A Finite State Approach to Interactive Transcription

William Lane and Steven Bird
Northern Institute
Charles Darwin University

Abstract

We describe a novel approach to transcribing morphologically complex, local, oral languages. The approach connects with local motivations for participating in language work which center on language learning, accessing the content of audio collections, and applying this knowledge in language revitalization and maintenance. We develop a constraint-based approach to interactive word completion, expressed using Optimality Theoretic constraints, implemented in a finite state transducer, and applied to an Indigenous language. We show that this approach suggests correct full word predictions on 57.9% of the test utterances, and correct partial word predictions on 67.5% of the test utterances. In total, 87% of the test utterances receive full or partial word suggestions which serve to guide the interactive transcription process.

1 Introduction

Thousands of the world's languages have small populations and are characterized by primary oral usage (Ong, 1982). These local languages co-exist alongside trade languages, i.e., languages of commerce, education, mass media, and government. Local languages are generally losing ground to larger languages, a process known as language shift (Fishman, 2001). Key features of local languages are that they generally have no literary tradition, and little incentive exists for writing. There is often no established or widely known orthography, and usually no widely accepted standard variety to render into writing. The point where a related dialect becomes a distinct language may not be clearly understood or widely agreed.

Many heritage communities seek to reclaim or revitalize their ancestral languages (Hinton and Hale, 2001; Grenoble and Whaley, 2006). Here, people often depend on historical sources, including informal collections of audio recordings, in order to access the ancestral code. Scholars are also involved, using historical recordings in the process of language documentation and description (Woodbury, 2003). Ideally, everything would be transcribed, and it would be easy to access the content of such collections for the purposes of learning and scholarship. However, given that these are oral languages, there is usually no pool of readily available transcribers to call upon.

None of the above is systematically addressed by current low-resource approaches to transcription, which require upwards of 100k words (or 12-27 hours) of training data in the language, in order for sufficiently accurate phone recognition to support reasonable word error rates. Such work generally assumes that a comprehensive lexicon is available, and we find that this is generally not the case.

We seek a new approach, one that works with the locally available resources and human capacities. Our work is founded on three insights. First, work on Indigenous languages proceeds from locally meaningful, locally motivated activities. This usually prioritizes content over form, interpreting over transcribing (Bouquiaux and Thomas, 1992; Wilkins, 2000). Two important use cases are language learning and accessing the content of media collections. We devise tasks that leverage informal linguistic knowledge, such as the ability to form morphotactically valid words, and specialized knowledge of the vocabulary that pertains to a semantic domain of interest. This insight does not simply connect with local motivation, it is an effective way to meet the reciprocity requirement for ethical Indigenous research (NHMRC, 2018).

Second, work with speakers of Indigenous languages is more effective when it involves collaboration on realistic tasks. Thus, we operate within the skill set and time availability of speakers and linguists. In particular, we eschew artificial tasks like phonetic transcription, instead tapping into people's ability to identify words in connected speech (Meakins et al. 2018, 230; Bird 2020). This can in-

Proceedings of the first workshop on field linguistics, pages 1–10
October 16, 2022.

volve learning vocabulary, and getting clear about nuances of word meaning by drawing on usage in context, or speech concordances. Third, we apply what is known about the language, even when it is a non machine readable grammar, by interpreting it into a computational form that can be deployed to guide language tasks.

Thus, our contribution is a novel approach to transcribing local languages that is: locally motivated, feasible, and leverages what is already known about the language. As proof of concept we provide a finite state implementation, using the framework of constraint-based Optimality Theory (Prince and Smolensky, 2004; Ellison, 1994). We envisage that this implementation, suitably optimized, could be deployed in an interactive, collaborative, sparse transcription system.

2 Background

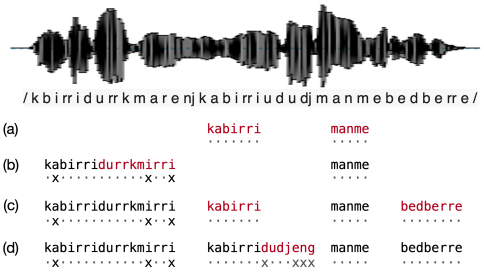
2.1 Sparse Transcription

The initial phase of working with a language – prior to having 100k words of transcribed audio – is characterized by uncertainty (Newman and Ratliff, 2001; Crowley, 2007). We have elicited enough words to establish the phonemic inventory, and transition to working with texts (Hale, 2001). However, when we listen to connected speech, we are only able to identify a few familiar words; the rest is a sea of undifferentiated speech sounds. We may attempt a transcription of those sounds, but the presence of coarticulation and disfluencies confounds our efforts to segment them and produce a contiguous transcription.

