Finite-state morphology for Kwak'wala: A phonological approach

Patrick Littell
National Research Council of Canada
1200 Montreal Road, Ottawa ON, K1A 0R6
patrick.littell@nrc.gc.ca

Abstract

This paper presents the phonological layer of a Kwak'wala finite-state morphological transducer, using the phonological hypotheses of Lincoln and Rath (1986) and the "lenient composition" operation of Karttunen (1998) to mediate the complicated relationship between underlying and surface forms. The resulting system decomposes the wide variety of surface forms in such a way that the morphological layer can be specified using unique and largely concatenative morphemes.

1 Introduction

Kwak'wala[1] (ISO 639-3: kwk) is a Northern Wakashan language of British Columbia, spoken primarily on the northern part of Vancouver Island, the adjacent mainland, and the islands in between. Kwak'wala morphology and morphophonology is famously complex; words are frequently made up of many morphemes, and these morphemes can cause dramatic changes in the surface realizations of words.

As a basic example, the root for "man" can occur in various forms depending on the suffixes with which it occurs, and in some of these words (the three ending in *-əm*) the identity of the suffix can *only* be distinguished by the effects (lenition, fortition, vowel lengthening, or reduplication) that it has on the root:

(1)

bəgʷanəm	"man"
bagʷans	"visitor" (literally, "unexpected man")
bəkʷəm	"without expression, sternly" (literally, "man-face") (FirstVoices, 2009)
bəkʼʷəs	"Wildman of the forest"
bakʼʷəm	"First Nations person" (literally, "genuine man")
ba**bag**ʷəm	"boy"

This is compounded by suffixes potentially changing syllable structure as well, which further increases the apparent variety of surface forms:

(2)

de	"to wipe" (Boas et al., 1947)
dixʔid	"to wipe something" (FirstVoices, 2009)
dayaǰstənd	"to wipe one's mouth" (FirstVoices, 2009)
dəʔeɬbənd	"to wipe one's nose" (FirstVoices, 2009)

The combination of mutation and resyllabification can cause a complete restructuring of the word. For example, adding the participial suffix -ɬ to the root piǰʷ- ("feel") results not in something like [*piǰʷɬ] but rather [pəyuɬ], in which only the first and last letters remain intact.

[1] Strictly speaking, Kwak'wala is the most-spoken variety of a larger language for which there is no completely-agreed-upon name. This language is often also called Kwak'wala, but some speakers prefer a more general term Bak'wəmk'ala (Littell, 2016, pp. 29–30).

Proceedings of Workshop on Polysynthetic Languages, pages 21–30
Santa Fe, New Mexico, USA, August 20-26, 2018.

(3)

piǩ^{w} (“feel”)	+ ɬ (participle)	= pəyuɬ (“felt”) (Boas et al., 1947)
məǩ^{w} (“desire”)	+ ɬ (participle)	= muɬ (“desired”) (Boas et al., 1947)
g^{w}as (“chap”)	+ ɬ (participle)	= g^{w}eɬ (“chapped”) (Boas et al., 1947)
k^{w}əns (“bake bread”)	+ k^{w} (nominal)	= k^{w}ənik^{w} (“bread”)
x^{w}as (“excite”)	+ k^{w} (nominal)	= x^{w}ek^{w} (“excited”) (Boas et al., 1947)
guɬ (“eat while traveling”)	+ k^{w} (nominal)	= gəwəlk^{w} (“food for travel”) (Boas et al., 1947)

Given such changes, a grammar engineer has two potential avenues of approach to Kwak’wala morphology:

1. Assume that Kwak’wala has a fairly straightforward relationship between phonemes and surface phones, but that roots and suffixes fall into a large number of derivational classes that behave differently when certain suffixes are added.

2. Assume that Kwak’wala has a comparatively straightforward agglutinative morphology, with morphemes that have unique forms and are mostly separable at the phonemic level, and that the apparent diversity of surface forms is due to a complex phonological component.

The FST described in this paper leans towards the second approach, meaning that the lion’s share of the difficult work will be in the phonological component, while the morphological component should be (mostly) concatenative and assume (where possible) one unique form for each morpheme. Given this, this paper concentrates on creating the phonological component, the success or failure of which determines what form the morphological component must take. The phonological component uses the “lenient composition” technique of Karttunen (1998) to express Kwak’wala phonology in an Optimality-Theoretic way, while maintaining the linear-time efficiency of a finite-state system. This is, to our knowledge, the first attempt at computationalizing Kwak’wala morphophonology.

