

GailBot: An automatic transcription system for Conversation Analysis

Muhammad Umair

Tufts University, Medford, MA, 02155

MUHAMMAD.UMAIR@TUFTS.EDU

Julia Mertens

Tufts University, Medford, MA, 02155

JULIA.MERTENS@TUFTS.EDU

Saul Albert

Loughborough University, Loughborough, United Kingdom

S.B.ALBERT@LBORO.AC.UK

Jan P. de Ruiter

Tufts University, Medford, MA, 02155

JP.DERUITER@TUFTS.EDU

Editor: Kallirroi Georgila

Submitted 09/2020; Accepted 04/2022; Published online 04/2022

Abstract

Researchers studying human interaction, such as conversation analysts, psychologists, and linguists, all rely on detailed transcriptions of language use. Ideally, these should include so-called *paralinguistic* features of talk, such as overlaps, prosody, and intonation, as they convey important information. However, creating conversational transcripts that include these features by hand requires substantial amounts of time by trained transcribers. There are currently no Speech to Text (STT) systems that are able to integrate these features in the generated transcript. To reduce the resources needed to create detailed conversation transcripts that include representation of paralinguistic features, we developed a program called *GailBot*. GailBot combines STT services with plugins to automatically generate first drafts of transcripts that largely follow the transcription standards common in the field of Conversation Analysis. It also enables researchers to add new plugins to transcribe additional features, or to improve the plugins it currently uses. We describe GailBot's architecture and its use of computational heuristics and machine learning. We also evaluate its output in relation to transcripts produced by both human transcribers and comparable automated transcription systems. We argue that despite its limitations, GailBot represents a substantial improvement over existing dialogue transcription software.

Keywords: Automated Transcription, Conversation Analysis, Natural Language Processing

1. Introduction

Researchers studying language and communication need to analyze the speech of participants in naturally occurring dialogue. Transcribing dialogue is usually the first step in analyzing verbal interaction. However, even creating basic 'verbatim' transcripts is notoriously painstaking and time-consuming (Tilley, 2003; Lapadat and Lindsay, 1999; Boyce and Neale, 2006). For example, Saon et al. (2017) took 12-14 times the duration of each recording to transcribe. As a workaround, re-

searchers can crowd-source (Novotney and Callison-Burch, 2010), automate (Bokhove and Downey, 2018), or eliminate (Stonehouse, 2019) costly parts of the data preparation process. Alternatively, researchers can hire professional transcribers, at a rate between \$0.75 and \$1.50 per minute of audio recording to produce verbatim transcripts containing words and timestamps.

Recently, researchers have started to use Automated Speech Recognition (ASR) systems to produce ‘good enough’ first draft transcripts (Bokhove and Downey, 2018). Since the 1980s, NLP researchers have developed and refined ASR technology, including speech-to-text (STT) systems (Juang and Rabiner, 2005; Moore, 2015). STT systems recognize word orthography and timing in audio streams, and are used in prominent applications (e.g., dictation systems, voice assistants, automated captioning, etc.). STT systems are between twenty to forty times cheaper than hiring professional transcribers. For example, at the moment of writing, Google’s cloud STT service costs \$0.036 per minute of audio and IBM Watson’s STT service costs \$0.02 per minute of audio. For data recorded in ideal conditions, these systems can achieve close to human level accuracy in recognizing words. IBM’s Watson can achieve a Word Error Rate (WER) of 5.5%-11% compared to a 5.1%-6.8% WER for human transcribers on the same data (Saon et al., 2017).

However, even if automated STT can approximate human-level word recognition and create ‘good enough’ first draft transcripts, it cannot identify *paralinguistic* features of speech and integrate them into the transcription (Moore, 2015). These are non-lexical aspects of speech, like volume, voice quality, intonation, and laughter, which carry strong interactive meaning. Transcribers using conversation-analytic transcription use what is often called *Jeffersonian* notation, which uses standard keyboard symbols¹ to visually represent not only the words, but also paralinguistic features in the speech (Jefferson, 2004a; Hepburn and Varney, 2013; Hepburn and Bolden, 2017). For brevity, we will refer to transcripts that use Jeffersonian notation as *CA transcripts*.

Excerpt 1: Machine transcription from HappyScribe, a commercial transcription service.

1. [00:00:00.000] - Speaker 1 It also makes me want to,
2. like, have, like, hot chocolate and like it from a
3. fireplace.
4. [00:00:05.250] - Speaker 2 Oh, my God. ... There’s no
5. fires at Tufts, and it really pisses me off.
6. [00:00:21.230] - Speaker 1 There’s too many fires in
7. California.
8. [00:00:23.080] - Speaker 2 No, like, fireplaces.
9. [00:00:25.060] - Speaker 1 Yeah.

Excerpt 2: Manual CA transcript for the same audio used in Excerpt 1.

1. *SP1: it also ma:kes me: w:ant to like (0.2) have like
2. (0.3) hot chocola:te >and like [sit in fron]t of

¹A summary of Jeffersonian transcription symbols is provided in Appendix A.

3. a firep[lace,<]
4. *SP2: | mm |
5. *SP2: [o:h my go]d.

6. ...
7. *SP2: [There's] no fires: at Tufts: and it really pisses
8. me of[f.]
9. *SP1: [Th]ere's too many fir:es in California,
10. (0.5)
11. *SP2: No like fire like firepla[ces]
12. *SP1: | oh | yeah,

Excerpt 1 shows a transcript generated using an online STT service. Excerpt 2 shows a manual CA transcript of the same audio. The paralinguistic features in Excerpt 2 affect the interpretation of the sequence. First, Excerpt 1 does not identify the non-lexical vocalization (“mm”) represented in line 4 of Excerpt 2. With this vocalization, also called a ‘continuer’ (Schegloff, 1982), SP2 signals that they have understood SP1 so far, and that SP1 can continue speaking. Second, changes in syllable rate (see Section 2.2.4) are not annotated in Excerpt 1 but are marked by angled brackets in Excerpt 2 (lines 2-3). Here, SP1 uses faster-than-normal speech to extend their turn, which could have been expected to be complete on syntactical grounds after “chocolate”, before SP2 can interject. Third, overlaps are not annotated in Excerpt 1, whereas Excerpt 2 shows the exact temporal order of events. For example, SP2’s minimal uptake (“mm”) is a reaction to SP1’s mention of hot chocolate, and not the fireplace. Later, SP2 produces “Oh my God,” on line 4 of Excerpt 1, in ‘last-item onset’ overlap (Drew, 2009). This shows that SP2 reacts enthusiastically to SP1’s mention of a fireplace. The precise temporal order of the utterances is not clear without the overlap markers. Fourth, silences, represented as the number of seconds between brackets in the manual transcript, are not represented in Excerpt 1. SP2 uses “fires” to refer to fireplaces, but on line 8, SP1 uses “fires” to refer to forest fires. Next is a 500 ms gap, which is longer than the average turn transition (Stivers et al., 2009). This gap projects a socially ‘dispreferred’ response (Pillet-Shore, 2017), in this case a correction of SP1’s use of “fires”.

Even this cursory comparison shows that CA transcripts provide information vital to interpreting verbal interaction, but that is absent in standard STT transcripts. However, while there are corpora with audio, words, timestamps and other useful annotation layers such as part-of-speech (POS) tags (e.g., Godfrey et al., 1992), there are no large-scale corpora of CA transcripts. In part, this is because CA transcription takes a lot of time, even for experts. Experienced CA transcribers usually estimate that one minute of dyadic conversation, with high quality audio, takes around an hour to transcribe in CA format (Wagner, 2020). Instead of transcribing entire corpora with CA notation, researchers often first produce a first draft transcript and then transcribe segments of interest in greater detail (Heath et al., 2010). Automating first draft CA transcription will increase the time interaction researchers can spend on analysis as well as the amount of transcribed data they have access to.

Similarly, NLP research has been held back by its reliance on a limited number of available corpora of spoken language, such as the Switchboard corpus (Godfrey et al., 1992), which does

not include annotations of paralinguistic features of talk, such as speech rate, pauses, and pitch. Corpora of CA transcripts could be used to improve NLP-based software systems in a variety of settings. For example, call centers use NLP to help agents and assess performance (Mishne et al., 2005). Clinicians use similar tools to help diagnose patients (Mirheidari et al., 2017; Blomberg et al., 2019). Enriching the data used to create NLP models will improve all these services. To reduce the resources required to produce conversation-analytic transcripts at scale, we developed *GailBot*: the first automated transcription system designed to generate draft CA transcripts.

Moore (2015) first introduced automated transcription to CA, using IBM’s Atilla System (Soltau et al., 2010). Although this software transcribed word orthography and timing, it did not annotate paralinguistic features. *GailBot* addresses this limitation by automatically generating draft CA transcripts, using one of multiple available STT services to generate words, timestamps, and identify speakers (speaker diarization). It then employs a series of plugins that identify paralinguistic features (see Section 2.1.2). Users can extend *GailBot* with new plugins to identify new paralinguistic features and improve *GailBot*’s performance. Finally, *GailBot* outputs a first draft CA transcript, which researchers can then improve and enhance manually. Note that *GailBot* aims to *facilitate*, and not *replace*, manual transcription. Nonetheless, we suggest that *GailBot* enables researchers to produce large scale corpora containing CA transcripts, which can create opportunities for creating new interfaces between computational-linguistic and conversation-analytic approaches.

