A Computational Model for Interactive Transcription

William Lane, Mat Bettinson, Steven Bird
Northern Institute, Charles Darwin University

Abstract

Transcribing low resource languages can be challenging in the absence of a comprehensive lexicon and proficient transcribers. Accordingly, we seek a way to enable interactive transcription, whereby the machine amplifies human efforts. This paper presents a computational model for interactive transcription, supporting multiple modes of interactivity and increasing the likelihood of finding tasks that stimulate local participation. The approach also supports other applications which are useful in low resource contexts, including spoken document retrieval and language learning.

1 Introduction

Understanding the "transcription challenge" is a prerequisite to designing effective solutions, minimizing bottlenecks (Himmelmann, 2018). We must face realities such as the lack of a good lexicon, the short supply of transcribers, and the difficulty of engaging people in arduous work. *Sparse transcription* is an approach to transcribing speech in these low-resource situations, an approach which is well suited to places where there is limited capacity for transcription. Sparse transcription admits multi-user workflows built around shared data, for human-in-the-loop transcriptional practices, or "interactive transcription" (Bird, 2020b; Le Ferrand et al., 2020).

Sparse transcription is 'sparse' because we do not produce contiguous transcriptions up front. Instead, we transcribe what we can, and lean on computational support to amplify those efforts across the corpus. This is not suggested as an alternative to contiguous transcription, but as a more efficient way to produce it, especially in those situations where linguists and speakers are "learning to transcribe" (Bird, 2020b, page 716). Sparse transcription relies on word spotting. Wordforms that occur frequently in the transcribed portion of a corpus are used to spot forms in the untranscribed portion. These are presented for manual verification, speeding up the contiguous transcription work while indexing the entire corpus.

Sparse transcription accepts the realities of early transcription: we lack a good lexicon; we need to grow the lexicon as we go; and we do not have a ready workforce of transcribers. Moreover, in the context of language documentation, transcription is iterative and interactive. Linguists and speakers leverage complementary skills to accomplish the task (Crowley, 2007; Austin, 2007; Rice, 2009).

Sparse transcription leverages the kind of work speakers are motivated to do. For example, when it comes to recordings, speakers tend to engage with the content more than the particular form of expression (Maddieson, 2001, page 215). Identifying key words and clarifying their meanings is often more engaging than puzzling over the transcription of unclear passages (Bird, 2020b). An indexed corpus can be searched to identify additional high-value recordings for transcription.

We report on a computational model for interactive transcription in low-resource situations. We discuss the kinds of interactivity which the sparse transcription model enables, and propose an extension which provides real-time word discovery in a sparse transcription system. For concreteness we also present a user interface which provides real-time suggestions as the user enters words.

We work with speakers of Kunwinjku (ISO gup), a polysynthetic Indigenous language of northern Australia. Members of this community have expressed interest using technology to support their own language goals. Through this work we hope to support language learning and corpus indexing, and produce locally meaningful results that help to decolonize the practice of language technology (Bird, 2020a).

This paper is organized as follows. Section 2 gives an overview of the sparse transcription model. Section 3 describes a particular use case of sparse

Proceedings of the 2nd Workshop on Data Science with Human in the Loop: Language Advances, pages 105–111
June 11, 2021. ©2021 Association for Computational Linguistics

transcription: interactive transcription. In Section 4 we describe the system architecture and the design decisions which enable an interactive human-computer workflow. Section 5 describes the user interface and shows screenshots of the implementation. We conclude with a summary in Section 6.

2 The Sparse Transcription Model

Following Bird (2020b), we understand transcription to be the task of identifying meaningful units in connected speech. These units belong to a growing inventory (the glossary, or lexicon); their orthographic representation is generally not settled. We add each new meaningful unit to the glossary as it is encountered, initializing the entry with a form and a gloss. Thus, a transcriptional token is a pairing of a locus in the speech stream with a glossary entry. We are agnostic about the size of this unit; it could be a morpheme, word, or multi-word expression.

Transcription begins with a lexicon. There is always a word list, since this is what is used for establishing the distinct identity of a language. There may also be some historical transcriptions, and these words can be included in the initial lexicon. From this point on, transcription involves growing the lexicon.