The downside of the phonological approach here is that there are few people who have a mastery of this particular style of Northern Wakashan phonological analysis, and thus while the resulting system is simpler, it can be difficult even for someone familiar with Kwak’wala to look at the resulting grammar and understand what is going on. To try to mitigate this, we are attempting to write this grammar in a “literate” (in the sense of Knuth (1992) and Maxwell (2012)) style, with a greater proportion of human-readable prose accompanied by relatively short snippets of executable code.

2 Motivation

Ẇä, lā′ᵍlaē á′laEl pâ′lēda ᵍwā′latsEma. Wä, laE′mᵍlaē hē′mEnałaEm ᵍnEmō′-kwēda pō′sdanäxa ᵍnēᵍnā′la. Wä, lā′ᵍlaē yā′qˌeg·ałēda ᵍnEmō′kwē lax aᵍyî′lkwäs Qa′wadiliqala lā′xēs g·ō′kulōtē. Lā′ᵍlaē ᵍnē′k·a: “ᵍyā′x·daᵍxᵘ, wä′Entsōs hō′Lēla g·ā′xEn, g·ō′kulōt, qaEn yā′qˌēg·ałēsg·a ǥwä′łaā′sg·asg·în nâ′qēk·. Wä, hë′ᵍmEn nâ′qaᵍēda, qEns lä hō′ǥwīL lax g·ō′-kwasa g·ī′ǥamaᵍyaEns, qaE′ns ha′waLī′-lagˌâ′lē qEns g·ā′yuˌasE′x haᵍmā′ᵍya.”

Figure 1: An excerpt from Boas and Hunt (1902). The stories, songs, history, and oratory collected by Boas and Hunt constitute a substantial body of text—still the largest corpus of Kwak’wala—and contain much that is of cultural and linguistic interest to this day. However, few modern readers can read this orthography.

The FST described in this paper is intended as part of a spell-checking system, initially intended to help guide the optical character recognition (OCR) of historical texts (e.g. Boas and Hunt (1902)). There

are extensive high-quality scans of documents from the early 20th century, but they are written in an orthography that most modern readers find impenetrable. OCR is the first step to unlocking this content for modern readers.[2] It may also be the case that the resulting spellchecker can be useful for end-users (e.g. in a word processor) and other downstream tasks.

Since we do not have a complete lexicon of Kwak'wala, we cannot at this point design a system that divides *actual* Kwak'wala words from *non-actual* ones. Rather, this system has to divide *possible* Kwak'wala words from *impossible* ones.

This is, however, probably much of what we want a low-resource spell-checker to do. We do not want to limit an OCR system for historical texts, for example, to words known in the modern era; part of the reason for engaging with these texts is to rediscover words that are no longer commonly used. We do, however, want to avoid hypothesizing forms that *could not* be Kwak'wala words.

The morphophonological complexity of Kwak'wala presents an opportunity here, and not just a challenge. Because the morphophonology shapes words in particular ways, given an unknown word we can, with some degree of confidence, (1) guess that it *is* a Kwak'wala word and (2) have some idea of its structure, even if its meaning, and the meaning of its components, are unclear. For example, even if we do not know that the following word means "school", we can determine from its shape that it probably is a Kwak'wala word (not a loanword, or a sequence of random characters, or an OCR error) and that it has four or five component morphemes (because it contains three or four phoneme sequences that we associate with changes that happen across morpheme boundaries).

(4) *q̓aq̓uƛ̓ʔaċi*
q̓a-q̓awƛ-$_s$a-as-$_s$i
try-know-try-LOC-NOMINAL

"school" (literally: "building where one learns")

3 A phonological approach to the complexity of Kwak'wala morphology

Kwak'wala is noted for its highly complex morphology and morphophonology, and is, by the definition of polysynthesis in Anderson (1985), the prototypical polysynthetic language.[3] There are roughly 400–500 suffixal or enclitic elements that can be added to roots, many of which (the "lexical suffixes") have quite concrete meanings of the sort that few languages (outside of the Wakashan and some neighboring languages) express in suffixes. These suffixes express body parts, different sorts of ground the action is done on (e.g. on the beach, on manmade surfaces, in a forest, in a boat, etc.), shapes, paths of motion, and even different kinds of physical technology (e.g. tools vs. containers vs. work surfaces vs. headgear). In addition to this, there is also a layer of inflectional morphology, and beyond this a tendency for all small particle-like words that follow the word to encliticize to the previous word.[4]

On top of this, Kwak'wala phonology and morphophonology is also highly complex, with suffixes causing a variety of mutations (particularly fortition and lenition) in the bases to which they attach. These mutations can then interact with the syllabification, stress, and vowel derivation systems to cause surprising alternations, as in (3).

