Research Methods in Applied Linguistics 5 (2026) 100302

Contents lists available at ScienceDirect

Research Methods in Applied Linguistics

journal homepage: www.elsevier.com/locate/rmal

Evaluating and enhancing the accuracy of automated fluency annotation tools in L2 research

Jueyu Lu, John Rogers *

Department of English and Communication, Faculty of Humanities, The Hong Kong Polytechnic University, Hung Hom, Hong Kong SAR, China

ARTICLE INFO

Keywords:
Second language speech
Temporal fluency features
Automatic fluency assessment
Tool comparison
Hybrid automated-manual pipeline

ABSTRACT

Fluency is a central dimension of L2 oral proficiency. Further, fluency assessment is important for many applied contexts, including pedagogical and assessment purposes. Yet, the measurement of fluency using manual annotation is labor-intensive, which limits its broad application and scalability. We evaluate two automated tools — an acoustic-based tool (de Jong et al., 2021) and a machine-learning tool (Matsuura et al., 2025) — using data from L1-Chinese learners of English. Accuracy was assessed for three metrics, articulation rate (AR), pause ratio (PR), and mean pause duration (MPD), via Pearson correlations with manual annotation. We compared two automated tools and tested whether targeted manual post-processing (TextGrid checks and transcript adjustments) improves metric extraction using Steiger's test. Results from our sample indicated that de Jong et al. (2021) yielded higher accuracy for silence-based metrics (PR, MPD). However, text-dependent metrics (syllable number after removing disfluency words in AR) benefited from corrected TextGrids (for the acoustic tool) or corrected transcripts (for the machine-learning tool). These findings suggest a scalable division of labor: use an acoustic-based tool for silence-driven metrics, and apply corrected transcripts with a machine-learning tool when extracting text-sensitive metrics.

1. Introduction

Spoken fluency is a key component of second language (L2) proficiency (Tavakoli et al., 2023). It influences communicative effectiveness and is widely used in language and treatment assessment (Y. Suzuki & Hanzawa, 2022; Tavakoli et al., 2023). As fluency is a temporal phenomenon, its analysis requires time-aligned data on speech rate, pausing patterns, and disfluencies; all of these elements are time-consuming to annotate manually. In response to these challenges, the field of second language acquisition (SLA) has seen a growing use of automatic tools for spoken fluency analysis to increase the efficiency, objectivity, and scalability of fluency-related research (de Jong et al., 2021; de Jong & Wempe, 2009; Matsuura et al., 2025). Recent advances in speech technology, including acoustic analysis (e.g., syllables and pause detection), automatic speech recognition (ASR), and machine learning (ML), have enabled more precise extraction of fluency features (de Jong et al., 2021; Matsuura et al., 2025). Consequently, automated annotation tools are becoming an integral part of large-scale SLA studies, supporting systematic and replicable analyses of learner speech.

Despite wider access to automatic fluency annotation tools, their performance remains inconsistent across different learner populations and research contexts. Existing tools vary in terms of their input requirements, detectable fluency features, and output

* Corresponding author at: the Department of English and Communication, Faculty of Humanities, The Hong Kong Polytechnic University, Hung Hom, Hong Kong SAR, China.
E-mail addresses: jueyu.lu@polyu.edu.hk (J. Lu), john.rogers@polyu.edu.hk (J. Rogers).

https://doi.org/10.1016/j.rmal.2026.100302
Received 12 October 2025; Received in revised form 26 January 2026; Accepted 26 January 2026
Available online 30 January 2026

precision (de Jong et al., 2021; de Jong & Wempe, 2009; Matsuura et al., 2025). In the current literature, two representative approaches have been documented: one relies solely on acoustic signals (de Jong et al., 2021), and the other on accurate transcripts (Matsuura et al., 2025). These methodological differences often result in variation in both scope and reliability of the metric produced. A particular challenge arises when these tools are applied to L2 speakers whose L1 is typologically distant from the language represented in the automatic tools' training data. Prior work shows that individual speaking style in L1 (e.g., pausing, rate, repairs) transfers to L2 and that different L1s exhibit distinct styles that shape L2 fluency. To better assess L2-specific fluency, key metrics (e.g., syllable duration) should be L1-normalized in research and in selected testing contexts (de Jong et al., 2015; S. Suzuki & Kormos, 2025). Therefore, the automatic extraction of fluency metrics may become less reliable for learners from other linguistic backgrounds, potentially undermining the validity of downstream analyses. To mitigate this, many researchers continue to rely on human refinement, such as manual correction on automatically produced TextGrids, to improve the quality of fluency data (Bui et al., 2019; Takizawa & S. Suzuki, 2025). While such practices can enhance accuracy, they come at a cost to scalability: the workflow cannot be fully automated and therefore requires additional manual effort, and it inevitably involves annotator judgement in acoustically ambiguous regions (e.g., deciding where speech and silence begin and end). Clear operational rules and consensus coding can curb this risk, but cannot fully remove the accuracy-scalability trade-off.