A popular solution is for linguists to delegate transcription work to literate speakers (King, 2015). However, as we have noted, for many oral languages it can be difficult to find suitable people. Instead, we may ask someone to carefully "re-speak" a recording, phrase by phrase (Woodbury, 2003, 11). Here we have found, for every place where we have conducted fieldwork, that speakers find this task immensely tedious. A solution is offered by *collaborative transcription*, where non-linguist speakers and non-speaker linguists work together.

Collaborative transcription, as we have experienced it, involves a speaker and a linguist listening to a recording, while revising a partial transcription consisting of words that the linguist has identified in connected speech. Between the identified words is unidentified material, hence the term "sparse transcription" (Bird, 2020). We illustrate this in (1), showing four iterations of linguist guesses and speaker confirmations. Not shown here is the fact that, between each iteration, we consider dozens of other utterances containing the words, and detect new, frequent words to add to our lexicon. Steps (a) and (d) may be separated in time by several days, a period during which the linguist is steadily learning to recognize a larger set of words in connected speech.

(1)



In (1), the x's indicate mismatches between phone recognizer output and the canonical transcriptions of the lexicon. These are leveraged in the optimality theoretic approach we set out below.

Sparse transcription is a shift away from current practices of transcribing phones, transcribing first, and transcribing fully (Bird, 2020). Instead, the focus is on local capacity and aspirations, and how these feed into and draw from semi-structured linguistic activities.

Sparse transcription avoids segmenting the input on the way to recognizing words; after all, hard boundaries do not exist in the speech stream (Ostendorf, 1999). A sparse transcription is represented as an audio collection, a lexicon, and a collection of tokens that pair lexical entries with locations in the speech stream. For each such token, we keep track of whether it has been confirmed by a speaker.

The ultimate aim of sparse transcription is conventional, dense transcription. However, the intermediate products are useful: a lexicon with confirmed examples from the corpus; and a corpus indexed by terms of interest. These early outputs support oral language learning and access to the content of informal audio collections.

We envisage a context where a background process, a machine in the loop, continually detects putative new tokens of words, leveraging the lexicon and the grammar, presenting them for human confirmation. We anticipate a deployment of our solution inside a collaborative transcription system, increasing the quantity and quality of transcriptions in the early, bootstrapping stage of language work.

2.2 Local Word Discovery

The new task of "local word discovery" was proposed by Lane and Bird (2021) to complement the word spotting described in section 2.1 above. They observe that, for morphologically complex languages, a lexicon consists of morphemes, not full words, avoiding the combinatoric explosion of the vocabulary. Accordingly, we spot lexemes (morphs instead of words), and just require additional computational support to expand confirmed morph tokens into full words. They provide a baseline implementation, a finite state morphological analyzer, which recognizes morphotactically valid words conditioned on a previously confirmed morph together with its left and right phonemic context (See Figure 1).

Figure 1: Local word discovery seeks to discover words at the locus of known lexemes.

The phone recognizer is not, as Bird (2020) warns, used to create output for a linguist to post-edit. Rather, it is used as an intermediate representation of the speech, guiding the local word discovery model as it generates plausible word candidates.

A weakness of local word discovery is that it requires explicit alignment of the known lexemes with the phone sequence. The present work addresses this shortcoming by proposing a finite state solution which accepts a phone string and an ordered list of known lexemes as input, and handles the alignment of lexemes to phones implicitly as it generates predictions. This solution relieves the original local word discovery algorithm of its dependency on manual alignment.

2.3 Finite State Morphological Analysis

Finite state methods remain central to computational analysis of morphologically complex languages. Beesley and Karttunen (2003) give a thorough treatment of patterns for finite state modeling of morphology. FSTs continue to play an integral role in the morphological analysis of complex languages, from field grammars (Lane and Bird, 2019) to extensive multi-year projects (Harrigan et al., 2017; Arppe et al., 2017; Schmirler et al., 2017; Snoek et al., 2014), to robust neural models trained on data generated by an FST (Schwartz et al., 2019; Moeller et al., 2018; Lane and Bird, 2020a). Over the years, several finite state toolkits have become prevalent in research, including FOMA (Hulden, 2009) and HFST (Lindén et al., 2009).

2.4 Optimality Theory

Since the 1970's it has been accepted that phonological and syntactic processes can be influenced by constraints on the output of a grammar (McCarthy, 2007). Optimality theory (OT) arose as is a framework for modeling linguistic well-formedness by maximizing the harmonization of ranked constraints (Prince and Smolensky, 2004). In short, OT provides a formalism for flexible ranked constraints on the output of a process. Model output can be optimized by ordering constraints by their relative importance.