In order to compartmentalize this complexity, we assume a phonology roughly equivalent to that of Lincoln and Rath (1980) and Lincoln and Rath (1986), which posited that Northern Wakashan words consist underlyingly of sequences of consonants (e.g. /pyx̌^{w}ɬ/ for [pəyuɬ] and /k^{w}nsk^{w}/ for [k^{w}ənik^{w}]),

[2]The second step, conversion between historical and modern orthographies, is already available at `orth.nfshost.com`, although this component will probably also undergo further development with the flood of new historical text that will become available due to OCR.

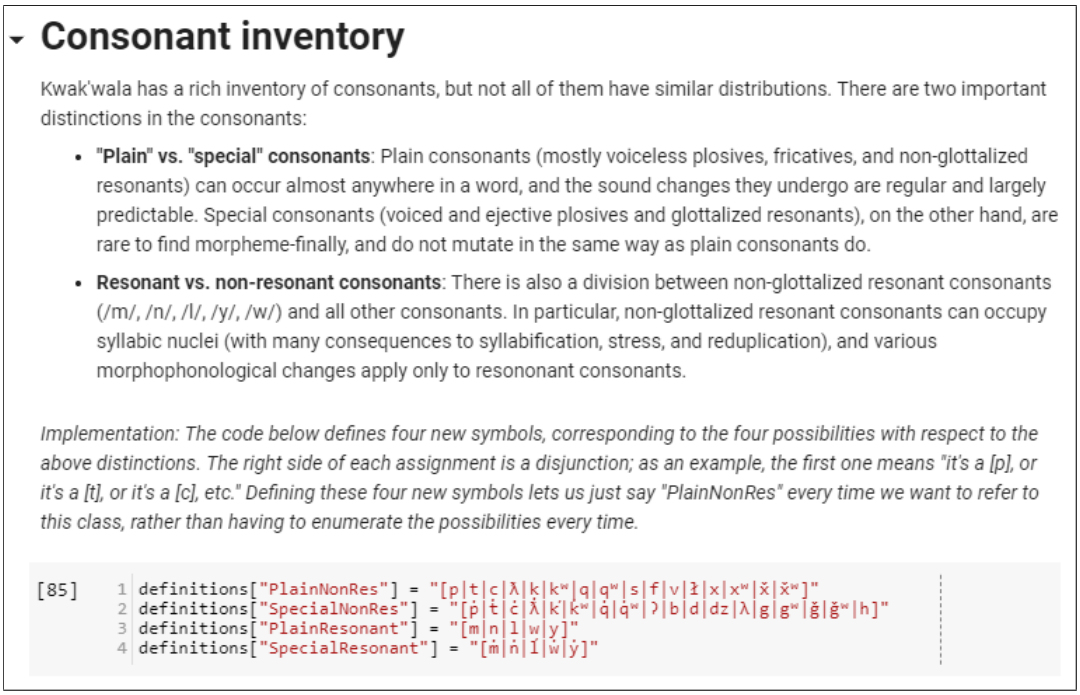
[3]It depends, however, on which definition of polysynthesis one uses. Anderson's definition (which comes out of his own research on Kwak'wala) only requires that the language typically expresses within words what other languages require whole sentences for. On the other hand, if we take the definition of polysynthesis in Baker (1996), which requires verbal inflection for particular structural arguments, Kwak'wala is probably not polysynthetic; much of the complexity of Kwak'wala morphology is probably not *inflectional* per se and does not necessarily involve the arguments intended by Baker.

[4]Kwak'wala also has famously complex reduplication patterns (Struijke, 1998; Struijke, 2000), but this system does not yet attempt to account for them.

which are vocalized by the epenthesis of schwas and the realization of particular consonants as syllabic nuclei (here, the lenition of /x̌ʷ/ to /w/ and /s/ to /y/, respectively, and their subsequent vocalization to [u] and [i]). Since suffixes affect syllabification and can mutate consonants (and thus change their potential vocalizations), different root+suffix combinations can appear to have dramatically different surface forms.