The speech stream is broken up into 'breath groups' which we use as manageable chunks for transcription. In the course of transcription, it is a natural thing for a non-speaker linguist to attempt to repeat any new word and have a speaker say it correctly and give a meaning. Thus, the process is interactive in the interpersonal sense. We hear and confirm the word in context, and record it in the lexicon with a lexical identifier and a pointer to where it occurs in the media. In the background, a sparse transcription system uses this confirmed glossary entry to spot more instances.

Word spotting is an automatic task which discovers putative tokens of glossary entries. Glossary entries are already stored with pointers to occurrences in particular breath groups. Discovering new instances through word spotting then becomes a retrieval task, where each breath group is seen as a mini-document. Breath groups which are determined to contain the exemplar lexical entry are queued for speaker confirmation. Confirmed spottings are updated with pointers to their respective breath groups.

Word spotting proceeds iteratively and interactively, continually expanding the lexicon while transcribing more speech. As we focus on completing the contiguous transcription of a particular text, we grow the lexicon and the system attempts to discover other instances across the wider corpus. As the system calls our attention to untranscribed regions, which may be difficult to complete for a variety of reasons, we effectively marshall the whole corpus to help us.

A sparse transcription system is a form of computer supported collaborative work, in that it alleviates productivity bottlenecks via automation and asynchronous workflows (Greif, 1988; Hanke, 2017). The sparse transcription model—organized around a growing glossary of entries with pointers to instances in speech—can underlie a variety of special-purpose apps which support various tasks in the transcription workflow. For example, Le Ferrand et al. (2020) demonstrate the use of a word confirmation app based on word-spotted data for the purpose of confirming automatically-generated hypotheses.

We have prototyped a system which implements the core functionalities described in this section, and which includes a user interface which supports interactive transcription. Figure 2 gives a schematic view of the sparse transcription model.[1]

3 Learning to Transcribe

A linguist, learning to transcribe, is capable of listening to audio and quickly transcribing the lexemes they recognize. As lexemes are recorded, they are added to the transcriber's personal glossary. Entries in this glossary may be morphs, words, or other longer units such as multi-word expressions. The record-keeping of the glossary helps manage the linguist's uncertainty in an accountable way, as they give the task their best first-pass. As is the standard behavior in sparse transcription, a glossary is updated with links from glossary entries to the segment of audio in which they were found.

Speakers of the language can access a view of the linguist's glossary entries, and confirm entry tokens for admission to the global glossary. The design decision to maintain personal glossaries for individual users and postpone adjudication with a shared, canonical glossary is an extension of the concept defined in the sparse transcription model.

[1] The system prototype and a reference implementation of the sparse transcription model can both be found at https://cdu-tell.gitlab.io/tech-resources/.

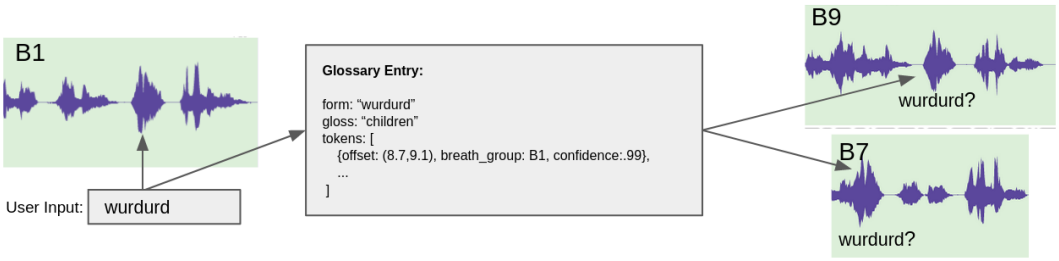

Figure 1: Word spotting in the sparse transcription model begins when the user confirms the existence of a glossary entry in the audio. A token is created for that instance of the glossary entry, and can be used to spot similar instances in other breath groups across the corpus.