Motivated by these concerns, this study systematically compares two widely used automatic fluency annotation tools: an acoustic tool (de Jong et al., 2021) and an ML tool (Matsuura et al., 2025). We focus on three temporal fluency metrics: articulation rate, pause ratio, and mean pause duration (S. Suzuki et al., 2021), and test how manual correction can enhance the accuracy of tool-generated fluency metrics. By evaluating raw versus corrected outputs, the study seeks to clarify when and how human refinement is most beneficial, offering practical guidance on selecting appropriate tools and designing efficient and reliable L2 fluency workflows.

2. Background

2.1. Challenges in automated fluency analysis

Despite significant advances in automated fluency analysis, several challenges remain concerning both cross-linguistic applicability and tool design limitations. One potential issue is the generalizability across different L1 backgrounds. For instance, although the tool developed by de Jong et al. (2021) was trained on speech data from learners with a range of L1s, its validation was conducted solely on speakers with L1 Dutch. Similarly, the tool proposed by Matsuura et al. (2025) was both trained and tested exclusively on L1 Japanese learners. As such, validation evidence is lacking for learners from different L1s. Existing evidence suggests that cross-linguistic influence — including segment, prosodic, and syntactic transfer — can affect L2 fluency patterns in ways that current tools may not adequately capture (de Jong et al., 2015; S. Suzuki & Kormos, 2025).

A second challenge relates to the design of the annotation tools themselves. Acoustic-based tools are robust in detecting acoustic/temporal features (e.g., syllable nuclei, silent pauses), but cannot identify content- or discourse-level disfluencies that require lexical identity, such as repetitions (repeating a word or phrase, e.g., "I, I think"), repairs (correcting a wrong start, e.g., "he go, he goes to school"), or self-corrections (replacing or rephrasing mid-utterance, e.g., "the lion, sorry, the tiger") (de Jong et al., 2021; de Jong & Wempe, 2009). On the other hand, the deep neural network-based forced alignment used in machine learning-based tools may yield imprecise word-silence boundaries, which can introduce errors in pause detection (e.g., missed pauses or spurious pauses) (Zhang & Hira, 2025). Additionally, recording quality and background noise affect both approaches: acoustic detectors (e.g., de Jong et al., 2021; de Jong & Wempe, 2009) can suffer when envelope peaks are masked or spurious, and ASR-dependent pipelines (Matsuura et al., 2025) can degrade when errors produced by the ASR recognition stage (e.g., insertions, deletions, or substitutions) propagate to downstream modules that use the ASR transcript, thereby affecting transcript-derived counts (e.g., syllable/words/clauses) and related segmentation/detection outputs.

2.2. Operationalizing temporal fluency: metrics for speed and breakdown

In L2 fluency research, fluency is commonly differentiated into three complementary dimensions: cognitive, utterance, and perceived fluency (Segalowitz, 2010). The temporal fluency metrics are typically understood as components of utterance fluency, which encompasses three dimensions: speed fluency, breakdown fluency, and repair fluency (Tavakoli & Skehan, 2005). The present study focuses on speed fluency and breakdown fluency, which S. Suzuki et al. (2021) reported to contribute more strongly to perceived fluency. Specifically, they found that perceived fluency was strongly associated with speed and pause frequency ($r = |.59–0.62|$), moderately with pause duration ($r = |.46|$), and only weakly with repair fluency ($r = |.20|$). Accordingly, three core temporal metrics were selected for cross-tool comparison: articulation rate, pause ratio, and mean pause duration, which we consider jointly as a composite characterization of temporal fluency. Articulation rate (AR), representing speed fluency, refers to the number of syllables produced per second of speaking time, excluding all pauses (S. Suzuki & Kormos, 2023, 2025). Pause ratio (PR) and mean pause duration (MPD), representing breakdown fluency, respectively, quantify how frequently pauses occur and how long they last on average (Peltonen, 2017; S. Suzuki & Kormos, 2023).