4 Implementation

The phonological transducer is written in the `Foma` (Hulden, 2009) implementation of the XFST language (Beesley and Karttunen, 2003). More specifically, it is written using the Python bindings for Foma, allowing the automation of boilerplate code in Python and the use of Jupyter for "literate programming" (Knuth, 1992; Maxwell and Amith, 2005; Maxwell, 2012) (Fig. 2).



Consonant inventory

Kwak'wala has a rich inventory of consonants, but not all of them have similar distributions. There are two important distinctions in the consonants:

- **"Plain" vs. "special" consonants**: Plain consonants (mostly voiceless plosives, fricatives, and non-glottalized resonants) can occur almost anywhere in a word, and the sound changes they undergo are regular and largely predictable. Special consonants (voiced and ejective plosives and glottalized resonants), on the other hand, are rare to find morpheme-finally, and do not mutate in the same way as plain consonants do.
- **Resonant vs. non-resonant consonants**: There is also a division between non-glottalized resonant consonants (/m/, /n/, /l/, /y/, /w/) and all other consonants. In particular, non-glottalized resonant consonants can occupy syllabic nuclei (with many consequences to syllabification, stress, and reduplication), and various morphophonological changes apply only to resononant consonants.

Implementation: The code below defines four new symbols, corresponding to the four possibilities with respect to the above distinctions. The right side of each assignment is a disjunction; as an example, the first one means "it's a [p], or it's a [t], or it's a [c], etc." Defining these four new symbols lets us just say "PlainNonRes" every time we want to refer to this class, rather than having to enumerate the possibilities every time.

```
[85]  1 definitions["PlainNonRes"] = "[p|t|c|ƛ|k|kʷ|q|qʷ|s|f|v|ɬ|x|xʷ|x̌|x̌ʷ]"
      2 definitions["SpecialNonRes"] = "[ṗ|ṫ|ċ|ƛ̓|k̓|k̓ʷ|q̓|q̓ʷ|ʔ|b|d|dz|λ|g|gʷ|ǧ|ǧʷ|h]"
      3 definitions["PlainResonant"] = "[m|n|l|w|y]"
      4 definitions["SpecialResonant"] = "[ṁ|ṅ|l̓|ẇ|ẏ]"
```

Figure 2: Example of a "literate programming" style, which conceptualizes the primary consumer of code to be human (and thus interested in understanding either the code or the phenomenon that the code purports to capture), and augments this human-consumable description with pieces of machine-interpretable code (in this case XFST regular expressions, interpreted by Foma via a Python interface).

There is not currently a lexical component (e.g. an LEXC file filled with known morphemes); rather, a "guesser" allows any well-formed underlying form. In the future, this will be filled by a devoted lexical component, but since the structure of that component depends largely on whether this phonological component is successful, it has been left to future research. In this experiment, the underlying forms are drawn from a field corpus that includes proposals of underlying forms (§6.1).

5 Phonology and morphophonology in XFST

5.1 Phonetic and phonemic inventory

Kwak'wala has a large phonetic inventory and a complex phonology that is not yet completely understood (particularly concerning the vowel inventory). There are 42 consonants, all of which are underlying[5]. There are approximately 10-12 distinct vowel qualities, but this system follows most modern Kwak'wala

[5]It is possible that [h] is epenthetic, and may historically have been so, but it is not possible to posit that *both* [h] and [ə] are epenthetic due to words like [həmumu] ("butterfly"). Assuming that [ə] is always epenthetic has significant explanatory power for many otherwise-puzzling forms, so this system must assume that /h/ is underlying.

orthographies in representing six distinct surface vowels [ə, a, e, i, o, u]; the surface vowel qualities are almost entirely predictable from the orthographic form.

In a Lincoln and Rath-style (1980, 1986) Northern Wakashan phonology, underlying forms consist primarily of consonants and most vowels are derived (either epenthetic or derived from consonants). Unlike Lincoln and Rath, whose phonemicization is entirely consonantal, we allow three actual underlying vowels, /a/, /i/, and /u/, although all are marginal in some way. [i] and [u] are probably often underlyingly /y/ and /w/, but a few forms suggest that /i/˜/y/ and /u/˜/w/ cannot completely be unified, and we typically default to positing that surface [i] and [u] are underlyingly /i/ and /u/ unless there is specific evidence otherwise. Lincoln and Rath also posit that [a] is a realization of /h/; we instead treat it as a separate phoneme.

5.2 Orthography

Roughly six distinct Kwak'wala orthographies can be identified, in three families:

1. Two stages (early and late) of the orthography used by Boas and his collaborators, and seen in Fig. 1. Most modern readers cannot read this orthography.