In the Architecture section, we describe *GailBot*’s internal data structures, data flow, and the algorithms used in current plugins. In the Performance section, we compare *GailBot* transcripts to manual transcripts and those used by Moore (2015), provide an estimate of the transcription time saved by using *GailBot*, and compare it to existing data annotation systems. Finally, in the Discussion section, we address the technical and theoretical limitations of *GailBot* and automated transcription in general and suggest steps for future development.

2. Architecture and Algorithms

In this section, we begin with an overview of *GailBot*’s architecture and internal data representations. We then highlight *GailBot* *plugins*, which are wrapper objects for custom algorithms that identify and annotate specific paralinguistic features. Finally, we describe and analyze these algorithms.

2.1 Architecture

2.1.1 DATA FLOW AND INTERNAL REPRESENTATIONS

GailBot implements an API that abstracts over the transcription process. Conceptually, this API has two components: an organizer and a core pipeline, connected by a controller. The organizer handles both 1) media files of conversations and 2) previous *GailBot* outputs. It determines whether the input can be processed and, if so, creates a *conversation object* for that input. A conversation object stores information required by the core pipeline to generate a transcript. Each conversation object has a *settings profile* that defines how the object is processed by the pipeline. For example, a settings profile allows the user to select which STT service to use and which plugins to apply. Once a settings profile has been created, it can be saved, re-used, modified, and applied to multiple conversation objects. However, each conversation object must have a settings profile attached before it can be processed.

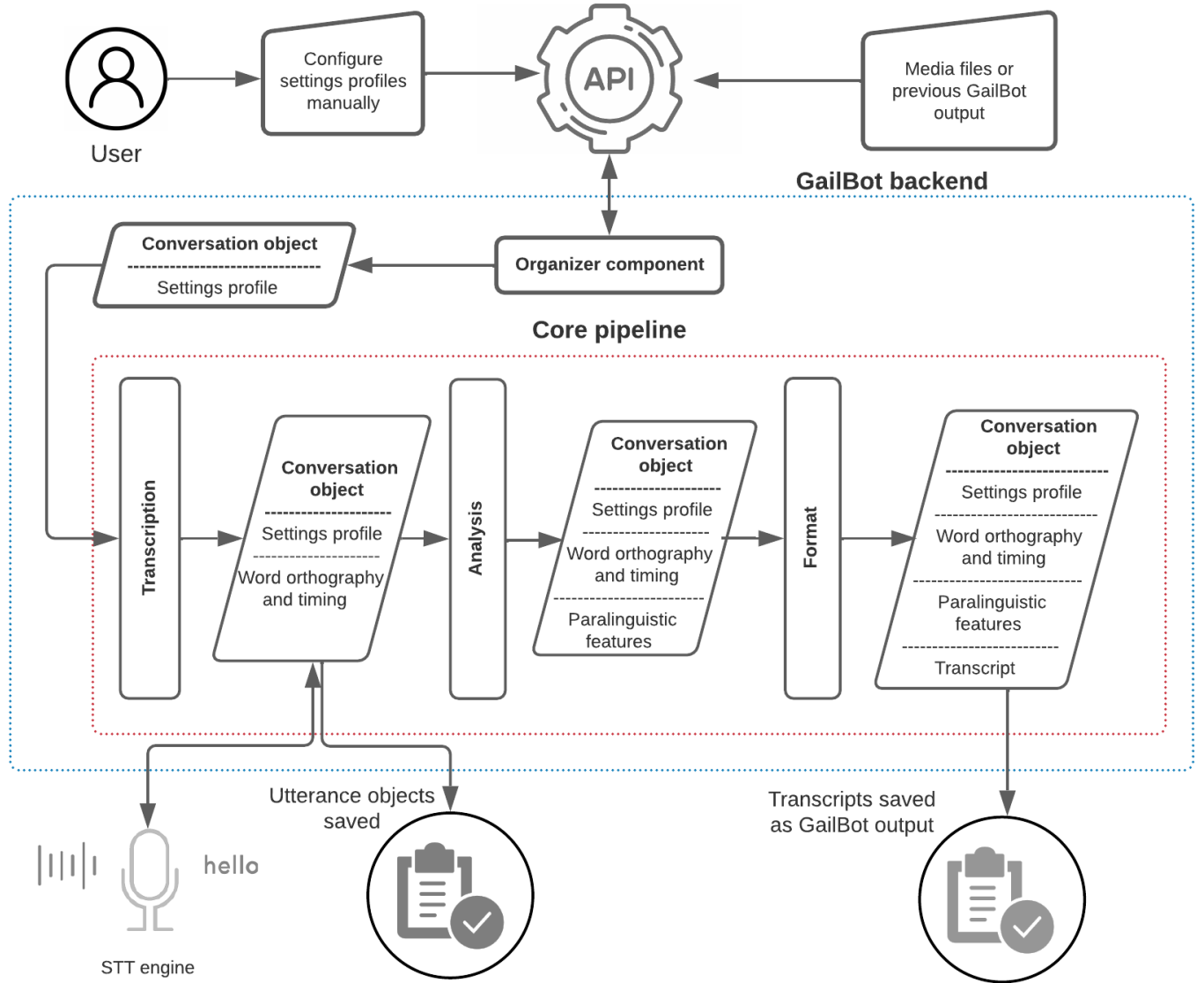


Figure 1: An overview of GailBot’s architecture and processes.

Once all conversation objects have a settings profile, the controller transfers them from the organizer to the core pipeline. The core pipeline transcribes, analyzes, and formats each conversation object in separate processes. These processes are executed sequentially, each augmenting the conversation object with information required to produce a final transcript. The transcription process sends raw audio data to one of multiple STT services, such as IBM’s Watson and Google Cloud (Soltau et al., 2010), based on the settings profile. Different STT services differ in accuracy, but they all provide the same basic functionality: they process an audio waveform, recognize words,

and return the orthographic representation and timing of each recognized word. This returned information is organized into and stored as lists of *utterance objects*, with each object containing a speaker label, start time, end time, and text. At this stage, utterance objects are saved to disk and can be reused if needed, bypassing the costly and time consuming transcription process. For example, users can apply different sets of plugins on previous GailBot outputs without executing the transcription process. Next, the analysis process executes *plugins* (see Section 2.1.2) that identify paralinguistic features of talk. Finally, the format process consolidates information from the conversation object into a CA transcript. Figure 1 visualizes this entire process.

2.1.2 PLUGINS

GailBot provides a framework for users to add custom algorithms to identify specific paralinguistic features. Plugins are wrapper objects (implemented as Python classes) that provide a standard API for these algorithms to interact with the core pipeline. This means that customizing and changing plugins does not require modifying GailBot’s source code. Plugins may or may not be dependent on each other. For example, plugin B is dependent on plugin A if the feature identified by plugin A is required as an input by plugin B. The internal plugin pipeline, executed during the analysis process, manages these dependencies. It represents each plugin as a node in a Directed Acyclic Graph (DAG) and uses edges to encode dependencies between plugins. This ensures that plugins can be executed concurrently when possible and that plugins whose dependencies fail are not executed. It also allows plugins to receive outputs from all their dependencies before they are executed. Configuration files are used to add plugins and define their dependencies in a standard format. By default, GailBot applies a few plugins that are required for most transcripts. For example, one default plugin constructs turns from word timing and orthography. Users can select plugins by configuring the settings profile of a conversation object.

2.2 Algorithms

In this subsection, we describe algorithms we developed to identify certain paralinguistic features. We highlight the relevance of each algorithm and discuss its strengths and limitations. Note that these algorithms are implemented in the current version of GailBot as plugins.

2.2.1 TURNS

Turn Construction Units (TCUs) are fragments of speech that are ‘hearably’, pragmatically, grammatically, and prosodically complete. After a TCU, speakers have the opportunity to start the next turn at a Transition Relevance Place (TRP) (Sacks et al., 1974; Clayman, 2012). Conversation analysts use TCUs and TRPs to decide whether to start a new line in a transcript, although they do so in different ways. Some transcribe each TCU on a new line, while some only create a new line when there is a long gap, and yet others only create new lines when there are speaker transitions. A common convention is to put TCUs that contain overlapping speech on their own line, with the overlapping TCU below it, in order to make the overlap clear. However, there is no general consensus among transcribers on how to split speech into different lines.

There are also some features that are only relevant at the end of TCUs, like latches or turn-final intonation (see Appendix A). ASR systems are typically limited in their ability to analyze pragmatics, grammar, and prosody to reliably infer the location of TCUs and TRPs. Therefore, GailBot exploits a normative pattern in turn-taking instead of trying to implement human-like TCU

perception: interlocutors frequently begin speaking when their social partner finishes their turn (Schegloff, 1979). The algorithm assumes that when a new speaker initiates a turn, the current turn has ended.

This method is fast and simple, but limited. The main drawback is that the current plugin cannot identify the beginnings or ends of consecutive TCUs produced by the same speaker. This means that GailBot cannot currently identify turn-final characteristics of many TCUs such as latches and turn-final intonation. In addition, when speakers overlap, GailBot cannot use TCUs to choose when to start a new line. Researchers interested in those features are encouraged to replace the algorithm used in the turn construction plugin with a more sophisticated one (e.g., Masumura et al., 2018).

2.2.2 SILENCES

Silences often have interactional significance (e.g., Pomerantz, 1984), and provide evidence for cognitive theories about communication (e.g., De Ruiter et al., 2006). There are two types of silences annotated in CA transcripts: *gaps* (silences between turns) and *pauses* (silences within turns). In this paper, silences refer to any period of non-talk.