Methodologically, a 250 millisecond (ms) threshold was adopted to define a silent pause, to exclude brief "micropauses" (Riggenbach, 1991), which are less relevant to L2 proficiency (de Jong, 2016; de Jong & Bosker, 2013). Without this threshold, pause frequency can become dominated by numerous very short silences that contribute little to overall pause duration. This 250 ms criterion is also widely adopted in several recent studies (e.g., Matsuura et al., 2025). Syllable counts used for AR and PR excluded disfluency words (e.g., repetitions, repairs, self-corrections). This exclusion is consistent with current methodological syntheses that recommend

pruning such items to avoid inflating syllable output (Hanzawa, 2024; S. Suzuki & Révész, 2023; Y. Suzuki, 2021).

Regarding breakdown fluency granularity, fluency research distinguishes pauses occurring within clauses from those occurring at clause boundaries, commonly termed mid-clause pauses (MCP) and end-clause pauses (ECP), because pause placement can affect how pausing behavior is interpreted. In the present comparison, however, we do not separate MCP from ECP. This choice is driven by comparability constraints: the de Jong et al. (2021) acoustic tool does not identify syntactic boundaries and therefore cannot localize pause positions relative to clause structure. To ensure a unified standard across tools, we analyze pause frequency and duration at the global level rather than at MCP/ECP levels.

2.3. *Overview of the de Jong et al. (2021) tool*

The de Jong et al. (2021) tool is an updated version of de Jong and Wempe's (2009) original script. Beyond detecting syllable nuclei and silent pauses directly from the speech signal, the 2021 version also includes the detection of filled pauses. Implemented through Praat scripts, it auto-generates TextGrid annotations without transcripts or manual segmentation.

A strength of this tool is low technological demand: it operates directly on the audio signal via a Praat script, and does not require transcripts or high-performance hardware typically needed for machine-learning pipelines.

However, a major limitation, shared with the 2009 script, is the inability to detect disfluency words (e.g., repetitions, repairs). Thus, the syllable count used in this tool includes disfluent elements, potentially biasing some metrics, such as the number of syllables (excluding disfluencies). Because of this, studies using the 2009 script commonly are often combined with manual refinement of disfluency features (e.g., repetitions, repairs, and corrections) to ensure the accuracy and completeness of the fluency analysis (Hanzawa, 2024; Y. Suzuki, 2021).

2.4. *Overview of the Matsuura et al. (2025) tool*

The Matsuura et al. (2025) tool is built on aligned audio-text input, relying on transcripts rather than acoustic signals to extract a wider range of fluency features. They recommend using ASR transcripts (e.g., Rev.ai) for practical scalability: to improve workflow efficiency by reducing the time and labor cost of manual transcription, and to provide transcript input for subsequent automated processing in their annotation pipeline.

A key strength of this tool is the rich output. With natural language processing (NLP) and ML, it can identify disfluency words, detect clausal boundaries, and classify pause types. Thus, the syllable count used in fluency metrics excluded disfluent elements (e.g., repetitions, repairs, self-corrections), aligning with current methodological syntheses recommending the pruning of disfluency items to avoid inflating syllable output (Hanzawa, 2024; S. Suzuki & Révész, 2023; Y. Suzuki, 2021), thereby reducing or eliminating manual coding and improving scalability.

However, the tool also has notable limitations. Given that its forced-alignment component is built on an ASR/CTC-style model,[1] the boundaries between speech and silence can be imprecise, which may lead to errors in pause detection (e.g., missed pauses, spurious pauses, or biased pause durations) (Zhang & Hira, 2025). Additionally, ML models entail greater computational demands (e.g., hardware requirements, see Section 3.1.2 for details), reducing accessibility in low-resource environments.