The process works as follows. A function GEN generates all possible output candidates given a particular input, or lexical, underlying form. Then all candidates are marked for any violations of the constraints. Finally, an evaluation function EVAL filters out candidates which violate constraints. The candidates which violate the fewest high-ranking constraints are said to be the most harmonic. Suboptimal candidates are culled.

The application of OT to specific input is expressed in a *tableau*, a visual representation of generated candidates (GEN) and the selection of optimal candidates (EVAL) (see Fig. 2).

/input/	Constraint 1	Constraint 2	Constraint 3
Candidate 1	*!		
Candidate 2		*!	
→ Candidate 3			*

Figure 2: Sample OT tableau: candidates are marked for violations of constraints ranked from left to right. Candidates violating more highly-ranked constraints are rejected in favor of those which only violate lesser constraints. Chosen candidates are marked with an arrow.

In this example, some input has prompted the generation of several candidates (column 1). We also see that three constraints have been chosen and ordered according to importance, such that *Constraint 1* ≫ *Constraint 2* ≫ *Constraint 3* (row 1). The candidates are marked for violations of various constraints with asterisks (columns 2-4). To

identify the optimal candidate, we examine the violations marked in the columns from left to right. When a violation occurs, the cell is marked with an exclamation mark. So long as other, viable candidates remain, the current candidate is removed from consideration. After the EVAL process is complete, the optimal candidates are those which violated the fewest, most minimal constraints.

Ellison (1994) showed how OT can be implemented using finite state transducers, so long as three requirements are met: all constraints are binary; the output of GEN is a regular set; and all constraints are regular. For the present application, the GEN function, a morphological transducer, is regular. Equally, the transducers which count violations of phone matches can be converted into a suite of binary, regular transducers.

3 Available Resources

Low-resource languages are not necessarily understudied; many have significant description. This work benefits from an existing finite state morphological analyzer (Lane and Bird, 2019). We use it as an acceptor of morphotactically valid strings in the language, combining canonical lexemes with noisy phone recognizer output.

Additionally, it is common for linguists to maintain a bilingual lexicon, and a corpus of up to 10k words of human-transcribed speech. The computational model described in the following section incorporates these resources. We define two lexicon classes: "topical" words, semantically relevant to the audio we are transcribing, and "attested" words, those known to exist in the overall corpus.

Finally, recent advances in phone recognition have made it possible to train or fine-tune models capable of producing phone sequences from audio (Adams, 2017; Li et al., 2020). Allosaurus is a pretrained universal phone recognizer which allows for language-specific fine-tuning. We obtained the fine-tuned model of (Lane and Bird, 2021) and used it to automatically generate noisy phone sequences from field recordings of Kunwinjku speakers.

4 Joint Alignment and Local Word Discovery

The goal of the proposed local word discovery model is to give useful signal to the transcriber in the form of full word suggestions–which may be completely or partially correct–conditioned on known lexemes provided by the transcriber. Equally, we would like the model to be able to provide high confidence suggestions when possible, and back off to cast a wider net when necessary.

In this section we propose a finite state implementation of local word discovery which accomplishes this, while also incorporating implicit alignment. The GEN function takes a phone string and an ordered list of known lexemes, converts them to FSTs, and produces a list of candidate strings marked for constraint violations. The EVAL function converts these candidate strings to FSTs, and passes them through a cascade of constraints, implemented as FSTs and combined using lenient composition (see Fig. 3).

We employ three types of constraint: (a) anchored – these constraints are anchored to the beginning or end of the phone string; (b) topical – a lexical constraint consisting of words already discovered in the recording we are transcribing; (c) attested – a lexical constraint consisting of words which are attested in the language.[1]

We give more detail about the function of each of these components in the following sections. The Python implementation is available[2].

4.1 GEN

The responsibility of the GEN function is to produce a list of candidate strings which could plausibly be completions of the input anchor lexemes. In this section we describe an implementation of the different pieces of this function in detail.

Input The GEN module requires as input an ordered list of *known lexemes*, and *phone string*. Known lexemes are the anchor morphs, partial words, or full words which the transcriber has recognized in the audio. The phone string comes from access to a phone recognizer, fine-tuned for the target language. We use the Allosaurus model from (Lane and Bird, 2021), which was fine-tuned on 79 minutes of Kunwinjku field recordings and which achieved a 31.8% average word error rate in a 6-fold cross-validation.

Additionally as input we require, for each utterance, an ordered list of orthographic strings. These are the forms that have already been identified in the utterance, the "known lexemes". For example, a linguist might be able to recognize the top N most frequent morphs in the language, and write

[1] In practice, the attested constraints can be spread across multiple lexical buckets according to probability estimations.