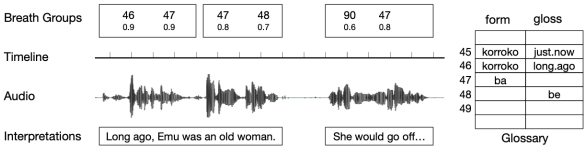

Figure 2: The Sparse Transcription Model: Audio is segmented into breath groups, each one a mini spoken document where words may be spotted (with given probability); interpretations span one or more breath groups (Bird, 2020b).

Multiple transcribers can contribute to the shared glossary, initializing their own project with the current state of the global lexicon.

Confirmed glossary entries can be used to spot similar entries across the whole corpus, maximizing the efforts of the learner, and providing more pointers from a glossary entry to breath groups where it occurs. Over time, this process leads to more contiguous transcriptions as the transcriber revisits and revises their lexicon in the course of their transcription work.

However, there is an opportunity here to get more immediate feedback from the system. A sparsely transcribed breath group (whether system or human transcribed) provides signal about the breath group as a whole. Combined with the fact that the human is currently engaged in entering their hypotheses, we can provide system suggestions conditioned on sparsely transcribed data which are updated interactively as the user types. Anchored at the locus of a known lexeme, and conditioned on additional available signal i.e., a predicted phone sequence, the system posits suggestions for untranscribed regions. We can refer to this as 'local word discovery' (Fig. 3).

Working together with the system, a linguist's hypotheses can be queued for confirmation in the same way that word spotting queues hypotheses for speaker confirmation. Simultaneously, the transcriber leverages a model to get immediate feedback on the connections between what they hear and what a model encodes about the language, potentially aiding language learning (Hermes and Engman, 2017).

Up to this point, we have established the interactive nature of transcription on three levels. First, it is interpersonally interactive, as a linguist works with speakers to associate forms with meanings. Second, sparse transcription is interactive in the sense that it attempts to amplify the effort of transcribers by propagating lexical entries across the whole corpus via word spotting.

Finally, the implementation of local word discovery is interactive in the context of the "learning to transcribe" use case. It occupies a distinct niche with a smaller feedback loop than word spotting: transcription hints are polled from the model and filtered with every keystroke (Figs. 6-8). It is improved by word spotting because contiguous transcriptions reduce uncertainty in the input to the local word discovery model. It allows a linguist to prepare and prioritize work for the interpersonally interactive task of confirming entries with a speaker.

Figure 3: Sparsely transcribed input can be leveraged for local word discovery methods which are complementary to word spotting.

4 System Architecture

The interactive transcription use case calls for a variety of computational agents. Some agents ser-

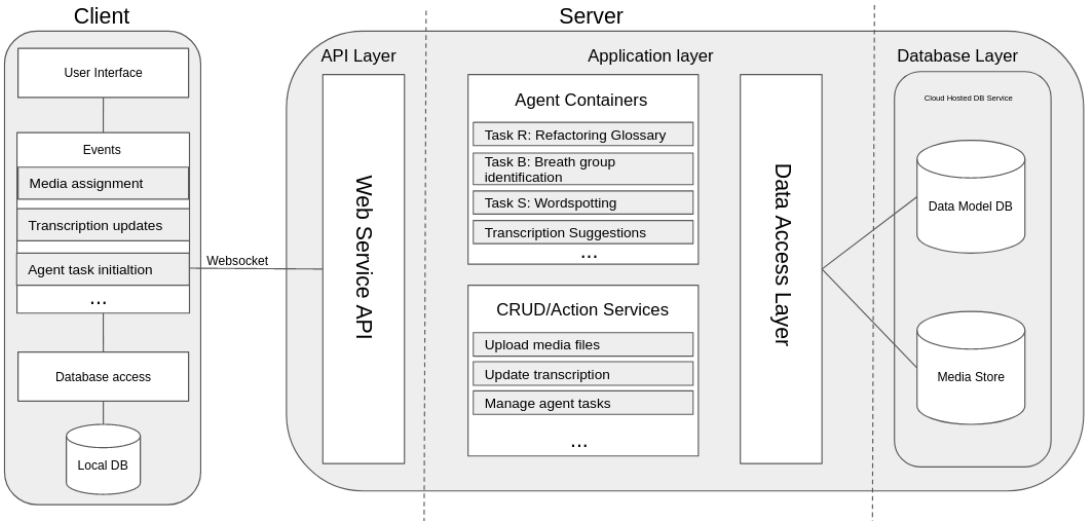

Figure 4: The system architecture

vice computationally-expensive batch tasks, while others are coupled with user events down to the level of keystrokes.