2.5. *The current study*

Although automated fluency annotation has advanced, key questions remain due to limited evidence for cross-linguistic validity and tool-specific constraints. Existing tools are often validated on limited learner populations, and both tools have limitations: de Jong et al. (2021) lacks access to lexical identity, whereas Matsuura et al.'s (2025) forced alignment may yield imprecise speech-silence boundaries, leading to pause detection errors. To address this, the present study systematically compares two widely used tools: the acoustic-based tool by de Jong et al. (2021) and the ML-based tool by Matsuura et al. (2025). Both are applied to data from L1 Chinese learners of English, using a picture-based narrative task to test their cross-linguistic validity and practical usability. This study addresses the following research questions (RQs):

RQ1: How accurately do the two tools measure temporal fluency in L1 Chinese learners?
RQ2: Does human correction enhance the reliability of tool-generated fluency metrics?

3. Methodology

3.1. *Data and computational environment*

3.1.1. *Corpus*

17 Chinese L1 undergraduate students (5 male, 12 female, $M = 19.76$ years, $SD = 1.64$) participated in the study. All reported Chinese (Mandarin or Cantonese) as their L1 and had studied English for an average of 15.35 years ($SD = 2.32$). We assessed

[1] CTC-based alignment can yield "peaky" frame-level posteriors, and the blank label may absorb acoustically ambiguous frames. As a result, the localization of speech-silence boundaries (and thus silence intervals) can be unstable (Zeyer et al., 2021).

participants' general English proficiency descriptively using Brown's (1980) cloze test, scored with the Acceptable-Answer method. The acceptable-answer list follows Brown's (1980) as provided by The TwiLex Group (2024), with minor spelling errors ignored. Scores ranged from 17 to 50 ($M = 37.88$, $SD = 9.85$).

Speech data were collected via the Audio Recording (Beta) tool on the Gorilla Experiment Builder platform (Anwyl-Irvine et al., 2020). Participants used their own devices to complete a picture-based narrative task with 10 six-frame cartoon stories adapted from Heaton (1966), presented across four sessions. For each story, participants had one minute to prepare and three minutes to record, with no minimum speaking time required.

Each participant completed 10 recordings (170 in total). Valid recordings were defined as those that could be successfully processed by two automated fluency annotation tools, specifically de Jong et al. (2021) and Matsuura et al. (2025), and were intelligible to human listeners. Based on these criteria, 42 recordings were excluded due to file corruption ($n = 29$, 69.0 %), tool processing issues ($n = 11$, 26.2 %), or unintelligibility ($n = 2$, 4.8 %), resulting in 128 valid recordings retained for analysis, averaging 84.69 s ($SD = 27.26$), totaling roughly 3 h, and 25,010 syllables. All recordings were collected in .webm format, and subsequently analyzed by both tools.

3.1.2. Computational environment, inputs/operation, and outputs

All processing ran on a MacBook Pro (Apple M1 Pro chip, macOS Sequoia 15.5, 32GB RAM). Gorilla recordings (.webm) were converted to .wav using FFmpeg for tool compatibility. We evaluated two automated fluency annotation tools that differ in required inputs and operation. Matsuura et al. (2025) requires both .wav files and .txt transcripts. Its documentation specifies a 3-minute file cap, 5GB of free disk space, macOS Ventura 13.4, and 16GB RAM, with no state batch-duration limit. On our machine, we observed practical instability (frequent crashes) above approximately 2.5 min per file or 38 min per batch. de Jong et al. (2021) accepted .wav input without transcripts and file length or hardware constraints. There were occasional warnings related to sdF0, likely due to poor audio quality, flat pitch, or F0-tracking issues in specific utterances. For outputs, both tools produce machine-readable files suitable for downstream analysis. Matsuura et al. (2025) returns per-utterance CSV files containing the fluency metrics reported in this study (AR, MPD, and related counts), along with annotated TextGrids. de Jong et al. (2021) returns a per-utterance CSV with rate and pause-based metric (e.g., AR) plus annotated TextGrids. Table 1 summarizes the required inputs, how it is operated, and the outputs returned.