2. Two similar orthographies based on Royal British Columbia Museum conventions. The more recent version of this style, called "U'mista" script after the U'mista Cultural Centre in Alert Bay, is the de facto standard for most communities, and is the orthography in which modern books are published.

3. Two variants of the Americanist Phonetic Alphabet, typically used by linguists in the region, and seen in this paper.

The example forms given in this paper are orthographic rather than phonetic, using the typical six orthographic vowels; specifically, this paper is written using the University of Victoria variant of the North American Phonetic Alphabet (NAPA). A caron indicates a uvular consonant, and an apostrophe above a letter represents glottalization; the barred lambda [ƛ] indicates a voiceless lateral affricate.

Although it is intended to be used mostly for documents written in (1)- or (2)-type orthographies, the transducer uses a NAPA orthography internally, because NAPA-type orthographies allow the unambiguous expression of Lincoln-and-Rath-style underlying forms, and allow the differentiation of all the sounds in the test corpus (§6.1).

5.3 Syllabification

For some languages, one can dispense with a detailed syllabification when writing a practical morphological FST, since one can define environments (like "onset") in terms of linear consonant/vowel phonotactics (e.g., "consonant before a vowel"). In Kwak'wala, it is crucial to determine the actual syllabification, because the entire word might consist of consonant phonemes; a phoneme will be *realized* as a consonant or a vowel depending on its syllabification, which can change depending on a variety of factors.

To determine syllable structure, we adopt an approach outlined in Karttunen (1998), in which Optimality Theory-like violable constraints (Prince and Smolensky, 1993) on syllable structure are implemented via the "lenient composition" of transducers.

A lenient composition `X .O. Y` acts as a regular composition `X .o. Y` when the range of `X` overlaps with the domain of `Y`; otherwise, `X` is used alone. This allows the expression of constraints that can be violated: they apply if they would produce output – that is, if there are some "live" candidates that would successfully pass their test – but if they would result in an empty set of outputs they do not apply.

5.4 Counting constraints

Much of the implementation complexity – and the resulting size of the network – comes from the necessity for some constraints to count how many violations of them occur.

For example, consider a DEP constraint ("don't epenthesize") against the epenthesis of schwas – call this "NoSchwa". We can compose this (by lenient composition) with our generator function GEN (which

```
define NoSchwa ~$[ schwa ] ;
define GRAMMAR GEN .O. NoSchwa
```

Figure 3: Example XFST code illustrating a constraint that *cannot* count the number of violations.

creates candidate forms) to exclude forms with schwas when schwa-less forms exist, but to allow forms with schwas when that is the only possibility (Fig. 3).

This is not, however, what we want from the system: we want it to *minimize* the number of schwas. The automaton above cannot count schwas; a word with two schwas (like ƛətəmɬ, "hat") is just as bad within this system as a word with three (like ƛətəməɬ), so any input that successfully generates the correct form ƛətəmɬ will also generate every possible form with additional schwas like ƛətəməɬ.

It is therefore necessary to compose constraints that *count* schwas (Fig. 4).

```
define NoSchwa0 ~$[ schwa ] ;
define NoSchwa1 ~[[$ schwa ]^>1] ;
define NoSchwa2 ~[[$ schwa ]^>2] ;
define NoSchwa3 ~[[$ schwa ]^>3] ;
define GRAMMAR GEN .O. NoSchwa .O. NoSchwa1 .O. NoSchwa2 .O. NoSchwa3 ;
```

Figure 4: Example XFST code illustrating constraints that can count violations.

Each of these constraints allows a specific number of schwas through, but no more. This allows us to capture the idea that violable constraints are sensitive to the number of violations; we can picture this implementation as a decomposition of the tableau on the left of Fig. 5 to the tableau on the right.

	/ƛtmɬ/	NoSchwa
☞	[ƛətəmɬ]	**
	[ƛətəməɬ]	***

	/ƛtmɬ/	NoSchwa0	NoSchwa1	NoSchwa2	NoSchwa3
☞	[ƛətəmɬ]	*	*	*	
	[ƛətəməɬ]	*	*	*	*

Figure 5: Left: An Optimality-Theoretic tableau illustrating how a form with fewer schwas is preferred over a form with more. Right: A representation of the actual implementation of this constraint in a finite-state system. The pointing finger indicates the "winning" candidate: the form that remains when all other forms have been excluded due to violating a higher-ranked constraint.