[2] http://cdu-tell.gitlab.io/tech-resources/

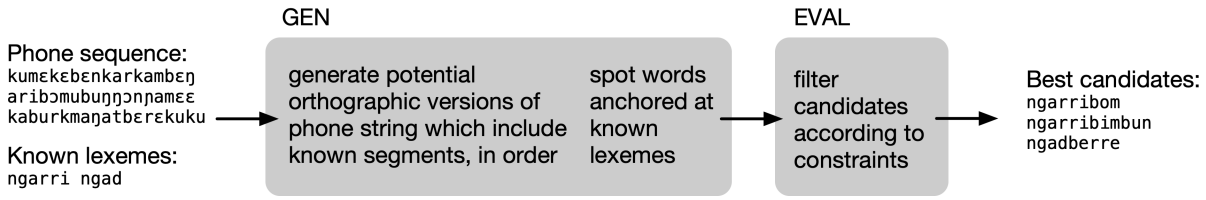


Figure 3: High-level view of local word discovery with implicit alignment.

them out in the order that they hear them in a particular utterance. See lines 1 - 6 of Figure 5 for the definition of input FSTs which would ideally be composed dynamically with the input in a real local word discovery system.

Alignment and Candidate Generation The phone input string is converted into an FST which recognizes and transduces the original string. This FST is composed with an FST which converts phones to their possible orthographic realizations. We automatically construct an FST which transduces the known lexemes into a string of the same ordered lexemes interspersed with zero or more of any character. The XFST code on line 17 of Fig. 5 defines the FST which accepts a list of lexemes and transduces all possible strings which include the lexemes interspersed with arbitrary characters.

The lower side of this relation represents the space of all possible alignments of known lexemes to the phone string. This FST can be composed with edit distance FSTs to transduce the language of candidate string alignments, allowing for alignment of known lexemes with up to N insertion/deletions of flexibility. For example, consider the utterance in (1).

In (1a), we have no known lexemes, and have just identified *kabirri* "they" and *manme* "food." In the next iteration, *kabirri* and *manme* are known lexemes, and we have identified *durrkmirri* "work." Thanks to the implicit alignment in (1b), we see that *kabirri* has been aligned to a less optimal position (requiring the insertion of *a*), in order to accommodate *durrkmirri*.

As the number of known lexemes grows, the potential number of insertions and deletions required to produce valid alignment candidates also grows. Accordingly, we allow greater edit distance for as the length and number of known lexemes grows: 1 edit per known segment of length 1-3 characters, and 2 edits for each longer segment. Our edit distance FST is a minor variation of the pattern set out by Hulden (2013) (e.g., see Fig. 5, lines 18-19).

Phone string: kbiriturkmareŋkabiriutuɟmanmɛbɛtbɛrɛ
Known lexemes: kabirri, kabirri
Gold Transcription: kabirridurrkmirri kabirridudjeng manme bedberre

	Anchored	Attested	Topical	Edit Violations
manme	*!			
→ kabirridurrkmirri^			*!	*
birridurrkmi^	*!	*	*	*
birridurrkmarrinj^^	*!	*	*	**
→ kabirridi^			*!	*
kabirridurrkma^		*!	*	*
kabirridu^^		*!	*	**
→ kabirridung^^			*!	**

Figure 4: Constraint tableau for local word discovery with implicit alignment.

After having composed the FST which accepts a phone string and known lexemes as input and transduces all possible variants of the phone string with known lexemes aligned, we compose it with a series of FSTs which recognizes and transduces any word licensed by the morphological FST, allowing any characters to the left or right. This word discovery block can also be altered to allow for some edit distance in order to widen the range of possible licensed words recognized by the morphological analyzer FST (e.g., see Fig. 5, lines 29-32).

Note that depending on which edit distance path is taken, we can append a corresponding tag (In our case the "^" character) to mark how many edit violations were required to produce a particular word candidate (See Fig. 5, lines 24-25, 30-31).

4.2 EVAL

The EVAL function filters candidates according to prioritized constraints, assuming an FST that accepts a phone string and a list of known lexemes, and produces full word candidates marked for edit distance violations (see Fig. 4).