Agents are implemented as containerized services, some corresponding to long-running tasks, e.g. media processing, while others are integral to the user interface, e.g. phone alignment. The implementation supports RESTful endpoints, and a real-time websocket-based API.

The API layer responds to events in the client, and endpoints support the methods in the data model. There are three main kinds of operation; simple CRUD operations like uploading media, data model operations such as adding a token to a glossary, and real-time queries such as word discovery. Data validation is distributed across the client and the server, for performance reasons and to mitigate the effects of network dropouts. The client replicates a subset of the server data model, storing this in the browser's database and synchronizing it with the server opportunistically.

We utilise a continuous web socket session to relay user input to the server, fetching and displaying results in real time. Commonly seen in web search, this is a form of distributed user interface where computational resources are distributed across platforms and architectures (Elmqvist, 2011). This is achieved via asynchronous programming with observable streams, via implementations of the Reactive X pattern for JavaScript (rxjs) on the client and Python (rxpy) on the server. Input events from the browser are filtered, debounced and piped through a websocket transport to a session handler on the back end. Similarly, components of the client subscribe to session event streams coming from the back end, such as aligning user input to a phone stream, and presenting a series of word completions.

The system makes use of several agents whose implementation may vary across contexts or evolve over time. We have implemented the following agents:

Audio pre-processing. When a user adds an audio file to a transcription project, the audio is pre-processed and we store metadata and alternative representations which are useful for downstream tasks. For example, the pipeline includes voice activity detection (VAD), which identifies breath groups. Next, we calculate peaks–acoustic amplitude values–which we use to visualize speech activity over time. Finally, the audio is resampled and sent to the phone recognition agent, and the results are displayed beneath the waveform as extra information to support transcription.

Phone recognition. Allosaurus is a universal phone recognizer trained on over 2,000 languages (Li et al., 2020). The model can be used as-is to provide phones from a universal set, or it can be fine-tuned with language specific phonemic transcriptions. The model currently we currently deploy is fine-tuned on 68 minutes of Kunwinjku speech across 5 speakers. We calculated a 25.6% phone error rate on 10 minutes of speech from a hold-out speaker.

Word spotting. Word spotting traditionally is audio exemplar matching against spans of raw audio

(Myers et al., 1980). It has been shown to be feasible in low resource scenarios using neural approaches (Menon et al., 2018b,a). Le Ferrand et al. (2020) describes several plausible speech representations suited for low-resource word spotting.

Local word discovery. This is distinct from word spotting, which locates more tokens of existing glossary entries. Local word discovery attempts to fill in untranscribed regions between existing tokens. This agent provides transcription hints via a smaller feedback loop, the third kind of interactivity discussed in Section 3. The system retrieves the potentially large set of suggested words, and filters it down interactively as the transcriber types. The model is free to favor recall, because the raw suggestions do not need to be immediately revealed.

We implement local word discovery using a finite state analyzer for Kunwinjku (Lane and Bird, 2019), modified to recognize possible word-forms given a stream of phones and the offsets of known lexemes. We use PanPhon to estimate articulatory distances between lexemes and phone subsequences to obtain rough alignments (Mortensen et al., 2016).

5 User Interface

The user interface (Fig. 5) is inspired by minimalist design, motivated by the need for an inclusive agenda in language work (cf. Hatton, 2013). In the left column is a waveform which has been automatically segmented into breath groups. Below the waveform is a map of waveform peaks, to facilitate navigation across long audio files. Useful context is also displayed, including the transcript of the preceding breath group, followed by the sequence of phones produced from the audio, with user transcriptions aligned roughly to the phone sequence. Below this is the input box, scoped to the current breath group, where users enter lexemes, with occasional suggestions offered by the local word discovery module, and which filter interactively per keystroke (Figs. 6-8).