*Excerpt 3: GailBot transcript demonstrating turn construction.*²

```

1      *SP1:  Um (.)  it's about like this (0.5) man (0.3)
           who
2          it's like (.)  he's like (.)  pilgrim (0.5) and
3          he's (.)  like
4          (2.5)
5      *SP1:  Kind of like an undercover agent (0.4) and he
6          (0.5) does (.)  is like (0.3) living out like
           (.)
7          in Europe and traveling in doing like

```

Excerpt 4: Manual transcription of Excerpt 3.

```

1      *SP1:  um (0.2) it's about like this (0.5) ma:n (0.3)
2          who it's like (0.2) he's like (0.2) pilgrim
3          (0.5) and he's (0.2) like (2.5) >kinda like<
           an
4          undercover agent (0.3) a:nd he (0.5) does (0.2)
5          is like (0.3) living out like (0.2) in Europe
6          and traveling and doing like

```

²Unless otherwise stated, data comes from the Human Interaction Laboratory's In Conversation Corpus (ICC) (see Section 3.2.3). Audio for select ICC transcripts is available at the [HI-Lab website](#).

The silence algorithm first uses word timing, regardless of speaker identity, to determine silence start and end times. Next, it uses these start and end times to calculate the duration, in seconds, of each silence. Finally, it rounds the silence duration to the nearest 100 ms and classifies it as either a micro-pause, pause, or gap. By default, the plugin classifies a silence based on CA notation guidelines. A micro-pause is a within-turn silence between 100 ms and 200 ms long. A pause is a within-turn silence between 200 ms and 1000 ms long. If a pause is greater than 1000 ms, it is considered a gap for a number of reasons. Jefferson (1983) proposed that one second is the maximum standard silence in conversation. In addition, Yang (2004) suggests that silences longer than one second are more likely to be between than within turns. In Excerpt 3, GailBot transcribed a 2500 ms silence (line 4) as a gap - separating two TCUs - instead of a silence, as transcribed in Excerpt 4, line 3. Finally, GailBot only transcribes gaps that are at least 300 ms long. Users can adjust these duration thresholds based on their preference.³ For example, a user may wish to increase the gap threshold to 2000 ms when transcribing speech produced by someone who struggles with lexical retrieval, or in situations where activities by the participants may provide an account for longer silences (Jefferson, 1983).

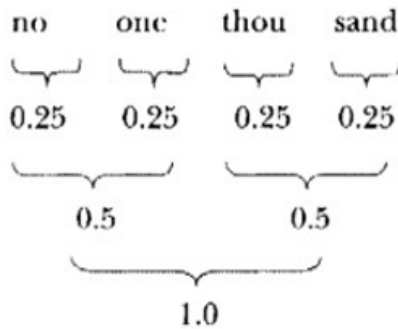


Figure 2: Example of Beat Timing. (Liddicoat, 2007)

How interlocutors perceive silence can depend on how a speaker talks (Hepburn and Bolden, 2017: 26-27). If a speaker talks slower than usual, a long silence may feel short; if a speaker speaks faster than usual, the same silence may feel longer. Therefore, some researchers transcribe silences in *beats* instead of seconds. GailBot has two modes, the ‘absolute’ and ‘beat’ timing modes, to approximate each method. In ‘absolute’ mode, a silence is the time between the end of one word and the start of the next. In contrast, ‘beat mode’ takes the syllable rate into account. CA transcribers estimate the number of beats by counting “one one thousand, two one thousand, ...” while adopting the same speech rhythm as that in the transcribed conversation (Jefferson, 1983). Because “one one thousand” represents one unit of time, and has four syllables, a syllable equals 0.25 beats (see Figure 2, taken from Liddicoat, 2007: 31). To estimate beats, this algorithm first calculates the syllable rate (see Section 2.2.4) for each speaker individually. Next, it uses the syllable rate to determine the number of syllables that fit each silence. This value is divided by 4 to obtain the final number of ‘beats’. By default, GailBot uses the absolute timing mode, but the user can choose to have GailBot use beat mode instead if desired.

³For all heuristics and thresholds, see Appendix B.

2.2.3 OVERLAPPING SPEECH

Interlocutors can start turns at the same time, interrupt each other, or start turns before another speaker has finished (Roger et al., 1988; Drew, 2009; Jefferson, 2004b). These overlaps affect the sequential ordering of actions. The overlap algorithm⁴ uses speaker identity and utterance timing to identify moments where speakers overlap. Excerpts 5 and 6 highlight the differences in overlap annotations between GailBot and a manual transcript. In Excerpt 5, the algorithm places overlap markers at the start and end of word. In some cases, this strategy can reduce the accuracy of the overlap markers, especially when a very long word is overlapped.

Excerpt 5: GailBot transcript demonstrating overlap markers.

```

1      *SP2:  Uhm so this semester we have some new members
2              and the total number would be like (0.5) about
3              twenty I [  think  ]
4      *SP1:              [Oh that's] that's a nice (.)
5              number

```

Excerpt 6: Manual transcription of Excerpt 5.

```

1      *SP2:  um:  so this semester:  we have some new members
2              and the total number would be like (0.5) about
3              twenty I thin[k ]
4      *SP1:              [Oh] that's that's a
5              nice (.)  number

```

2.2.4 SYLLABLE RATE

Interlocutors can speak at different rates, and speak faster or slower to convey information (e.g., display urgency) or perform actions (e.g., compete for turn space; French and Local, 1983). Transcribers represent meaningful changes in speech rate by annotating speech that is faster or slower than normal, as well as syllables that are ‘stretched’.

Excerpt 7: GailBot transcript demonstrating the syllable rate algorithm.

```

1      *SP2:  Yeah we will perform in the parade of nations and
2              like we will have our own show case maybe next
3              semester (0.3) >usually in the spring semester<
4              (0.4)
5      *SP1:  Wow that's awesome how many people are in that

```

⁴The overlap algorithm is described in Appendix C.

Excerpt 8: Manual transcript of Excerpt 7.

```

1      *SP2:  yeah we will perform in the parade of nations
2              and like we will have our own showcase maybe
3              next semester (0.4) >usually in the spring
4              semester<
5              (0.4)
6      *SP1:  wow that's awesome how many people are in that
    
```

The syllable rate algorithm works by identifying outliers on the segment level. Segments are defined as speech surrounded by silences of at least 100 ms. GailBot estimates the number of syllables in each segment using the Big Phoney Python package (Epp, 2018), and divides the result by the duration of each turn (see Section 2.2.1). For each speaker, the algorithm calculates the median absolute deviation (MAD) (see Equation 1), a measure of variance that is robust to outliers (Leys et al., 2013). By default, turns with syllable rate two MAD above or below the median are considered fast or slow speech. For example, in Excerpt 7, faster than normal speech is annotated on line 3. Excerpt 8 is a manual transcript for the same audio.

$$MAD = Median(|X_{si} - \tilde{X}_s|) \quad (1)$$

X_{si} = Segment syllable rate

\tilde{X}_s = Speaker syllable rate

Additionally, the algorithm identifies some stretched syllables. It measures how much slower a word is than normal by using the median syllable rate for the specific speaker. One marker is inserted for every MAD the syllable rate is below the median. The sound-stretch markers are added after the last vowel in the word. Currently, this addition position is arbitrary as we do not yet have access to algorithms that determine which specific phonemes are elongated within a word.

2.2.5 LAUGHTER

Laughter, like most non-lexical vocalizations, provides an important resource for participants in interaction (Keevallik and Ogden, 2020) and is always annotated in CA transcripts. These representations describe generic laughter as well as sophisticated inbreaths, outbreaths, pulses, and other vocal characteristics of laughter (Hepburn and Varney, 2013). The current laughter algorithm, inspired by Ryokai et al. (2018); Abadi et al. (2016), identifies the start and end times of laughter. It starts by removing background noise using Google’s Voice Activity Detector. Next, it uses a three-layer feed-forward neural network. The first two layers perform batch normalization, dropout (to prevent overfitting), and use RELU as the activation function. The final layer is a dense layer with a Sigmoid activation function. Given a target frame and standard audio features (MFCCs and delta-MFCCs), the network produces laughter probabilities for all 10 ms segments in the audio. The network is trained on the Switchboard corpus (Godfrey et al., 1992) and achieves an 88% per-frame accuracy on a validation set (Ryokai et al., 2018; Abadi et al., 2016). Finally, the algorithm uses a low-pass filter to segment the full duration of a laugh from the recording. Currently, this algorithm cannot determine more granular vocal components of laughter (e.g., inhalations, exhalations

etc.). Instead, it identifies complete laughter segments and leaves it to the human transcriber to add interactionally relevant levels of detail.

3. Performance

In this section, similar to Moore (2015), we evaluate GailBot’s performance on conversation from different corpora. The corpora differ in their sampling rate, audio separation, and background noise. The Newport Beach corpus has a sampling rate of 44.1 KHz, does not have speaker audio separation, and has background noise. The Callhome corpus has speaker audio separation, an 8KHz sampling rate, and background noises. The In Conversation Corpus has audio separation, no background noise, and a high sampling rate of 48 KHz. We compare GailBot’s performance on these data to manual and automated transcripts. We also provide the results of a brief experiment estimating the amount of time saved when using GailBot compared to transcribing from scratch. Finally, we compare GailBot with existing data annotation systems.