Constraints

In optimality theory, constraints are prioritized conditions that must be maximally satisfied in order to select the optimal candidate set. Optimality theory is traditionally applied to filter for linguistic well-

```
# Set up Lexicons and Mappings as FSTs
define LEXICON [ k u m e kk e | ng a rr i b o m | n a m e kk e | ng a d b e rr e ];
define PHONESTR [k u m k ɛ b ɛ n k a r k a m b ɛ ŋ r b ɔ m ɲ a m ɛ ɛ k a b u r k m a ŋ a t b ɛ r ɛ k u k u];
define LEXEMES [X k u X ng a rr i X ng a d X];
define LEXEMESB [k u | ng a rr i | ng a d];
define TOPICAL [b i m | k u kk u | b i m b o m];
define ATTESTED [k u m e kk e | ng a rr i b o m | ng a d b e rr e];
define PHONES2ORTH  b -> [ b | bb ] .o.
                    ɲ -> [ nj ] .o.
                    ŋ -> [ ng ] .o.
                    n -> [ n ] .o.
                    t -> [ d ] .o.
                    k -> [ k | kk ] .o.
                    d -> [ t | d ] .o.
                    ...
                    i -> [ i ];

 define LexemePattern [[LEXEMES .o. [X -> ?*]].i].u;
 define Edit1 [?* [?:0|0:?|?:?-?] ?*]^<2;
 define Edit2 [?* [?:0|0:?|?:?-?] ?*]^<3;

 # GEN: Generate alignment candidates
 define OrthStrs  [PHONESTR .o. PHONES2ORTH];
 define Edit0Align [[?]* LexemePattern [?]*]["^"]*;
 define Edit1Align [[?]* [ Edit1 .o. LexemePattern] [?]*][0:"^"];
 define Edit2Align [[?]* [ Edit2 .o. LexemePattern] [?]*][0:"^"][0:"^"];
 define AlignedOrth [OrthStrs .o. [ Edit0Align | Edit1Align | Edit2Align ]];

 # GEN: Generate word candidates from alignment candidates
 define Edit0Discover [[?:0]* LEXICON [?:0]*]["^"]*;
 define Edit1Discover [[?:0]* [Edit1 .o. LEXICON] [?:0]*][0:"^"];
 define Edit2Discover [[?:0]* [Edit2 .o. LEXICON] [?:0]*][0:"^"][0:"^"];
 define DiscoverWords [AlignedOrth .o. [ Edit0Discover | Edit1Discover | Edit2Discover]];

 # EVAL: Evaluate word candidates
 define AnchoredWords [?+ LEXEMESB] | [LEXEMESB ?+] | [?+ LEXEMESB ?+];
 define TopicalWords [?* TOPICAL ?*];
 define AttestedWords [?* ATTESTED ?*];
 define edit1Words [[?-"^"]* ["^"]^<2 ];
 define edit2Words [[?-"^"]* ["^"]^<3 ];

 regex DiscoverWords  .O. AnchoredWords
                      .O. AttestedWords
                      .O. TopicalWords
                      .O. edit2Words
                      .O. edit1Words;
```

Figure 5: Minimal example of local word discovery with implicit alignment. NB `LEXEMES` and `PHONESTR` FSTs would typically be built on the fly using input to the algorithm. For this reason, our final algorithm implements the logic presented here with the HFST Python bindings, enabling parts of the network to be compiled at input time.

formedness. However, in the case of local word discovery, grammaticality is already captured by GEN, and the morphological FST. Therefore, we only need to constrain candidates on pragmatics grounds: what context can we leverage to elevate some words over the others? Through a trial and error process typical of OT, we identified the following ranking: *anchored* $\gg$ *attested* $\gg$ *topical* $\gg$ edit distance.

Anchored candidates are words which are attached to a known segment provided by the user. The model could easily hallucinate candidates across the entire phone string. However, such a broad search with loose edit distance parameters generates many spurious candidates. It is preferable to focus search on candidates for which we already have strong priors.

Attested candidates are words which are attested in some form across a wider corpus of language. We represent attested candidates which occur in a lexicon of the top $N\%$ most frequently words drawn from a corpus of public texts published in Kunwinjku: a bible translation, a set of 45 Kunwinjku children's books accessed from AIATSIS (AIATSIS Mura Collections Catalogue, 2021), and the example sentences scraped from the Kunwinjku dictionary (Bininj Kunwok Regional Language Centre, 2021). For the model evaluated in this work, we set N to 30%.

Topical candidates are words which have already been transcribed from audio related to the current audio. This lexicon grows over time, but its scope should remain topically limited to relevant themes and locations of the audio we are currently annotating. In this work, the audio we are transcribing comes from a tour of the outstation of Kabulwarnamyo. We have previous recordings which have been transcribed with other speakers giving similar

tours, and so we sample a small set of words from these to simulate a small set of 8 words to seed the *topical* lexicon.

Edit Violations are the final and lowest-priority of the constraints. Essentially, if we arrive at a set of words which are already anchored, attested, and topical, then we would further filter that result set by taking those with the least number of edit distance violations.