3.2. de Jong et al. (2021) procedure

3.2.1. Automated processing

The de Jong et al. (2021) tool was implemented using their Praat script to auto-annotate syllables and pauses with a 250 ms pause threshold. The initial output, "de Jong-Raw" (DJ-Raw), computed AR from total syllables without excluding disfluencies. By contrast, the tool does not compute PR or MPD. Therefore, these two metrics required additional post-processing, using values extracted by the script:

- $\text{PR} = \frac{\text{number of silent pauses}}{\text{detected syllable number}}$
- $\text{MPD} = \frac{(\text{total duration} - \text{phonation time})}{\text{number of silent pauses}}$

Table 1
Inputs, operation, and outputs of the two automated tools.

	de Jong et al. (2021)	Matsuura et al. (2025)
Hardware requirements	Not documented.	Docs: Apple M-series; ≥16 GB RAM; ≥5 GB disk; macOS Ventura 13.4+.
Preprocessing required	Format conversion (.webm → .wav).	Format conversion (.webm → .wav); Transcription.
Inputs required	.wav audio only.	.wav + paired transcript.
Installation & invocation	Praat script; batch processing supported.	Docker image; batch processing supported.
Audio processing limits	Not documented; none observed on our setup.	Docs: 3-min per-file cap; Observed on our setup: Single file: ≈2.5 min; total batch: ≈38 min.
Metrics available (used in this study)	Articulation rate; number of silent pauses; detected syllable count; total duration; phonation time.	Articulation rate; mean pause length; mid-clause pause ratio; end-clause pause ratio.
Other outputs	Annotated TextGrid.	Annotated TextGrid.

3.2.2. Manual annotation

3.2.2.1. Automatic annotation (de Jong et al., 2021). Following prior work using automatic Praat-based initialization (Bui et al., 2019; Takizawa & S. Suzuki, 2025), each recording was automatically pre-annotated in Praat (Boersma & Weenink, 2022) using the script by de Jong et al. (2021). The script generated a Praat TextGrid with three tiers that served as an initialization for subsequent manual verification and correction. Specifically, Tier 1 contained automatically detected syllable nuclei (as point annotations), Tier 2 contained automatically detected pausal speech segmentation represented as alternating speech and silence intervals, and Tier 3 contained short interval annotations anchored to each Tier 1 nucleus; intervals corresponding to potential filled pauses were labeled as "fp". This automatic TextGrid initialization reduced the manual coding workload by providing the required annotation framework and candidate locations, without constraining the final human decisions.

3.2.2.2. Manual correction and annotation. The auto-generated TextGrids were then manually verified and corrected in Praat by trained coders. The auto-generated tiers remained visible during editing; however, coders were explicitly instructed to treat them as provisional hypotheses. All final boundary and labeling decisions were made based on auditory inspection in conjunction with acoustic displays (waveform and spectrogram), following the operational definitions described below. To minimize the risk of simply "confirming" the initial automatic output, coders reviewed the entire recording sequentially (i.e., all automatically segmented intervals) and performed split/merge operations and boundary re-placement whenever the speech evidence warranted it.

First, coders adjusted interval boundaries in Tier 2 to ensure that the "sound" portions captured actual speech production. In particular, boundaries were moved to exclude non-speech events (e.g., background noise or breathing) when these had been erroneously included as speech, and to include low-intensity or irregular phonation that was part of the intended utterance (e.g., creaky voice and word-final consonants) when these had been erroneously classified as silence. Coders also corrected segmentation by merging or splitting intervals after moving the boundaries: (a) silence intervals shorter than 250 ms were merged with adjacent speech intervals, (b) intervals initially labeled as silence were merged if auditory and spectral inspection indicated the presence of speech, and (c) speech intervals were split when a non-speech event longer than 250 ms occurred within an automatically annotated speech interval, yielding two speech intervals separated by a silent interval.

Next, coders listened to each boundary-adjusted speech interval in full and estimated the number of syllables produced. Then they added or removed nuclei marks in Tier 1 so that the number of nuclei matched the perceived syllable count. Because our analyses relied on syllable counts rather than fine-grained temporal alignment, this procedure prioritized the accuracy of the number of syllable nuclei (i.e., point annotations) over the precise placement of each point annotation.

Last, coders annotated disfluent material as "df" in Tier 3, including filled pauses, repetitions, repairs, and self-corrections. Automatically detected filled-pause candidates were not accepted as final labels. Instead, they were removed and replaced with manual "df" annotations based on auditory and acoustic inspection. For example, filled pauses (e.g., uh, um) were labeled as one "df" interval. In repetitions (e.g., "I, I think"), the first repeated item (the first "I") was labeled as one "df" interval. In repairs (e.g., "he go, he goes …"), the pre-repair material ("he go") was labeled as two "df" intervals; and in self-corrections (e.g., "the lion, sorry, the tiger"), the aborted phrase ("the lion") and editing term ("sorry") were labeled as five "df" intervals.