3.1 Metrics

3.1.1 WORD ERROR RATE

To be useful, ASR technology must correctly identify most words in an audio stream. The Word Error Rate (WER), defined by Equation 2, is one measure of the accuracy of a speech to text system, defined as the number of errors made by the service relative to the total number of words it attempts to recognize. It accounts for three types of errors: additions, deletions, and substitutions. Addition occurs when the system identifies a word that is not in the audio stream. A deletion occurs when the system does not identify a word. Finally, a substitution occurs when the system replaces one word with another. Note that the WER can be over 100% in cases where more words are misidentified than actually exist. In this paper, we use two types of WERs: a strict WER (SWER) and a relaxed WER (RWER), both computed by manually comparing GailBot transcripts to human-produced transcripts.

$$WER = \frac{(\text{additions} + \text{deletions} + \text{substitutions})}{\text{words in transcript}} \times 100 \quad (2)$$

3.1.2 RELAXED WORD ERRORS

RWER is calculated with the same equation as WER, except it does not count vocalizations with no dictionary spelling (e.g., “um” vs. “uhm”) as errors. In addition, RWER does not count substitutions for phonetically identical words (e.g., “for” vs. “four”) as errors, because it is very difficult for most STT systems to tell the difference between the two. We present this metric along with SWER to provide a more forgiving metric and a range of expected performance for GailBot.

3.1.3 STRICT WORD ERRORS

The Strict Word Error Rate (SWER) also uses the same equation as WER, but considers any differences between the GailBot produced and manually produced transcript as errors. For example, SWER considers substitutions for phonetically identical words (e.g., “for” vs. “four”) as errors. Therefore, the SWER is either higher than or equal to the RWER.

3.1.4 OVERLAP AND SILENCE PERFORMANCE

Overlaps and silence errors can occur when GailBot and a human transcriber either 1) disagree on the *presence* of a silence or overlap or 2) agree on the presence but disagree on the *duration* of the overlap or silence. Depending on the researcher’s goals, small differences in the magnitude of an overlap or gap may not be considered significant. However, it is important to indicate whether a gap or overlap exists, as it informs readers regarding the sequence of the conversation.

GailBot silence and overlap identification is influenced by the accuracy of word recognition preformed by the STT service. If GailBot cannot recognize a word, then it assumes the word is silence. This may occur if the speaker whispers, uses ‘creaky voice’, or uses a word not contained in the STT service’s vocabulary. As a consequence, GailBot will mark a silence that is not there, and might not notice an overlap. Alternatively, GailBot may transcribe a breath or other noise as a word. This will eliminate a true silence, or cause GailBot to erroneously identify an overlap when it does not exist. We did not want to penalize the overlap and silence algorithms for an error caused by the STT service. Therefore, we do not reduce the silence or overlap accuracy for errors caused by inaccurate word recognition.

3.1.5 SYLLABLE RATE PERFORMANCE

Words, overlaps, and silences can be measured relatively objectively and have standard annotations in CA transcripts. However, there is no clear metric to determine whether a speaker’s turn should be considered ‘fast speech’. Therefore, while we present examples of syllable rate annotations in this paper, we do not use a quantitative measure for syllable rate accuracy.

This module is relatively robust to word recognition errors for a few reasons. If GailBot does not recognize a word and decides it is silence, the syllable rate algorithm splits the speech into two segments (instead of calculating a very slow speech rate). Further, by default, GailBot identifies segments with speech rate 2 median absolute deviations away from the median. In practice, this includes relatively few turns. Therefore, one additional word is unlikely to cause a ‘normal’ turn to be misidentified as a ‘fast’ turn. Instead, performance is affected by exactly how researchers analyze speech rate. It’s likely that transcribers consider speech rate changes based on the most immediate context, while GailBot considers speech rate changes based on the entire conversation (including segments after the target segment). As we learn more about how listeners perceive changes in syllable rates, this algorithm may be refined and better evaluated.

3.1.6 TURN TAKING PERFORMANCE

Transcribers differ in their representations of the ordering of talk. For example, some transcribers may want each TCU on a separate line. Others may split consecutive TCUs by the same speaker only when there is a long pause. Some may even choose to separate TCUs only when another speaker begins speaking. Because there is no objective standard for annotating turn structure, we do not evaluate how well GailBot splits turns into lines. However, our subjective impression, based on having used GailBot extensively for several years, is that although it is far from perfect, its annotations of turn structure are adequate for generating a useful first pass.

3.1.7 LAUGHTER PERFORMANCE

We could not perform a conclusive evaluation of the laughter detection algorithm because we did not have enough unique instances of laughter in our data to analyze. In addition, GailBot does not yet attempt to represent laughter in CA format, which would require a representation of each separate laughter particle, including for instances where the speaker produces a laugh particle mid-word. Instead, GailBot marks laughter with “(&=laughs)”. The machine learning model in our laughter plugin has an 88% per-frame accuracy (Ryokai et al., 2018; Abadi et al., 2016).

3.1.8 SPEAKER DIARIZATION PERFORMANCE

Speaker diarization, or the identification of speakers, is of central importance for any transcripts of dialogue. However, it is notoriously difficult to achieve reliably in mixed speaker, mono recordings, although advances in deep learning are improving the performance of STT systems (Park et al., 2021). GailBot relies on the diarization capability of the STT system being used. When speakers are recorded separately, GailBot labels identity based on the audio source, and therefore has an accuracy rate of 100%. When speakers are recorded on the same channel, STT systems, and by extension, GailBot struggles to identify speakers. We therefore recommend that researchers using GailBot record multiple speakers on separate audio channels. We do not evaluate speaker diarization in GailBot. We expect improvements in speaker diarization on single-channel audio once diarization algorithms in current STT systems have improved.

3.2 Evaluation

In this subsection, we compare GailBot transcripts to transcripts produced by other automated software (cf. Moore, 2015), corpora available online, and transcripts of the same data presented in a CA publication (Walker, 2017).

3.2.1 NEWPORT BEACH CORPUS

The Newport beach corpus (Jefferson, 2007) contains mono audio files with 44.1KHz sampling rates and high background noise. Transcribing this corpus is difficult for automated systems because it requires speaker diarization and background noise suppression. Excerpt 9 is a GailBot transcript of part of a phone call discussing the assassination of Robert F. Kennedy.⁵ GailBot achieved a SWER/RWER of 44.33%. For the same audio, (Moore, 2015) reported a 36% WER. Both systems produced errors based on a single phoneme (e.g., “makes” → “make”, “God” → “got”). They also mis-transcribed words with ‘nonstandard’ pronunciation (e.g., “didju” as “that you” and “of’m” as “up”). Finally, the systems were largely unable to transcribe quiet speech (“oh no. They drag it out so” on line 83 of Appendix D, Transcript 1) and overlapping speech.

Excerpt 9: GailBot transcript of the assassination call.

```

1   *SP2:  World uhm long week (0.3) my god I'm
2   *SP1:  Glad it's over I wanted to have a TV are on the
3           there

```

⁵Comparison transcripts for Excerpt 9 are in Appendix D.

4 (0.7)
 5 *SP2: Like check it out
 6 *SP1: That's where they be took off on our charter
 7 flight that same spot that you see it (0.8) >when
 8 I took him in the our<
 9 (1.0)
 10 *SP2: Would be more did I think it's so ridiculous I
 11 mean it's
 12 (0.7)
 13 *SP1: It's a horrible thing that my god play up that
 14 *SP2: Thing is
 15 *SP1: Just horrible guy people not
 16 *SP2: Ride::
 17 *SP1: In a make American people think well they're no
 18 good (0.5) well they aren't very good some up

In most cases, both GailBot and the Atilla system evaluated by Moore (2015) annotated quiet or non-canonical speech as silences. GailBot transcribed extra silences on lines 1, 4, 9, and 12. We believe this is because GailBot transcribed overlapping speech as silences due to word recognition errors. In comparison, the manual transcript contained only two silences: a micro-pause on line 80 and a 700 ms gap on line 86. Neither system transcribed the micro-pause and both estimated the gap to be 800 ms. Moore (2015) additionally notes Atilla's misidentification of a 100 ms pause on line 74. Excluding silences that could be attributed to word recognition errors, both GailBot and Atilla had 100% silence error rate.

Overlaps depend on speaker identity because they occur between different speakers. Neither Moore (2015) nor GailBot perform accurate speaker diarization on mono audio. Therefore, both were unable to accurately transcribe overlaps. Moore (2015) only transcribed one speaker during overlapping talk and did not annotate an overlap. Comparatively, GailBot transcribed overlaps as silences. Neither system transcribed any of the overlap markers that are present in the manual transcript, resulting in an overlap error rate of 100%.

3.2.2 CALLHOME CORPUS

The CallHome Corpus contains stereo telephone conversations between family and friends with a low 8KHz sampling rate. For Excerpt 10, GailBot has a SWER of 18% and a RWER of 9%.⁶ Word errors included phonological errors (e.g., “theories” was transcribed as “series”), misspellings of phonologically identical words (e.g., “four” was transcribed as “for”), and difficulty transcribing word cutoffs (“bul-” was transcribed as “Bo”). The overlap was correctly placed. Additionally,

⁶Comparison transcripts for Excerpt 10 are in Appendix E.

GailBot transcribed each silence as approximately 100 ms shorter than in the same data as transcribed by Walker (2017). This timing discrepancy eliminates the micropause identified by Walker (2017) in line 5 and the subsequent 900 ms to be transcribed as 800 ms long. However, this still marks an improvement over the CallHome corpus transcript, which does not annotate silence at all.