These constraints are operationalized in the EVAL function through the use of lenient composition, a finite state operator that allows strings to violate constraints as long as there are no other strings which do not violate that constraint. That is, a set of string candidates can be passed through a chain of leniently-composed FSTs which check for adherence to their individual constraints. At each successive state, strings which violate the constraint are filtered out, and the remaining strings are passed to the next constraint. If no more strings are able to pass a filter, then the last viable set of strings is returned as the result set (Karttunen, 1998).

5 Model Evaluation and Results

The objective of this model is to provide a reasonable set of word candidates which lead to correct transcriptions. As such, any model suggestions which correctly predict subword units beyond the anchor lexemes can be useful for helping the transcriber discern the full word they are hearing, as they interactively poll the model. Accordingly, we chose to evaluate the performance of this model by automatically simulating a first pass at transcribing 126 utterances of the test set. These 126 utterances are recorded audio segmented by breath group, from a tour of Kabulwarnamyo, conducted in Kunwinjku.

The input of the model requires a phone string, and an ordered list of known lexemes. As already mentioned, we use the Allosaurus model fine-tuned for Kunwinjku of (Lane and Bird, 2021). Similarly, we adopt their sparse transcription data preparation method: we simulate a sparse transcription of the audio by selecting a vocabulary of the top 20 morphs occurring in the training set, and use that vocabulary to manually annotate the test set. To see how this works, suppose that the test set includes an audio file (2).

(2) birri-wam balanda birri-bo-ngu-ni
they-went white.person they-liquid-eat-PI
'the white people went off drinking'

The corresponding prepared sparse transcription would be the unaligned, ordered list of morphs from the "known" vocabulary, i.e., *birri*, *balanda*, *birri*.

Using these sparse transcriptions and the automatically derived phone strings, we fed the test set to the local word discovery model to generate a list of candidate words anchored at the locus of the known lexemes. In this way, we found that 12.7% of all predictions across all utterances were correct full word predictions. Additionally, 38.2% of all model suggestions were partially correct, i.e., a substring of the suggested word attached to the anchor segment was correct, and thus a useful signal for the transcriber to decide how to continue transcribing the word (Fig. 7).

On the utterance level, 57.9% of result sets contained correct full word suggestions, and 66.7% contained correct partial word suggestions. In total, 86.5% of utterance-level result sets contained correct full or partial word suggestions. A sample of these results can be seen in Fig. 6.

6 Discussion

The key feature of this model is that we drop manual lexeme-to-phone alignment, and instead perform alignment on the fly, incorporating newly identified lexemes. This is illustrated in (1), where newly identified morphemes for each iteration are marked in red.

This innovation has an important consequence for the transcription process: it can be *iterative*. Each time the linguist and/or speaker revisit an utterance, they consider a new set of suggestions for building the transcription out from known lexemes where the model has taken care of working out where everything fits. They may also posit entirely new lexemes and add them to the lexicon. For each new lexeme, the user only needs to indicate their relative position with respect to existing lexemes.

For each new visit to an utterance, the lexicon is in an expanded state due to transcription of other utterances, and the model makes new suggestions.

The model handles some subtle issues in transcription. For example, when a morph appears multiple times in an utterance, as we see for *kabirri* "they" (1). When a transcriber adds a lexeme, the model assigns it to the best location, but only for the purpose of discovering words anchored at this lexeme (1a). When the transcriber identifies a new lexeme, e.g. *durrkmirri* "work", the previously

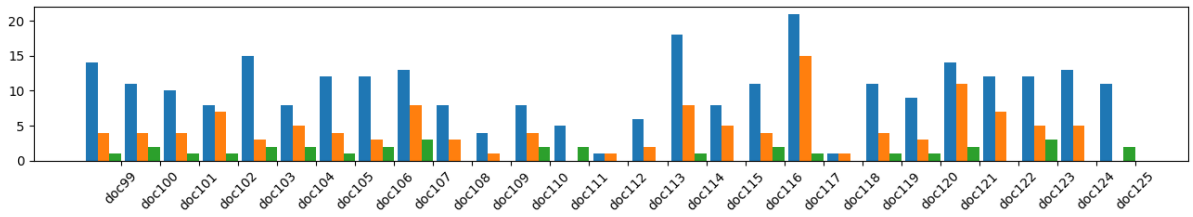


Figure 6: Sample of test set utterances with #suggestions by model (blue); #correct partial word suggestions (orange); #correct full word suggestions (green).