Since this is largely boilerplate code, we automate it by defining a slightly higher-level language in Python (e.g., a macro-style function `constrain("schwa", max=3)`) and then transpile that to code like that in Fig. 4.

6 Experiment and Results

In this paper, we evaluate the phonological transducer by considering whether it can generate the attested surface forms in a corpus that contains both surface (e.g. ƛətəmɬ) and proposed underlying (e.g. ƛtmɬ) forms.

We are primarily interested in recall here (how many of the attested surface forms the system can generate), but since the underlying-to-surface relationship in this corpus is one-to-many (there are multiple valid ways to transcribe a given form[6]), we also report precision and F1, as an attempt to avoid overgenerating and producing unattested surface forms.[7]

[6]There is little valid ambiguity, however, in how a surface form corresponds to an underlying form. There are many instances where the underlying form is *unclear* due to our incomplete understanding of the phonology, and many instances where the corpus happens to be inconsistent in how it presents underlying forms, but there are few if any instances in which different underlying forms happen to be pronounced identically.

[7]Although this transducer is scored on a somewhat "canned" dataset, it is still not possible to achieve 1.0 precision and F1; leaving aside errors in the corpus and loanwords that do not follow Kwak'wala phonology, there is genuine variation in

The documents in the corpus are divided into a 75% development set (on which we did error analysis while writing the grammar) and a 25% test set (which we did not look at), to test whether the rules proposed to handle the development set generalize to unseen documents.

6.1 Corpus

	Development set	Test set
Documents	230	77
Word types	4575	1490
Word tokens	12610	3317

Table 1: Document, type, and token counts for the development and test sets

This transducer was tested on our own fieldwork corpus, currently part of the private archives at the Whatcom Museum in Bellingham, WA, representing field interviews with eight speakers of Kwak'wala.

Each word in the corpus is given a proposed underlying form, although there is some variation in how these forms are presented. In particular, there are cases where morphemes are or are not separated, or distinctions that are or are not made, according to the purpose of this example. In addition, there are various morphemes whose status as a suffix or enclitic is unclear, and which may at different points be analyzed as either. For this reason, we are only interested here in evaluating the "downward" direction of the transducer (that is, generating possible orthographic surface forms from underlying forms), rather than the "upward" direction (that is, parsing surface forms into proposed underlying forms); the latter represents a set of changing conventions of little current practical interest.

6.2 Results and Discussion

Figure 6 gives the recall, precision, and F1 for the baseline system and a number of improvements to the phonological rules and constraints.

While this is more detail than is typically reported for grammar development, and the particular changes are probably of interest only to Wakashanist phonologists, we thought it was illustrative to show the progression of development. In particular, it illustrates that expert system development is not always hill-climbing, and some changes cause losses that are repaired only by later development; for example, the epenthesis of schwas leads to a large precision drop due to overgeneration, but many of these forms are later avoided by allowing a more complex syllable structure.

The baseline system only removes elements like word and morpheme boundaries, and makes no further changes in between the underlying form and the surface form. As the baseline system only has a recall of 30.6%, this means that about 69.4% of Kwak'wala word tokens[8] have a more complex syllable structure or undergo some sort of phonological or morphological change before surfacing.

Beyond the baseline, additional improvements to the phonological system typically made steady improvements to recall, and had various effects (positive and negative) on precision. We generally did not take a loss of precision to be necessarily bad, as typically many of the new forms predicted were indeed *possible* pronunciations of words, although not attested in this small corpus; a precision loss is something to investigate further here, but not necessarily reject a rule or constraint over.

Of special note is the spirantization of /x̌/, /k/, /k^{w}/, /q/, and /q^{w}/ in syllable codas to [ɬ], [x], [x^{w}], [x̌], and [x̌^{w}] respectively. This change is nearly (but not entirely) obligatory in the speech of our consultants, but historically it was more variable. Specifying this change as optional caused a noticeable drop in precision (as many of the predicted non-spirant forms are not attested in our modern corpus), but it is still valuable to allow them given that such forms *do* occur more frequently in historical texts.

Analysis of a sample of errors suggests that most are of two types: errors in the corpus itself, and phenomena that we had inadequately annotated in underlying forms (especially which initial phonemes

pronunciation (both free variation and variation between speakers) such that not every *possible* realization of an underlying form is attested in the corpus.

[8] At the word type level, the baseline system has a recall of 14.1%, meaning that about 85.9% of word types have more complex syllable structure or phonology.