In the right column, there is a running transcript of the audio file, with the text of the transcript for the current breath group shown in bold.

The user interface is designed to be navigable entirely through the keyboard, to support ergonomic transcription (cf. Luz et al., 2008).

6 Conclusion

Transcription is especially challenging when we lack a good lexicon and trained transcribers. Consequently, we seek to bring all available resources to bear, including the knowledge of speakers, linguists, and a system, all of whom are "learning to transcribe."

We presented a use case for interactive transcription and showed how this can be supported within the sparse transcription model. In designing and implementing a sparse transcription system for a specific use case, we elaborated on some concepts presented in (Bird, 2020b). We examined various kinds of interactivity in low-resource language transcription, and we proposed local word discovery as a grammatically-informed approach to word spotting. This allows individual users to manage their local lexicon independently of the task of curating a canonical lexicon, enabling multi-user workflows.

Finally, we reported on the architecture and implementation of an interactive transcription system. It enables a transcriber to take care of much of the arduous transcription task up front, and to allocate more meaningful work for speakers. The product of interaction with the system is an expanded lexicon, which can be used to index the corpus for information retrieval, thus supporting the community goal of access to knowledge locked up in many hours of recorded audio. Additionally, we anticipate that support for growing personal lexicons will be a valuable resource for the language learning that takes place alongside transcription. In short, the system is designed to produce the content that language communities care about, in a way that leverages the kind of language work that people are willing to do.

Operationalizing the sparse transcription model makes it possible to streamline field-based transcriptional practices, and is expected to lead to further implementations of special purpose interfaces that support transcription of low-resource languages.

Acknowledgments

We are grateful for the support of the Warddeken Rangers of West Arnhem. This work was covered by a research permit from the Northern Land Council, and was sponsored by the Australian government through a PhD scholarship, and grants from the Australian Research Council and the Indigenous Language and Arts Program.

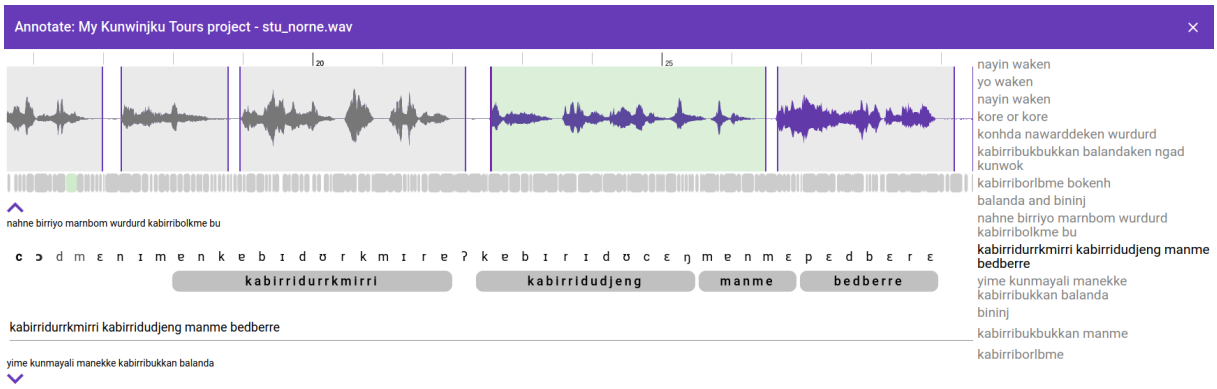

Figure 5: The transcription user interface connects to the data model, which facilitates word spotting and local word discovery agents.

Figure 6: Local word discovery predicts possible words in the audio, conditioned on known lexemes and a flexible interpretation of the surrounding sounds.

Figure 7: As the user continues typing, the list of suggestions is filtered down to those which remain compatible.