Excerpt 10: GailBot transcript of audio from the CallHome corpus.

```

1      *SP2: [They] they go through the series of the
2              three bullets of the magic one bullet
3      *SP1: [Yeah]
4              (2.9)
5      *SP1: Yeah for Bo yeah (0.8) it was interesting

```

Excerpt 11 is a GailBot transcript of a single speaker. GailBot produced one word-segmentation error (“and to” was transcribed as “into” in line 4), resulting in a SWER/RWER of 2%. Again, GailBot decreased silence duration by 100 ms, considering the 200 ms pause in line 1 to be a micropause and removing the micropauses in lines 3 and 5 of the transcript published in Walker (2017)⁷. GailBot also fails to identify the inhalations that are clearly audible in the recording. Instead, it marks inhalations as silences. Depending on the research question and the activities underway for speakers, CA transcribers may or may not mark all inhalations as interactionally relevant silences (Trouvain et al., 2020). We suggest that researchers interested in breathing patterns manually correct inhalation related inaccuracies in GailBot’s transcripts. Researchers may also be interesting in designing a custom inhalation plugin based on state of the art deep learning techniques (Nallanthighal et al., 2021).

Excerpt 11: GailBot transcript of audio from the CallHome Corpus, single speaker.

```

1      *SP1: Clinton just came out and said that he (.)
2              doesn't believe (0.4) in quota systems (0.4) and
3              in reverse discrimination but that he does
4              believe that affirmative action is necessary
5              (0.3) to move uhm you know black Americans (0.3)
6              forward into give them the opportunities that
7              they've been denied

```

⁷Comparison transcripts for Excerpt 11 are in Appendix F.

3.2.3 IN CONVERSATION CORPUS

The In Conversation Corpus (ICC) is a high audio quality (48 KHz sampling rate) conversation corpus, collected in the Human Interaction Lab (HI-Lab) at Tufts University. Each conversation features two students in two sound-proofed rooms that are separated by a glass window. Students communicated using a microphone and headset, and each student was recorded on a separate channel. We use the ICC to evaluate GailBot in two ways. First, we compare ICC and GailBot transcripts presented earlier in this paper. Second, we compare transcripts of 15 short segments of talk collected as part of a miscommunication study.

Excerpts 3, 5, and 7 are all GailBot transcripts from the the ICC and have a RWER of 0%. Excerpt 3 has a SWER of 2.8% (“kinda” transcribed as “kind of” (line 5)). Excerpt 5 has a SWER of 3.8% (“um” transcribed as “uhm” (line 1)). Finally, Excerpt 7 has a SWER of 2.9% (“showcase” transcribed as “show case” (line 2)). Additionally, we selected 15 miscommunication sequences that we expected would be difficult to transcribe. Combined, these sequences lasted 7 minutes and 30 seconds. They had a RWER of 19.7%, a SWER of 23.2%, an overlap error rate of 53.6%, and a silence error rate of 63.3%.

Excerpt 12: GailBot transcript from the ICC demonstrating overlap error.

```

1      *SP1: Love it (1.0) [what's your] favorite show
2      *SP2:                [      But      ] (0.9) oh (0.6) my
3                favorite show on Netflix will be house of cards
4                and friends
    
```

Excerpt 13: Manual transcription of Excerpt 12.

```

1      *SP1: love it
2                (1.0)
3      *SP1: w[hat's yo]ur favorite show
4      *SP2: [      But      ]
5      *SP2: um:  (0.2) my favorite show on Netflix will be:
6                house of cards and friends
    
```

Figure 3 shows that most overlap and silence errors have a small magnitude. Most overlap errors are 0-5 characters, and most silence errors have magnitude less than 300 ms. However, there are systematic sources of errors in GailBot’s transcripts. Its limited ability to detect TCUs and other pragmatic cues causes larger overlap and silence errors. For example, Excerpt 12, line 2 should be transcribed on multiple lines (see Excerpt 13, lines 4-5). Since GailBot’s default setting is to separate turns that are greater than 1000 ms apart, it transcribes the time between “But” and “um” (when SP1 is talking) as a within-turn pause. A manual transcriber was able to mark the inhalation of a new action (the initial question at line 3 in Excerpt 13), with a new line.

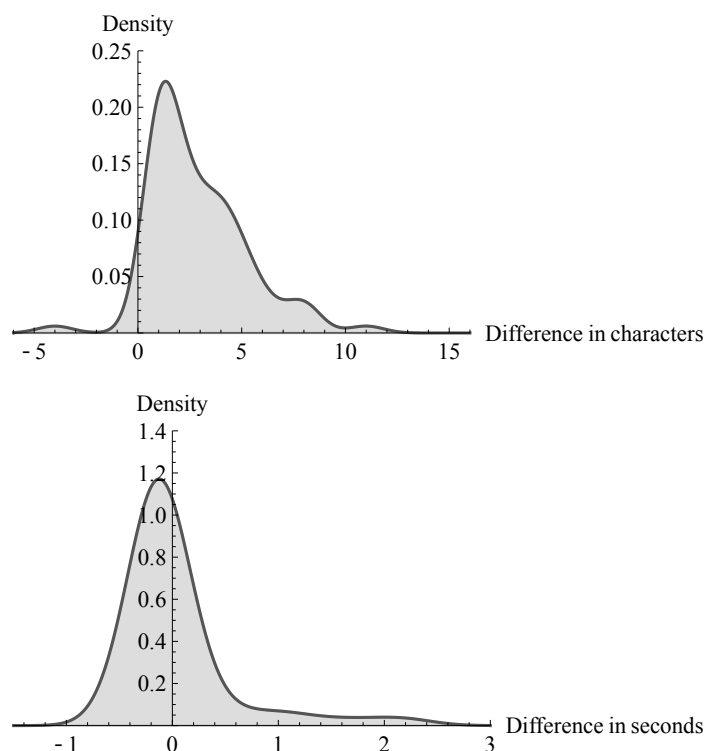


Figure 3: The magnitude of overlap (top) and silence (bottom) errors from the ICC. A positive value means that GailBot marked a wider overlap or longer silence than the manual transcriber.

Additionally, overlap errors can cause large silence errors. This occurs when one speaker speaks partway through their interlocutor's turn but finishes before their interlocutor. For example, in Excerpt 14, the 1.3 second gap in line 4 marks the time between the end of line 3 in Excerpt 15 ("Yeah") and the start of line 4 in Excerpt 15 ("Go..."). This includes the duration of Speaker 2's "other clubs I just like" (lines 2-3 in Excerpt 14, line 3 in Excerpt 15). This is a significant limitation of the overlap detection algorithm that should be manually corrected.

Excerpt 14: Overlap error: gap error.

```

1      *SP2: That's (0.8) yeah that's basically the (0.2) like
2          the club that I am really committed [to but other]
3          clubs I just like
3      *SP1:                                     [   Yeah   ]
4          (1.3)
5      *SP2: Go to the activities and (0.2) like I'm not
6          really P official (0.4) [member]
7      *SP1:                                     [Member] [yeah]
```

8	*SP2:	Yeah
---	-------	------

Excerpt 15: Overlap error: gap error, corrected.

1 *SP2: that's (0.5) yeah that's basically the (0.2) like
2 the club that I am really committed t[o but]
3 other clubs I just like

3 *SP1: [Yeah]

4 *SP2: Go to the activities and (0.3) like I'm not
5 really (0.5) the official mem[ber]

6 *SP1: [Memb]er y[eah]

7 *SP2: |yeah|

Similarly, overlapping talk can interfere with the turn construction algorithm. Human transcribers will split the first turn (the one that is overlapped) when there is a TRP or TCU. However, GailBot splits turns when there are pauses and not when the TCU ends. For example, in Excerpt 15, Speaker 1 interjects in Speaker 2's turn with "yeah." Instead of splitting Speaker 2's utterance when it approaches a TRP (after "committed to"), GailBot waits until Speaker 2 pauses slightly (after "I just like"). This creates a visual break that does not match the real structure of the conversation. Another limitation of GailBot is that it cannot produce turn-final Jeffersonian prosody markers until it has a more sophisticated TCU detection system. This is a clear next step for researchers.

3.3 Transcription Time Estimation

One of the co-authors with experience in CA transcription measured the time it took to transcribe a conversation from scratch compared to improving a ‘first draft’ transcript produced by GailBot. We selected a random conversation recording from the ICC. The co-author transcribed the first five minutes of the conversation manually and the second five minutes using GailBot output, using CLAN (MacWhinney, 2000) to transcribe the data. The transcriber focused on words, timing, pauses, overlap markers, and speech rate changes on the manual transcript to match the level of detail provided in GailBot’s output.⁸ It took 179 minutes to transcribe the first five minutes of the conversation from scratch without using GailBot. In contrast, it took 63 minutes to correct a GailBot transcription of the second five minutes of the conversation. This is a substantial reduction, with GailBot reducing transcription time by approximately two thirds. However, we note that the exact amount of time savings using GailBot will depend on the expertise of the transcriber (expert CA transcribers are faster than novices), the purpose of the transcript, the speaker accent or dialect (ASR accuracy decreases for speaker dialects not seen during training), as well as the frequency of overlaps. In short, using GailBot substantially reduces the time needed for CA transcription, but the exact amount of time reduction will depend on the particular conversation and transcriber.

⁸This level of detail is lower than for manually annotated CA transcripts, which also include annotations for a number of prosodic features that are not yet included in GailBot. This means that our manual transcription took less time than a full manual CA transcript, which influences the comparison.