System	Development set Recall	Precision	F1	Test set Recall	Precision	F1
Baseline system	0.306	0.595	0.404	0.288	0.558	0.380
Epenthesis of schwas	0.382	0.405	0.393	0.351	0.372	0.361
Avoid final schwas	0.382	0.436	0.408	0.351	0.400	0.374
Resonant and special nuclei	0.479	0.548	0.511	0.458	0.523	0.488
Additional coda possibilities	0.523	0.607	0.562	0.492	0.571	0.528
Monophthongize /aw, ay, a̓w, a̓y/	0.525	0.604	0.562	0.494	0.564	0.527
Fortition/lenition of plain consonants	0.589	0.576	0.582	0.561	0.539	0.550
Fortition/lenition special cases	0.599	0.582	0.590	0.573	0.548	0.560
Glottalization of ʔm, ʔl	0.633	0.586	0.609	0.610	0.561	0.584
Spirantization of /x̌, k, k^w, q, q^w/ in codas	0.652	0.546	0.595	0.634	0.515	0.568
Avoiding resonant onsets	0.684	0.624	0.653	0.667	0.591	0.627
Morpheme-initial deletion	0.722	0.662	0.690	0.716	0.635	0.673
Hiatus resolution special cases	0.757	0.612	0.677	0.743	0.610	0.670

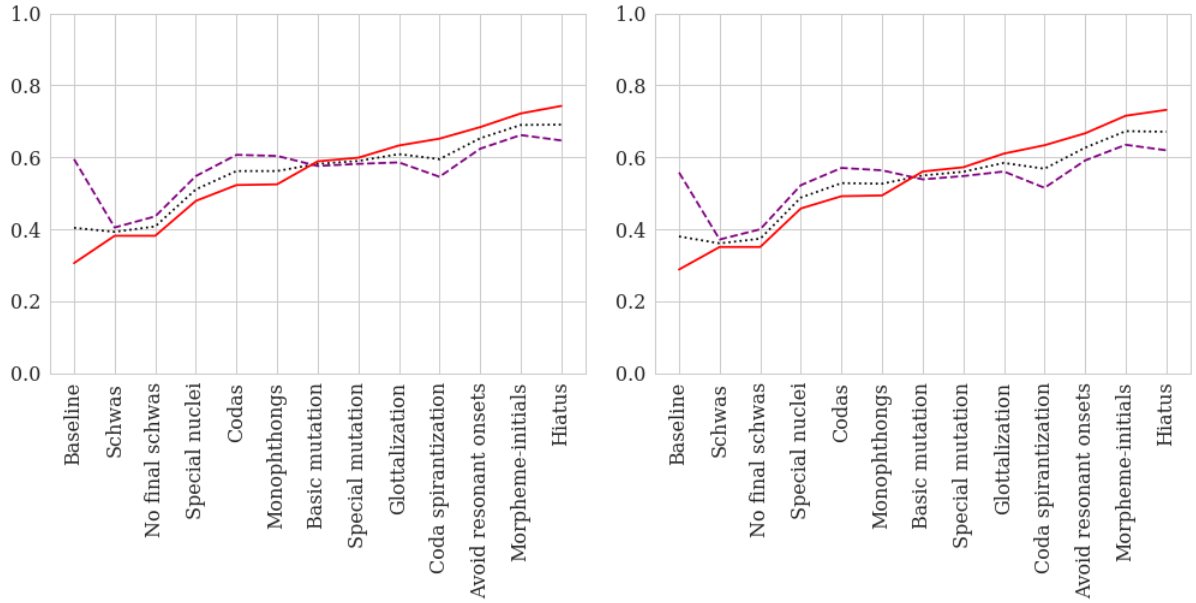


Figure 6: Improvements to recall (red solid line), precision (purple dashed line), and F1 (black dotted line) on the development (left) and test (right) documents by the implementation of specific phonological rules and constraints.

of suffixes can and cannot be dropped). We did not, however, fix any errors during the course of this experiment, so that score improvement would reflect only development effort, not re-annotation.

The remaining errors, however, suggest there are still some missing aspects of our understanding of Kwak'wala morphophonology (e.g., exactly when [ə] or [a] is inserted at morpheme boundaries) or that some of the assumptions that we had made when proposing underlying forms may be too strict (e.g. the assumption that there are only three underlying vowels). These are things we had previously suspected, but the development of an explicit computational system such as this helps to identify (and perhaps even quantify) those parts of Kwak'wala phonology and morphophonology for which our understanding is incomplete.