Number of utterances	126
% predictions full correct	12.7
% predictions partial correct	38.2
% docs with full correct	57.9
% docs with partial correct	66.7
% docs with any correct	86.5

Figure 7: LWD-A Test Set Statistics

identified lexeme is aligned elsewhere (1b). Further suggestions–by the human transcriber or an automatic word spotter–identify a second instance of *kabirri* (1c).

Alongside these benefits are some shortcomings. First, if a transcriber is mistaken about the identity of a lexeme, the model will not be able to come up with better suggestions for that locus, except in the unlikely event that there is another locus where that incorrect lexeme can be aligned. Second, the model may generate suggestions for a given anchor lexeme, when a user wants to work on a different part of the utterance. Here, the user may need to accept high priority suggestions (if they are correct) and wait for a later iteration to get model suggestions for other parts of the utterance. Third, thanks to the iterative nature of this approach, the precision and recall of the model for a given utterance depends on how high we are in the constraint hierarchy when results are returned. High priority constraints are more precise, with results sets of 1 or 2 candidates (varying on the size of the topical lexicon). Low priority constraints contain edit distance-based variations on the source signal, and therefore can grow quite large with as a function of uncertainty.

This variability with precision and recall leads to a further benefit. The model is able to prioritize precision when possible, while backing off to recall when necessary. Accordingly, for our test data, the average number of predictions per utterance is just 6.5, compared to an average of 64.1 predictions per utterance of the non-constraint based model of Lane and Bird (2021).

A further shortcoming of our approach is that we must compile unique FSTs at runtime. This means we cannot precompile the network with LEXC and FOMA or HFST compilers, but must use Python bindings, and compile FSTs dynamically with each new input. This could be prohibitively slow in some instances, as complexity increases exponentially with the size of the phone stream and known segment lists. A solution is to add a preprocessing step: utterances are already segmented from the original audio using silence; any overlong utterances are further split on confirmed full words.

7 Conclusion

We have proposed a novel approach to collaborative transcription, which works with locally available resources and human capacities. In particular, local Indigenous participation is not reduced to laborious and unmotivated phone transcription, but focuses on the identification of keywords in connected speech. These may be relevant to a concurrent cultural activity, or to language learning, in which the meaning of words in context is of more interest than their phonemic representation. The results suggest that this model does well in leveraging a computational grammar to give meaningful, interactive signal in a collaborative transcription context. This model improves on previous local word discovery models in that it is able to suggest words while performing alignment implicitly. We anticipate that this approach will integrate with interactive, collaborative transcription systems, such as (Lane and Bird, 2020b). We also hope to have shown a way of bridging language data collection to locally-motivated language work.

Ethical Considerations

This research has been approved by traditional owners in the communities where it was conducted, and the board of Warddeken, the Aboriginal land management company which hosted the research. It is covered by a research permit from the Northern Land Council and by human research ethics approval of Charles Darwin University. The authors have made several visits to an Aboriginal communities over several years, working closely with elders and traditional owners in pursuing their agenda for the future of their languages.

References

Oliver Adams. 2017. *Automatic Understanding of Unwritten Languages*. Ph.D. thesis, University of Melbourne.

AIATSIS Mura Collections Catalogue. 2021. Accessed 2021-12-10. [link].

Antti Arppe, Christopher Cox, Mans Hulden, Jordan Lachler, Sjur N Moshagen, Miikka Silfverberg, and Trond Trosterud. 2017. Computational modeling of verbs in Dene languages: The case of Tsuut'ina. In *Proceedings of the 2016 Dene Languages Conference*, pages 51–69. Alaska Native Language Center, University of Alaska, Fairbanks.

Kenneth R. Beesley and Lauri Karttunen. 2003. *Finite-State Morphology: Xerox Tools and Techniques*. Stanford: CSLI.

Bininj Kunwok Regional Language Centre. 2021. Bininj Kunwok Dictionary. njamed.com. Accessed 2021-07-19.

Steven Bird. 2020. Sparse transcription. *Computational Linguistics*, 46:713–744.

Luc Bouquiaux and Jacqueline M. C. Thomas. 1992. *Studying and describing unwritten languages*. Dallas: Summer Institute of Linguistics.

Terry Crowley. 2007. *Field Linguistics: A Beginner's Guide*. Oxford University Press.

T. Mark Ellison. 1994. Phonological derivation in Optimality Theory. In *COLING 1994 Volume 2: The 15th International Conference on Computational Linguistics*, pages 1007–1013.

Joshua A. Fishman, editor. 2001. *Can Threatened Languages be Saved?: Reversing Language Shift, Revisited: A 21st Century Perspective*. Multilingual Matters.

Lenore Grenoble and Lindsay Whaley. 2006. *Saving Languages: An Introduction to Language Revitalization*. Cambridge University Press.