3.4 Comparison with Data Annotation Systems

Desirable Features	Tools			
	GailBot	LabelStudio	Praat	ELAN
Speech To Text	Default-Auto	Enhanceable-Auto	Manual	Manual
Turn Construction	Default-Auto	Enhanceable-Auto	Manual	Manual
Silences	Default-Auto	Enhanceable-Auto	Default-Auto	Manual
Overlapping Speech	Default-Auto	Manual	Manual	Manual
Syllable Rate	Default-Auto	Manual	Manual	Manual
Laughter Detection	Default-Auto	Enhanceable-Auto	Enhanceable-Auto	Manual

Table 1: Comparison of automatic feature annotation capabilities between GailBot and existing data annotation systems.

GailBot aims to produce human-readable transcripts of dialogue in a format designed by conversation analysts to refine and interpret qualitative data (Ayaß, 2015; Ochs, 1979). This differs from data annotation systems that aim to produce categorical annotations, usually for computational linguistics or machine learning applications (e.g., Apostolova et al. 2010; Stenetorp et al. 2012; Yimam et al. 2013; Klie et al. 2018; Tkachenko et al. 2020-2021).

Data annotation systems exist in various domains including machine learning (Tkachenko et al., 2020-2021), phonetic analysis (Boersma and Weenink, 2009), and gesture analysis (Wittenburg et al., 2006). Table 1 compares GailBot capabilities with data annotation tools⁹ selected from various domains. The table includes a list of features desirable for automatically producing CA transcripts. Each tool may perform automatic annotation by default (Default-Auto), require enhancements for automation (Enhanceable-Auto), or only support manual annotations (Manual). Note that enhancements to a tool may range from additional package installations to implementing complex algorithms. Additionally, this comparison only identifies each tool’s ability to annotate a feature, not its ability to integrate these annotations into CA transcripts that use special symbols, as GailBot does¹⁰. The comparison highlights that, in principle, it might be possible to emulate CA transcripts by using generic data labelling systems to segment and annotate ASR transcripts, before exporting them using a template to generate CA-style symbols. However, enhancements to existing annotation systems may require fundamental software changes, which in turn can require significant amounts of additional resources. Instead, GailBot provides out of the box capabilities for generating first draft CA transcripts.

3.5 Summary

We evaluated GailBot across a range of mono and stereo audio sources and found that lower quality and/or noisy recordings resulted in higher error rates across the board. We know of no transcription system – including GailBot – that performs speaker diarization well enough to reliably identify overlapping speech on a single audio channel. GailBot was most accurate when transcribing high

⁹Our determination of each tool’s capability is based on publicly available documentation.

¹⁰A summary of Jeffersonian transcription symbols is provided in Appendix A.

quality stereo audio in low noise environments. Additionally, we find that word recognition and timing errors introduced by external STT services in GailBot decrease accuracy when identifying paralinguistic features. As ASR technology and speaker diarization in STT services develop further, GailBot’s ability to identify paralinguistic features in low quality data can be expected to improve accordingly. Finally, GailBot rarely identified the exact location to place overlap markers or the exact duration of silences. It often underestimated the duration of silences by approximately 100 ms to 200 ms and the location of overlap markers by 0-5 characters. Although these errors were relatively minor, some (e.g., overlap errors) had a significant effect on the accuracy of the transcripts and require manual correction for in-depth studies. Nonetheless, given GailBot’s framework for improvement, and the richness of its transcripts relative to existing automatically generated transcripts, it still produces valuable first draft CA transcripts.

4. Discussion

GailBot is an extensible, customizable framework for transcribing paralinguistic features of conversation. It produces first draft CA transcripts that are adequate for preliminary analyses and for identifying fragments of interest requiring more accurate manual transcription. As we learn more about how transcribers identify paralinguistic features, and as NLP technology improves, GailBot will become even more accurate and useful over time.

However, GailBot’s accuracy is currently limited by its plugins, each producing significant errors. One method of improving performance may be to use data-driven instead of heuristic-based approaches. For example, a recent Deep Neural Network (DNN) based end-of-turn detection model achieved an 82% accuracy (Masumura et al., 2018). However, training models is extremely time-consuming and often necessitates parallelization, which may not be accessible to some end users with limited resources (Narayanan et al., 2019). Additionally, there are a limited number of corpora of Jeffersonian transcripts available, especially given that training these models would require the time-synced audio along with the transcripts. This is a barrier to developing models that identify complex conversational features. With this in mind, our goal in developing GailBot was two-fold – to provide user-friendly software for generating first draft CA transcripts, and to provide an adaptable framework to integrate and improve the underlying models as they are developed by the research community.

More broadly, automated transcription software can only go so far (Ogden, 2015; Bolden, 2015). As Bolden (2015) argues, even if perfect automatically generated CA transcripts were available, researchers should still review and familiarize themselves with their data as a part of their analytic workflow. Another concern articulated by Bolden (2015) is that researchers may change their research agendas, including the type of data they collect, to use automatic transcription. This concern is complicated by the bias inherent in ASR (Ferrer et al., 2021). While Google’s STT claims to recognize over 70 languages and over 120 different local dialects and accents (Barnes, 2020), the accuracy of ASR when transcribing “nonstandard” dialects is poor (Varis et al., 2021). *Standard* dialects are those that have been institutionalized by the culture. For example, ASR systems make twice as many errors when recognizing speech produced by African Americans than by Caucasians (Koenecke et al., 2020). In addition, automatic sign language recognition lags behind spoken language recognition – the best systems have around a 30% WER (Adaloglou et al., 2020). If automated systems make it cheaper and easier to study certain kinds of data, some researchers may avoid studying informal, overlapping talk, sign language, and/or interaction in marginalized groups

and languages – issues that CA already struggles with due, in part, to its emergence in a particular socio-historical context (Hoey and Raymond, *forth.*). In addition, GailBot plugins likely also contain unrecognized biases about how interaction is ‘supposed’ to work. For example, Aboriginal Australians may tolerate longer silences than Anglo-Australians and Americans (Mushin and Gardner, 2009); the standard thresholds for pauses, micropauses and gaps may be inappropriate assumptions for some cultures. This is why it is important that researchers can change the default parameter settings in GailBot.

However, while we acknowledge that GailBot, like any other research tool, can potentially be misused, this should not prevent us from using and developing such tools. It is the shared responsibility of the research community to keep scientists accountable for their methodologies, tools and study designs. In addition, it is our shared responsibility to attempt to improve automated transcription for so-called “nonstandard” dialects and sign languages.

GailBot dramatically improves on previous automated transcription software for the purposes of CA research. Automated CA transcription and the availability of large-scale CA corpora will unlock many new research ideas and opportunities for interdisciplinary research, ranging from computational replications and extensions of conversation-analytic findings to understanding how participants in interaction employ paralinguistic features of talk. Future research may also improve services in existing domains such as service call quality assurance, clinician-patient interaction, and in designing and evaluating language interventions (Antaki, 2011). We invite researchers from across the social and computational sciences to use and contribute to the development of GailBot to help achieve this goal.

Acknowledgements

This work was partly funded by AFOSR grant FA9550-18-1-0465, as well as by the School of Arts & Sciences and the School of Engineering at Tufts University.

References

- Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Bijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, 2016. ISBN 9781931971331. doi: 10.1016/0076-6879(83)01039-3.
- Nikolas Adaloglou, Theocharis Chatzis, Ilias Papastratis, Andreas Stergioulas, Georgios Th Papadopoulos, Vassia Zacharopoulou, George J Xydopoulos, Klimnis Atzakas, Dimitris Papazachariou, and Petros Daras. A comprehensive study on sign language recognition methods. *arXiv preprint arXiv:2007.12530*, 2, 2020.
- Charles Antaki. *Applied conversation analysis: Intervention and change in institutional talk*. Palgrave Macmillan, London, 2011.
- Liana G Apostolova, Lisa Mosconi, Paul M Thompson, Amity E Green, Kristy S Hwang, Anthony Ramirez, Rachel Mistur, Wai H Tsui, and Mony J de Leon. Subregional hippocampal atrophy