7 Further development

As continued development unearths an increasing percentage of errors and idiosyncrasies in the corpus itself, it may be beneficial in further development to switch to a new corpus, so that additional rules/-constraints are more likely to generalize to text by other authors, rather than overfit to our own style of

transcription and analysis.

Also, this component is only one part of the intended OCR pipeline, which will also require:

- An OCR model to extract texts in historical orthographies. Fortuitously, Hubert et al. (2016) has already trained a model to recognize historical text in Haida that used the same font and almost all the same diacritics.
- A conversion system between the historical orthography and modern orthographies. Such a system already exists at `orth.nfshost.com`, however it implements a one-to-one correspondence that would be inappropriate for this task. A many-to-many orthographic transducer may, however, be adapted using its conversion tables as a starting point.
- A morphological component specifying the known roots, suffixes, and enclitics of Kwak'wala, as well as the possible arrangements of these.

These three components, along with the phonological component here, should help to correct and normalize errors in the recognition of historical texts, and thus help make the Boas corpus more accessible to modern readers and researchers.

Acknowledgments

This research would not have been possible without my consultants' patient instruction and their many years of effort in sharing their language. Data collection was supported by the Jacobs Research Funds. Any errors or misconceptions are my own.

References

Stephen R. Anderson. 1985. Typological distinctions in word formation. In Timothy Shopen, editor, *Language Typology and Syntactic Description, Volume III: Grammatical Categories in the Lexicon*, pages 1–65. Cambridge University Press, Cambridge, UK.

Mark Baker. 1996. *The Polysynthesis Parameter*. Oxford University Press, Oxford.

Kenneth R. Beesley and Lauri Karttunen. 2003. *Finite State Morphology*. CSLI Publications.

Franz Boas and George Hunt. 1902. Kwakiutl texts. *Memoirs of the American Museum of Natural History*, 5.

Franz Boas, Helene Boas Yampolsky, and Zellig S Harris. 1947. Kwakiutl grammar with a glossary of the suffixes. *Transactions of the American Philosophical Society, New Series*, 37(3):203–377, Dec.

FirstVoices. 2009. Kwak'wala: Words. Retrieved from http://www.firstvoices.com/en/Kwakwala/words on Oct. 22, 2014.

Isabell Hubert, Antti Arppe, Jordan Lachler, and Eddie Antonio Santos. 2016. Training & quality assessment of an optical character recognition model for Northern Haida. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, May. European Language Resources Association (ELRA).

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.

Lauri Karttunen. 1998. The proper treatment of optimality theory in computational linguistics. In Lauri Karttunen and Kemal Oflazer, editors, *Proceedings of the International Workshop on Finite State Methods in Natural Language Processing (FSMNLP)*.

Donald Knuth. 1992. *Literate Programming*. California: Stanford University Center for the Study of Language and Information.

Neville J. Lincoln and John C. Rath. 1980. *North Wakashan Comparative Root List*. National Museums of Canada, Ottawa, ON.

Neville J. Lincoln and John C. Rath. 1986. *Phonology, dictionary and listing of roots and lexical derivates of the Haisla language of Kitlope and Kitimaat, B.C.* National Museums of Canada, Ottawa, ON.

Patrick Littell. 2016. *Focus, Predication, and Polarity in Kwak'wala*. Ph.D. thesis, University of British Columbia.

Mike Maxwell and Jonathan D. Amith. 2005. Language documentation: The Nahuatl grammar. In Alexander Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, pages 474–485, Berlin, Heidelberg. Springer Berlin Heidelberg.

Mike Maxwell. 2012. Electronic grammars and reproducible research. In Sebastian Nordoff, editor, *Electronic Grammaticography*, pages 207–235, Honolulu. University of Hawaii Press.

Alan Prince and Paul Smolensky. 1993. *Optimality Theory: Constraint Interaction in Generative Grammar*. Technical Report, Rutgers University Center for Cognitive Science and Computer Science Department, University of Colorado at Boulder.

Caro Struijke. 1998. Reduplicant and output TETU in Kwakwala. In H. Fukazawa, F. Morelli, C. Struijke, and Y. Su, editors, *University of Maryland Working Papers, Vol. 7: Papers in Phonology*, pages 150–178. University of Maryland Working Papers.

Caro Struijke. 2000. *Existential Faithfulness: A Study of Reduplicative TETU, Feature Movement, and Dissimilation.* Ph.D. thesis, University of Maryland.