Ken Hale. 2001. Ulwa (Southern Sumu): the beginnings of a language research project. In Paul Newman and Martha Ratliff, editors, *Linguistic Fieldwork*, pages 76–101. Cambridge University Press.

Atticus Harrigan, Katherine Schmirler, Antti Arppe, Lene Antonsen, Trond Trosterud, and Arok Wolvengrey. 2017. Learning from the computational modelling of Plains Cree verbs. *Morphology*, 27(4):565–598.

Leanne Hinton and Kenneth Hale, editors. 2001. *The Green Book of Language Revitalization in Practice*. Academic Press.

Mans Hulden. 2009. Foma: a finite-state compiler and library. In *Proceedings of the Demonstrations Session at EACL 2009*, pages 29–32, Athens, Greece. Association for Computational Linguistics.

Mans Hulden. 2013. Advanced finite-state techniques tutorial. http://clt.gu.se/sites/clt.gu.se/files/mkp/clttutorial.pdf. Accessed 2020-01-07.

Lauri Karttunen. 1998. The proper treatment of optimality in computational phonology. In *Finite State Methods in Natural Language Processing*.

Alexander D King. 2015. Add language documentation to any ethnographic project in six steps. *Anthropology Today*, 31:8–12.

William Lane and Steven Bird. 2019. Towards a robust morphological analyzer for kunwinjku. In *Proceedings of the The 17th Annual Workshop of the Australasian Language Technology Association*, pages 1–9, Sydney, Australia. Australasian Language Technology Association.

William Lane and Steven Bird. 2020a. Bootstrapping techniques for polysynthetic morphological analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6652–6661, Online. Association for Computational Linguistics.

William Lane and Steven Bird. 2020b. Interactive word completion for morphologically complex languages. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4600–4611, Barcelona, Spain (Online). International Committee on Computational Linguistics.

William Lane and Steven Bird. 2021. Local word discovery for interactive transcription. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2058–2067. Association for Computational Linguistics.

Xinjian Li, Siddharth Dalmia, Juncheng Li, Matthew Lee, Patrick Littell, Jiali Yao, Antonios Anastasopoulos, David R. Mortensen, Graham Neubig, Alan W Black, and Florian Metze. 2020. Universal phone recognition with a multilingual allophone system. In *Proceedings of the International Conference on*

Acoustics, Speech and Signal Processing, pages 8249–8253. IEEE.

Krister Lindén, Miikka Silfverberg, and Tommi Pirinen. 2009. Hfst tools for morphology–an efficient open-source package for construction of morphological analyzers. In *International Workshop on Systems and Frameworks for Computational Morphology*, pages 28–47. Springer.

John J McCarthy. 2007. What is optimality theory? 1. *Language and Linguistics Compass*, 1(4):260–291.

Felicity Meakins, Jenny Green, and Myfany Turpin. 2018. *Understanding Linguistic Fieldwork*. Routledge.

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, New Mexico, USA. Association for Computational Linguistics.

Paul Newman and Martha Ratliff. 2001. Introduction. In Paul Newman and Martha Ratliff, editors, *Linguistic Fieldwork*. Cambridge University Press.

NHMRC. 2018. *Ethical conduct in research with Aboriginal and Torres Strait Islander Peoples and communities: Guidelines for researchers and stakeholders*. National Health and Medical Research Council.

Walter Ong. 1982. *Orality and Literacy: The Technologizing of the Word*. Routledge.

Mari Ostendorf. 1999. Moving beyond the 'beads-on-a-string' model of speech. In *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop*, pages 79–84.

Alan Prince and Paul Smolensky. 2004. *Optimality Theory: Constraint interaction in generative grammar*. John Wiley & Sons.

Katherine Schmirler, Antti Arppe, Trond Trosterud, and Lene Antonsen. 2017. Computational modelling of Plains Cree syntax: A constraint grammar approach to verbs and arguments in a Plains Cree corpus. In *49th Algonquian Conference, Montreal, QC*.

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 Workshop on Computational Methods for Endangered Languages*, pages 87–96.

Conor Snoek, Dorothy Thunder, Kaidi Loo, Antti Arppe, Jordan Lachler, Sjur Moshagen, and Trond Trosterud. 2014. Modeling the noun morphology of Plains Cree. In *Proceedings of the 2014 Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 34–42. Association for Computational Linguistics.

David P Wilkins. 2000. Even with the best of intentions...: Some pitfalls in the fight for linguistic and cultural survival (one view of the Australian experience). *As linguas amazonicas hoje: the Amazonian languages today*, pages 61–83.

Anthony C. Woodbury. 2003. Defining documentary linguistics. *Language Documentation and Description*, 1:35–51.