- predicts alzheimer’s dementia in the cognitively normal. *Neurobiology of aging*, 31(7):1077–1088, 2010.
- Ruth Ayaß. Doing data: The status of transcripts in conversation analysis. *Discourse Studies*, 17(5):505–528, 2015.
- Calum Barnes. Announcing new features, models, and languages for Speech-to-Text, March 2020. URL <https://cloud.google.com/blog/products/ai-machine-learning/new-features-models-and-languages-for-speech-to-text/>.
- Stig Nikolaj Blomberg, Fredrik Folke, Annette Kjær Ersbøll, Helle Collatz Christensen, Christian Torp-Pedersen, Michael R. Sayre, Catherine R. Counts, and Freddy K. Lippert. Machine learning as a supportive tool to recognize cardiac arrest in emergency calls. *Resuscitation*, 138(October 2018):322–329, 2019. ISSN 18731570. doi: 10.1016/j.resuscitation.2019.01.015.
- Paul Boersma and David Weenink. Praat: doing phonetics by computer (version 5.1.13), 2009. URL <http://www.praat.org>.
- Christian Bokhove and Christopher Downey. Automated generation of ‘good enough’ transcripts as a first step to transcription of audio-recorded data. *Methodological Innovations*, 11(2): 2059799118790743, May 2018. ISSN 2059-7991. doi: 10.1177/2059799118790743. URL <https://doi.org/10.1177/2059799118790743>. Publisher: SAGE Publications Ltd.
- Galina B. Bolden. Transcribing as research: “Manual” transcription and Conversation Analysis. *Research on Language and Social Interaction*, 48(3):276–280, 2015. ISSN 0835-1813. doi: 10.1080/08351813.2015.1058603.
- Carolyn Boyce and P Neale. Conducting in-depth interviews: A Guide for designing and conducting in-depth interviews. *Evaluation*, 2(May):1–16, 2006. ISSN 1461-6734.
- Steven E. Clayman. Turn Constructional Units and the Transition Relevance Place. In *The Handbook of Conversation Analysis*, pages 150–166. John Wiley & Sons, 2012.
- Jan P. De Ruiter, Holger Mitterer, and N. J. Enfield. Projecting the end of a speaker’s turn: A cognitive cornerstone of conversation. *Language*, 82(3):515–535, 2006. ISSN 00978507. doi: 10.1353/lan.2006.0130.
- Paul Drew. “Quit talking while I’m interrupting”: A comparison between positions of overlap onset in conversation. In M. Haakana, M. Laakso, and J. Lindstrom, editors, *Talk in Interaction: Comparative Dimensions*, pages 70–93. Finnish Literature Society, 2009.
- Ryan Epp. Big phoney. 2018.
- Xavier Ferrer, Tom van Nuenen, Jose M. Such, Mark Coté, and Natalia Criado. Bias and Discrimination in AI: A Cross-Disciplinary Perspective. *IEEE Technology and Society Magazine*, 40(2):72–80, June 2021. ISSN 1937-416X. doi: 10.1109/MTS.2021.3056293. Conference Name: IEEE Technology and Society Magazine.
- Peter French and John Local. Turn-competitive incomings. *Journal of Pragmatics*, 7(1):17–38, 1983. ISSN 03782166. doi: 10.1016/0378-2166(83)90147-9.

- John J Godfrey, Edward C Holliman, and Jane McDaniel. Switchboard: Telephone speech corpus for research and development. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on*, volume 1, pages 517–520. IEEE Computer Society, 1992.
- Christian Heath, Jon Hindmarsh, and Paul Luff. *Video in qualitative research*. Sage Publications, 2010.
- Alexa Hepburn and Galina B. Bolden. *Transcribing for social research*. Sage, 2017.
- Alexa Hepburn and Scott Varney. Beyond ((laughter)): Some notes on transcription. In *Studies of Laughter in Interaction*, pages 25–38. Bloomsbury Academic, 2013.
- Elliott M. Hoey and Chase Wesley Raymond. Managing Conversation Analysis Data. In Andrea Berez-Kroeker, Brad McDonnell, and Eve Koller, editors, *The Open Handbook of Linguistic Data Management*. MIT Press, frth.
- G Jefferson. Notes on a possible metric which provides for a standard maximum silence of approximately one second in conversation (tilburg papers in language and literature 42). *Tilburg, Netherlands: Tilburg University*, 1983.
- Gail Jefferson. Glossary of transcript symbols with an introduction. In *Conversation Analysis: Studies from the first generation*, pages 13–31. John Benjamins, 2004a. doi: 10.1075/pbns.125.02jef.
- Gail Jefferson. A sketch of some orderly aspects of overlap in natural conversation. In Gene H. Lerner, editor, 2004, pages 43–60. John Benjamins Publishing Company, 2004b.
- Gail Jefferson. CABank English Jefferson NB Corpus [Data set]., 2007.
- Biing-Hwang Juang and Lawrence R Rabiner. Automatic speech recognition—a brief history of the technology development. *Georgia Institute of Technology. Atlanta Rutgers University and the University of California. Santa Barbara*, 1:67, 2005.
- Leelo Keevallik and Richard Ogden. Sounds on the margins of language at the heart of interaction. *Research on Language and Social Interaction*, 53(1):1–18, 2020. ISSN 08351813. doi: 10.1080/08351813.2020.1712961.
- Jan-Christoph Klie, Michael Bugert, Beto Boullosa, Richard Eckart de Castilho, and Iryna Gurevych. The inception platform: Machine-assisted and knowledge-oriented interactive annotation. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 5–9, 2018.
- Allison Koenecke, Andrew Nam, Emily Lake, Joe Nudell, Minnie Quartey, Zion Mengesha, Connor Touns, John R Rickford, Dan Jurafsky, and Sharad Goel. Racial disparities in automated speech recognition. *Proceedings of the National Academy of Sciences*, 117(14):7684–7689, 2020.
- Judith C. Lapadat and Anne C. Lindsay. Transcription in research and practice: From standardization of technique to interpretive positionings. *Qualitative Inquiry*, 5(1):64–86, 1999. ISSN 10778004. doi: 10.1177/107780049900500104.

- Christophe Leys, Christophe Ley, Olivier Klein, Philippe Bernard, and Laurent Licata. Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. *Journal of Experimental Social Psychology*, 49(4):764–766, 2013. ISSN 00221031. doi: 10.1016/j.jesp.2013.03.013.
- Anthony J. Liddicoat. *An introduction to Conversation Analysis continuum*. The Tower Building, New York, 2007.
- Brian MacWhinney. *The CHILDES Project: Tools for analyzing talk. transcription format and programs*, volume 1. Psychology Press, 2000.
- Ryo Masumura, Tomohiro Tanaka, Atsushi Ando, Ryo Ishii, Ryuichiro Higashinaka, and Yushi Aono. Neural dialogue context online end-of-turn detection. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 224–228, 2018.
- Bahman Mirheidari, Daniel Blackburn, Kirsty Harkness, and Traci Walker. Towards the automation of diagnostic Conversation Analysis in patients with memory complaints. *Journal of Alzheimer’s Disease*, pages 1387–2877, 2017.
- Gilad Mishne, David Carmel, Ron Hoory, Alexey Roytman, and Aya Soffer. Automatic analysis of call-center conversations. In *International Conference on Information and Knowledge Management, Proceedings*, May 2014, pages 453–459, 2005. ISBN 1595931406. doi: 10.1145/1099554.1099684.
- Robert J Moore. Automated transcription and Conversation Analysis. *Research on Language and Social Interaction*, 48(3):253–270, 2015. doi: 10.1080/08351813.2015.1058600.
- Ilana Mushin and Rod Gardner. Silence is talk: Conversational silence in australian aboriginal talk-in-interaction. *Journal of pragmatics*, 41(10):2033–2052, 2009.
- Venkata Srikanth Nallanthighal, Zohreh Mostaani, Aki Härmä, Helmer Strik, and Mathew Magimai-Doss. Deep learning architectures for estimating breathing signal and respiratory parameters from speech recordings. *Neural Networks*, 141:211–224, September 2021. ISSN 0893-6080. doi: 10.1016/j.neunet.2021.03.029. URL <https://www.sciencedirect.com/science/article/pii/S0893608021001179>.
- Deepak Narayanan, Aaron Harlap, Amar Phanishayee, Vivek Seshadri, Nikhil R. Devanur, Gregory R. Ganger, Phillip B. Gibbons, and Matei Zaharia. Pipedream: Generalized pipeline parallelism for dnn training. In *Proceedings of the 27th ACM Symposium on Operating Systems Principles, SOSP ’19*, page 1–15, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450368735. doi: 10.1145/3341301.3359646. URL <https://doi.org/10.1145/3341301.3359646>.
- Scott Novotney and Chris Callison-Burch. Cheap, Fast and Good Enough: Automatic Speech Recognition with Non-Expert Transcription. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 207–215, Los Angeles, California, June 2010. Association for Computational Linguistics. URL <https://aclanthology.org/N10-1024>.

- Elinor Ochs. Planned and unplanned discourse. In *Discourse and syntax*, pages 51–80. Brill, 1979.
- Richard Ogden. Data always invite us to listen again: Arguments for mixing our methods. *Research on Language and Social Interaction*, 48(3):271–275, 2015. ISSN 08351813. doi: 10.1080/08351813.2015.1058601.
- Tae Jin Park, Naoyuki Kanda, Dimitrios Dimitriadis, Kyu J. Han, Shinji Watanabe, and Shrikanth Narayanan. A Review of Speaker Diarization: Recent Advances with Deep Learning. *arXiv:2101.09624 [cs, eess]*, June 2021. URL <http://arxiv.org/abs/2101.09624>. arXiv: 2101.09624.
- Danielle M Pillet-Shore. Preference organization. *The Oxford research encyclopedia of communication*, 2017.
- Anita Pomerantz. Agreeing and disagreeing with assessments: some features of preferred/dispreferred turn shapes. In *Structures of social action*, pages 57–101. Cambridge University Press, Cambridge, 1984. ISBN 0521318629. doi: 10.1017/CBO9780511665868.008.
- Derek Roger, Peter Bull, and Sally Smith. The development of a comprehensive system for classifying interruptions. *Journal of Language and Social Psychology*, 7(1):27–34, 1988. ISSN 15526526. doi: 10.1177/0261927X8800700102.
- Kimiko Ryokai, Elena Durán López, Noura Howell, Jon Gillick, and David Bamman. Capturing, representing, and interacting with laughter. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pages 1–12, 2018. ISBN 9781450356206. doi: 10.1145/3173574.3173932.
- Harvey Sacks, Emanuel A. Schegloff, and Gail Jefferson. A simplest systematics for the organization of turn-taking for conversation. *Language*, 50(4):696–735, 1974. doi: 10.2307/412243.
- George Saon, Gakuto Kurata, Tom Sercu, Kartik Audhkhasi, Samuel Thomas, Dimitrios Dimitriadis, Xiaodong Cui, Bhuvana Ramabhadran, Michael Picheny, Lynn-Li Lim, et al. English conversational telephone speech recognition by humans and machines. *arXiv preprint arXiv:1703.02136*, 2017.
- Emanuel A. Schegloff. The relevance of repair to syntax-for-conversation. *Syntax and Semantics, Volume 12: Discourse and Semantics*, pages 261–286, 1979.
- Emanuel A Schegloff. Discourse as an interactional achievement: Some uses of ‘uh huh’ and other things that come between sentences. *Analyzing discourse: Text and talk*, 71:93, 1982.
- Hagen Soltau, George Saon, and Brian Kingsbury. The IBM Attila speech recognition toolkit. *2010 IEEE Workshop on Spoken Language Technology, SLT 2010 - Proceedings*, pages 97–102, 2010. doi: 10.1109/SLT.2010.5700829.
- Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun’ichi Tsujii. Brat: a web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107, 2012.