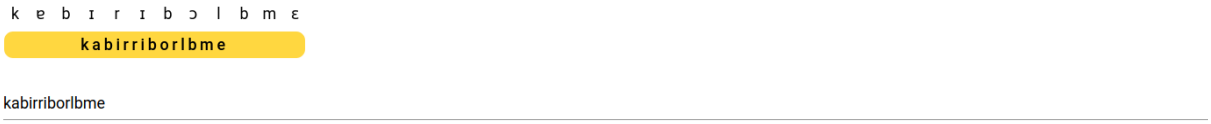

Figure 8: Thus, the user is guided to grammatically valid transcriptions which can be added to their lexicon.

References

Peter Austin. 2007. Training for language documentation: Experiences at the School of Oriental and African Studies. In Victoria Rau and Margaret Florey, editors, *Documenting and Revitalizing Austronesian Languages*, number 1 in Language Documentation and Conservation Special Issue, pages 25–41. University of Hawai'i Press.

Steven Bird. 2020a. Decolonising speech and language technology. In *Proceedings of the 28th International Conference on Computational Linguistics*, page 3504–19, Barcelona, Spain.

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

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

Niklas Elmqvist. 2011. Distributed user interfaces: State of the art. In *Distributed User Interfaces*, pages 1–12. Springer.

Irene Greif. 1988. *Computer-Supported Cooperative Work: A Book of Readings*. Morgan Kaufmann.

Florian Hanke. 2017. *Computer-Supported Cooperative Language Documentation*. Ph.D. thesis, University of Melbourne.

John Hatton. 2013. SayMore: Language documentation productivity. Presentation at International Conference Language Documentation and Conservation.

Mary Hermes and Mel Engman. 2017. Resounding the clarion call: Indigenous language learners and documentation. *Language Documentation and Description*, 14:59–87.

Nikolaus P Himmelmann. 2018. Meeting the transcription challenge. In *Reflections on Language Documentation 20 Years after Himmelmann 1998*, volume 15 of *Language Documentation and Conservation Special Publication*, pages 33–40. University of Hawai'i Press.

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

Éric Le Ferrand, Steven Bird, and Laurent Besacier. 2020. Enabling interactive transcription in an Indigenous community. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3422–28. International Committee on Computational Linguistics.

Xinjian Li, Siddharth Dalmia, Juncheng Li, Matthew Lee, Patrick Littell, Jiali Yao, Antonios Anastasopoulos, David R Mortensen, Graham Neubig, Alan 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 8249–53. IEEE.

Saturnino Luz, Masood Masoodian, Bill Rogers, and Chris Deering. 2008. Interface design strategies for computer-assisted speech transcription. In *Proceedings of the 20th Australasian Conference on Computer-Human Interaction: Designing for Habitus and Habitat*, pages 203–10.

Ian Maddieson. 2001. Phonetic fieldwork. In Paul Newman and Martha Ratcliff, editors, *Linguistic Fieldwork*, pages 211–229. Cambridge University Press.

Raghav Menon, Herman Kamper, John Quinn, and Thomas Niesler. 2018a. Fast ASR-free and almost zero-resource keyword spotting using DTW and CNNs for humanitarian monitoring. In *Interspeech*, pages 3475–79.

Raghav Menon, Herman Kamper, Emre Yilmaz, John Quinn, and Thomas Niesler. 2018b. ASR-free CNN-DTW keyword spotting using multilingual bottleneck features for almost zero-resource languages. In *Proceedings of the 6th International Workshop on Spoken Language Technologies for Under-Resourced Languages*, pages 182–186.

David R. Mortensen, Patrick Littell, Akash Bharadwaj, Kartik Goyal, Chris Dyer, and Lori Levin. 2016. Panphon: A resource for mapping IPA segments to articulatory feature vectors. In *Proceedings of the 26th International Conference on Computational Linguistics*, pages 3475–84. Association for Computational Linguistics.

Cory Myers, Lawrence Rabiner, and Andrew Rosenberg. 1980. An investigation of the use of dynamic time warping for word spotting and connected speech recognition. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, volume 5, pages 173–177. IEEE.

Keren Rice. 2009. Must there be two solitudes? Language activists and linguists working together. In Jon Reyhner and Louise Lockhard, editors, *Indigenous language revitalization: Encouragement, guidance, and lessons learned*, pages 37–59. Northern Arizona University.