Bates, Dawn, T. H. and Hilbert, V. (1994). *Lushootseed Dictionary*. University of Washington Press, Seattle.

Beck, D. (2013). Uni-directional flexibility and the noun-verb distinction in Lushootseed. In Rijkhoff, J. and van Lier, E., editors, *Flexible word classes: a typological study of underspecified parts-of-speech*. Oxford University Press, Oxford.

Beck, D. (n.d.). Lushootseed morphosyntax. University of Alberta.

Beck, D. and Hess, T. (2014a). *Tellings from Our Elders: Lushootseed syəyəhub. Volume 1, Snohomish texts*. UBC Press, Vancouver.

Beck, D. and Hess, T. (2014b). *Tellings from Our Elders: Lushootseed syəyəhub. Volume 2. Tales from the Skagit Valley*. UBC Press, Vancouver. Forthcoming.

Beesley, K. R. and Karttunen, L. (2003). *Finite State Morphology*. CSLI Publications, Stanford.

Bender, E. M. and Good, J. (2005). Implementation for Discovery: A bipartite lexicon to support morphological and syntactic analysis. *Chicago Linguistic Society*, 41(The Panels).

Bender, E. M. and Good, J. (2010). A Grand Challenge for Linguistics: Scaling up and integrating models. Technical report, National Science Foundation.

Bender, E. M. and Langendoen, D. T. (2010). Computational Linguistics in Support of Linguistic Theory. *Linguistic Issues in Language Technology*, 3(2):1–31.

Blevins, J. (2004). *Evolutionary Phonology: The Emergence of Sound Patterns*. Cambridge University Press, Cambridge.

Chomsky, N. (1959). On certain formal properties of grammars. *Information and Control*, 2(2):137 – 167.

Chomsky, N. and Halle, M. (1968). *The Sound Pattern of English*. Studies in Language. Harper & Row.

Comrie, B., Haspelmath, M., and Bickel, B. (2008). The Leipzig glossing rules: Conventions for interlinear morpheme-by-morpheme glosses. http://www.eva.mpg.de/lingua/resources/glossing-rules.php. Last accessed: July 2014.

Hess, T. (1995). *Lushootseed Reader with Introductory Grammar: vol. I: Four Stories from Edwards Sam*. Number 11 in University of Montana Occasional Papers in Linguistics. University of Montana, Missoula.

Hess, T. (1998). *Lushootseed Reader with Intermediate Grammar: vol. II: Four Stories from Martha Lamont*. Number 14 in University of Montana Occasional Papers in Linguistics. University of Montana, Missoula.

Hess, T. and Hilbert, V. (1995a). *Lushootseed Book 1; The language of the Skagit, Nisqually, and other tribes of the Puget Sound. An Introduction*. Lushootseed Press, Seattle.

Hess, T. and Hilbert, V. (1995b). *Lushootseed Book 2* (*Advanced Lushootseed*). Lushootseed Press, Seattle.

Hess, T. M. (1967). *Snohomish Grammatical Structure*. PhD thesis, University of Washington, Seattle.

Hopcroft, J. E., Motwani, R., and Ullman, J. D. (2006). *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley, Boston, 2nd edition.

Hulden, M. (2009). Foma: A finite-state compiler and library. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics: Demonstrations Session*, EACL '09, pages 29–32, Stroudsburg, PA, USA. Association for Computational Linguistics.

Johnson, C. D. (1972). *Formal Aspects of Phonological Description*. Mouton, The Hague.

Kaplan, R. M. and Kay, M. (1994). Regular models of phonological rule systems. *Computational Linguistics*, 20(3):331–378.

Kleene, S. C. (1956). Representation of events in nerve nets and finite automata. In Shannon, C. and McCarthy, J., editors, *Automata Studies*, pages 3–41. Princeton University Press, Princeton, NJ.

Lewis, W. and Xia, F. (2010). Developing ODIN: A multilingual repository of annotated language data for hundreds of the world's languages. *Journal of Literary and Linguistic Computing* (*LLC*), 25(3):303–319.

Lonsdale, D. and Matsushita, H. (2011). Annotating and exploring Lushootseed morphosyntax. In *Procedings of the International Conference on Salish and Neighboring Languages: volume 46*, volume 30 of *University of British Columbia Working Papers in Linguistics*.

Van Eijk, J. P. and Hess, T. (1986). Noun and verb in Salish. *Lingua*, 69(4):319–331.

A Deviations from ideal output form and gold data

While the second line of Beck and Hess (2014a,b) generally presents an ideal representation of Lushootseed lexical form for the purposes of a morphosyntactic interface, the exceptions all surround the canonical forms given for reduplication. Some examples are presented in (10).