- Tanya Stivers, N. J. Enfield, Penelope Brown, Christina Englert, Makoto Hayashi, Trine Heine-mann, Gertie Hoymann, Federico Rossano, Jan Peter de Ruiter, Kyung-Eun Yoon, and Stephen C. Levinson. Universals and cultural variation in turn-taking in conversation. *Proceedings of the National Academy of Sciences*, 106(26):10587–10592, 2009. ISSN 0027-8424. doi: 10.1073/pnas.0903616106. URL <http://www.pnas.org/cgi/doi/10.1073/pnas.0903616106>.
- Paul Stonehouse. The Unnecessary Prescription of Transcription: The Promise of Audio-coding in Interview Research. *Research in Outdoor Education*, 17:1–19, 2019. ISSN 2375-5830. URL <https://www.jstor.org/stable/10.1353/reseoutded.17.2019.0001>. Publisher: Cornell University Press.
- Susan A Tilley. “challenging” research practices: Turning a critical lens on the work of transcription. *Qualitative inquiry*, 9(5):750–773, 2003.
- Maxim Tkachenko, Mikhail Malyuk, Nikita Shevchenko, Andrey Holmanyuk, and Nikolai Liubimov. Label Studio: Data labeling software, 2020-2021. URL <https://github.com/heartexlabs/label-studio>. Open source software available from <https://github.com/heartexlabs/label-studio>.
- Juergen Trouvain, Raphael Werner, and Bernd Möbius. An Acoustic Analysis of Inbreath Noises in Read and Spontaneous Speech. In *10th International Conference on Speech Prosody 2020*, pages 789–793. ISCA, May 2020. doi: 10.21437/SpeechProsody.2020-161. URL http://www.isca-speech.org/archive/SpeechProsody_2020/abstracts/168.html.
- Erika Varis, Ryan Georgi, Alicia Tsai, Antonios Anastasopoulos, Kyathi Chandu, Xanda Schofield, Surangika Ranathunga, Haley Lepp, and Tirthankar Ghosal. Proceedings of the fifth workshop on widening natural language processing. In *Proceedings of the Fifth Workshop on Widening Natural Language Processing*, 2021.
- Johannes Wagner. Conversation Analysis: Transcriptions and Data. In *The Encyclopedia of Applied Linguistics*, pages 1–8. American Cancer Society, 2020. ISBN 978-1-4051-9843-1. doi: 10.1002/9781405198431.wbeal0215.pub2.
- Gareth Walker. Pitch and the projection of more talk. *Research on Language and Social Interaction*, 50(2):206–225, 2017. ISSN 08351813. doi: 10.1080/08351813.2017.1301310.
- Peter Wittenburg, Hennie Brugman, Albert Russel, Alex Klassmann, and Han Sloetjes. ELAN: a professional framework for multimodality research. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC’06)*, Genoa, Italy, May 2006. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2006/pdf/153_pdf.pdf.
- Li-chiung Yang. Duration and pauses as cues to discourse boundaries in speech. In *Speech Prosody 2004, International Conference*, 2004.
- Seid Muhie Yimam, Iryna Gurevych, Richard Eckart de Castilho, and Chris Biemann. Webanno: A flexible, web-based and visually supported system for distributed annotations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 1–6, 2013.

Appendix A: Selected Jeffersonian notation features

Feature	Symbol
Overlap	[, [= start of overlap],] = end of overlap
Pauses/Gaps	(.) = micropause (0.4) = pause or gap of approximately 400 ms
Changes in speech rate	>word<= start/end of fast speech <word>= start/end of slow speech
Syllable elongation	: = extended by one beat
Latching	≈ current turn is latched to the previous turn
Laughter	.h = inhalation h = exhalation
Volume	o = start or end of quiet speech
Changes in pitch	↑, ↓ = shift to high or low pitch ↗ = rising to high / mid pitch ↘ = falling to low or mid pitch . = falling turn-terminal pitch ? = rising turn-terminal

Appendix B: Heuristics and thresholds in GailBot

Heuristic	Default value
Micropause	$100 \leq \text{duration} \leq 200$ ms (after rounding)
Pause	$200 \text{ ms} \leq \text{duration} \leq 1000 \text{ ms}$ (after rounding)
Within-speaker gap	$1000 \text{ ms} \leq \text{duration}$ (after rounding)
Between-speaker gap	$300 \text{ ms} \leq \text{duration}$ (after rounding)
Syllable rate MADs from median	2

Appendix C: Overlap algorithm

For all turn pairs, do the following:

1. Calculate the time between the start times of two consecutive turns.
2. If the result from step 1 is zero:
 - (a) Place the overlap-start markers (\lceil , \lfloor) at the start of both turns.
3. Otherwise:
 - (a) calculate the duration of both turns.
 - (b) For each turn, calculate the proportion of the turn duration that occurs before the overlap begins. Specifically, divide the overlap time by the total time of each turn.
 - (c) For each turn, calculate the number of characters in the turn.
 - (d) Calculate the proportion of the turn characters that may occur before the overlap begins. Specifically, multiply the output from line 4 and the output from line 5.
 - (e) Place the start overlap markers after those characters.
4. Calculate the time between the end times of two consecutive turns.
5. If the result from step 4 is zero:
 - (a) Place the overlap-end markers (\rceil , \rfloor) at the end of both turns.
6. Otherwise:
 - (a) Use the calculations from 3a-c to calculate the proportion of turn characters that may occur after the overlap ends.
 - (b) Place the end overlap markers before those characters.

Appendix D: Transcripts to compare to Excerpt 9

Transcript 1: Jefferson (2007) transcript.

78 *Lot: Oh: ↓Go:d a lo:ng wee[k. Yeah.]
 79 *Emm: [Oh: my] ↓ God
 80 I'm (.) glad it's over I won't even turn the
 81 teevee o]:n.
 82 *Lot: [I won'eether.
 83 *Emm: °aOh no. They drag it out so° THAT'S WHERE THEY
 84 WE TOOK OFF on ar chartered flight that sa:me
 85 spot didju see it↗
 86 (0.7)
 87 *Emm: .hh when they took him in[the airpla:ne,]
 88 *Lot: [n:No:::.] Hell
 89 I wouldn' ev'n wa:tch it.
 90 *Lot: I think it's so ridiculous. I mean it's .hhh
 91 it's a horrible thing but my: Go:d. play up
 92 that thing it it's jst ↑ hōrri[ble.]
 93 *Emm: [It'll] drive
 94 people nu:ts.
 95 *Lot: Why id i-en makes Americ'n people think why ther
 96 no goo:d.
 97 *Emm: °°Mm:°° Well they aren't very good some of'm,

Transcript 2: Moore (2015) transcript.

69 oh god long week
 70 oh my god
 71 i've decided sober i want you to have a t.v.
 72
 73 i won't either
 73.5 (0.7)
 74 like uh you know (0.1) that's where they
 75 we took off on our charter flight that same spot
 76 did you see it

77 (0.8)
78 and they took him and here uh you
79 know i wouldn't
80 watch it
81 i think it's so ridiculous i mean it's (0.4) it's a
82 horrible thing but my god (0.1) play up that's thing
83 it's it's (.) horrible
84 die people that
84.5 (0.3)
85 why is it a native american people think well they're
86 no good
86.5 (0.5)
87 well they aren't very good some of

Appendix E: Transcripts to compare to Excerpt 10

Transcript 1: CallHome Corpus transcript 4686.

42 *B: Did they go through the theories of the three
43 bullets or the magic one bullet?
44 *A: yeah four bul- yeah.
44 *A: It was interesting.

Transcript 2: Walker (2017), Excerpt 2.

1 *A: [(but) yeh]
2 *B: [did (.)] they go through the theories of the
3 three bulLETS, (or/and) the magic ONE bullet,
4 *A: YEA:H. (.) FOUR <<creaky>bull>. YEAH₀ (0.9) it
5 was[?] INtresting.

Appendix F: Transcripts to compare to Excerpt 11

Transcript 1: CallHome Corpus transcript 4247.

37 *A: Clinton just came out and said that he doesn't
38 believe in quota systems .
39 *A: and in reverse discrimination but that he does
40 believe that affirmative action .
41 *A: is necessary &=inhales to move &uh you know black
42 Americans &=inhale .
43 *A: forward and to give them the opportunities that
44 they've been denied.

Transcript 2: Walker (2017), Excerpt 3.

1 *A: clinton just came out and said that he:: (0.2)
2 doesn't believe °h (0.2) in quota systems °h (0.2)
3 and (.) in reverse discriminAtion.=but that he
4 does believe that affirmative action is NECessaryo
5 °h to mo:ve uh:° (.) you know black americans °h
6 forward and to give them the opportunities that
7 they've been deNIED.