(10)	a.	day̓	tiʔəʔ	kikəwič	ləgʷəb	stubš	kʷi	łuʔux̌ʷtxʷ
		day̓	tiʔəʔ	ki-kəwič	ləgʷəb	stubš	kʷi	łu=ʔux̌ʷ-txʷ
		uniquely	PROX	attn-hunchback	youth	man	REM	IRR=go-ECS

tiʔəʔ	stawixʷaʔł	ʔal	tiʔəʔ	dəxʷʔahəxʷ	ʔə	tiʔəʔ
tiʔəʔ	stawixʷaʔł	ʔal	tiʔəʔ	dəxʷ=ʔa=axʷ	ʔə	tiʔəʔ
PROX	children	at	PROX	ADNM=be.there=now	Pr	PROX

swədəbš
swədəbš
Snohomish

The one who takes the children to Snohomish is Little Hunchback, a youth. [lut] Basket Ogress—ʔalatał Martin Sampson

b.	diłəxʷ	dəxʷʔa	ʔə	tiʔəʔ	sbababadil,
	dił=axʷ	dəxʷ=ʔa	ʔə	tiʔəʔ	sba-ba-badil
	FOC=now	ADNM=be.there	Pr	PROX	ATTN-ATTN-mountain

dəxʷʔahəxʷ	ʔə	tiʔəʔ	town	ʔə	La Conner
dəxʷ=ʔa=axʷ	ʔə	tiʔəʔ	town	ʔə	La Conner
ADNM=be.there=now	Pr	PROX	town	Pr	La.Conner

That̓s why there are so many little mountains, why there is a town of La Conner. [lut] Basket Ogress—ʔalatał Martin Sampson

c.	x̌ʷul̓	ʔiłdᶻak̓ʷadiʔəd	tiʔəʔ	suq̓ʷsuq̓ʷaʔs,	tiʔəʔ
	x̌ʷul̓	ʔił-dᶻak̓ʷadiʔ-t	tiʔəʔ	suq̓ʷ-suq̓ʷaʔ-s	tiʔəʔ
	only	PRTV-invite-ICS	PROX	DSTR-younger.sibling-3PO	PROX

ʔalalšs
ʔal-alš-s
PL-cross.sex.sibling-3PO

Only, he invites his younger brothers and his siblings. [lut]
Basket Ogress—Dewey Mitchell

The second line of the Beck and Hess (2014a,b) data consistently represents reduplication forms based on a segmenting of the orthographic forms. But this is not ideal for both practical and theoretical reasons. I have noted that the one of the use cases of the system being developed is to provide input to an implemented syntactic grammar to be developed in future work. In principle, therefore, representing the forms of reduplicative morphemes as being lexically equivalent to their orthographic forms essentially amounts to a requirement to posit a copy of the morpheme's lexical entry for every stem it can combine with. Instead, the decision made here was to represent the lexical shapes of reduplicative morphemes based on their gloss. Potential representations, then, could either follow the Leipzig Glossing Rules and use *RED* as in *RED1-kəwic* or could indicate the templatic shape of the morpheme as in *CVC-*. Finally, these options were not as attractive as using the gloss string found in the third line of the gold data. As seen by comparing the underlined forms in (10a) to (10b), the gloss line provides a uniform gram to represent this reduplication type (*attn-*) An extension to the evaluation script was then written which allows us to check these forms automatically. Essentially, when the output of the system for a wordform includes the string *attn-*, the gloss (third) line is consulted for evaluation of that segment of the word rather than the morphophonemic (second) line. Therefore, the current output of the system when analyzing *kikəwic* is *attn-kəwic*, and the evaluation script counts this as correct.

Example (10b) shows that CVC reduplication can iterate, and also highlights another challenge for the evaluation script: that of treating the linearization of the infixal nature of CVC reduplicands when they occur on stems which have a leading *s* which is synchronically fossilized. That is, the lexicon of noun stems lists *sbadil*, and prefixing to the lexical side yields *attn-attn-sbadil*, not *attn-attn-badil*. On the orthographic side, the current system rightly accepts/produces *sbababadil*, the issue lies in automatically evaluating the lexical representation *attn-attn-sbadil* against the gold data. That is, even given the modification described above, in which we evaluate against the gloss or the phonemic form when reduplicands are found, the evaluation will fail: the prefixes will both match the gold gloss *attn*, but the stem will fail to match because the gold data has *badil* rather than *sbadil*, and the gloss line has *mountain*. The system's output and inputs are in fact motivated in this case, the stem which means "mountain" is in fact *sbadil* not *badil*. Therefore the crux of the issue is just to give credit to the system in these cases without overgeneralizing and spuriously giving credit where none is due. For the present, the solution adopted is to relax the evaluation script such that when considering forms with reduplication morphemes, an optional "s" in the lexical form output by the system is allowed even when this does not occur in the representation given in the gold. The system has been written to output a warning in these cases so that it can be verified by hand that this relaxation is not spuriously improving the system's performance metrics.

The final example in (10c) shows that the representation of suffixing, or -VC reduplication provides further complications for the evaluation script. That is, the gold data treats the infixal reduplicand as a prefix in the gloss line so the left-to-right, 1-to-1 correspondence between morphs and glosses is broken, obviating the strategy used for prefixal reduplication discussed above—to check the gloss line for evaluation of replicative morphemes. That is, the system currently outputs *ʔal<pl>š-s* when run in the analysis direction against *ʔalalšs*. Again, the system's output is motivated here, but automatic evaluation is difficult. For the present, the evaluation script has been programmed to skip over these forms, printing them to the screen to be evaluated by hand.