To enhance procedural consistency, a double-coding procedure was applied. Following a consensus-coding procedure (see also Bui et al., 2019), coding was conducted by two coders. The two coders first co-coded 20 % of the recordings (26 randomly selected samples) and achieved 100 % agreement on annotation conventions by discussion. The remaining files were annotated independently, with regular discussion to resolve any uncertainties. This consensus-coding procedure follows common practice in this area (Kakitani & Kormos, 2024; Y. Suzuki & Hanzawa, 2022).

3.2.2.3. Data extraction and computation of measures. After manual correction, the revised TextGrids were processed with our custom script to extract the raw counts and durations required for the fluency measures. Pause-related information was derived from Tier 2 (speech/silent intervals), from which we obtained the number and total duration of silent intervals. Syllable information was derived from Tier 1 (syllable nuclei), and disfluent material to be excluded from syllable counts was derived from Tier 3. Specifically, the syllable count used in subsequent calculation was operationalized as the number of nuclei marks on Tier 1 minus the number of "df" intervals on Tier 3.

We then applied the same computation procedures as in Section 3.2.1 (i.e., the AR, PR, MPD formulas) to the manually corrected dataset. This finalized set of metrics, the "Standard Reference" (SR), served to evaluate the automated tools, because the TextGrids were manually verified for syllable and pause boundaries, ensuring human-verified timing and segmentation accuracy suitable for evaluation.

3.3. Matsuura et al. (2025) ***procedure***

3.3.1. Automated processing

The Matsuura et al. (2025) tool requires paired audios and transcripts. We generated ASR transcripts by Rev.ai and processed them with the tool. It implemented AR and MPD as intended, removing disfluency words from the syllable count, and including only silent pauses in pause-related calculations, but reported PR as two components: Mid-clause Pause Ratio (MCPR) and End-clause Pause Ratio (ECPR). To match the unified PR in this study, the two ratios were summed post hoc. Since both ratios use the same denominator (the total syllable count), MCPR + ECPR equals the overall PR. For consistency with de Jong et al. (2021), we used identical definitions,

including a 250 ms pause threshold. This output is termed "Matsuura-ASR" (M-ASR).

3.3.2. Manual annotation

To assess performance with higher-quality input, we manually corrected the ASR transcripts to fix common errors (e.g., misrecognition, omissions). This process was conducted collaboratively by the first author and a trained researcher. All transcripts were reviewed together, with disagreements resolved through discussion to ensure consistency. The Matsuura et al. (2025) tool was subsequently re-run using the corrected transcripts to obtain updated metrics. This output, "Matsuura-Corrected" (M-Corrected), reflects tool performance with human-verified input.

3.4. Comparison overview

To evaluate the accuracy and reliability of automated fluency metric extraction, we computed Pearson correlations with the Standard reference (SR) for three metrics: articulation rate (AR), mean pause duration (MPD), and pause ratio (PR). We did this for three datasets: DJ raw outputs (DJ-Raw), Matsuura with ASR transcripts (M-ASR), and Matsuura with human-corrected transcripts (M-corrected). We then conducted pairwise Steiger's tests for dependent and overlapping correlations with Holm adjustment. These comparisons addressed three aims:

(1) r (SR, DJ-Raw) vs. r (SR, M-ASR): to determine under fully automated conditions, which tool aligns better with SR (RQ1);
(2) r (SR, M-ASR) vs. r (SR, M-Corrected): to test whether transcript correction significantly improves alignment with SR (RQ2);
(3) r (SR, DJ-Raw) vs. r (SR, M-Corrected): to assess which tool, given its best input, achieves higher alignment with SR.

4. Results

4.1. Pearson correlations with SR

Fig. 1 shows the Pearson correlations with the SR for DJ-Raw, M-ASR, and M-Corrected across AR/MPD/PR. For the DJ-Raw, significant correlations were observed across all three metrics. AR showed a moderate correlation ($r = 0.561$ [0.429, 0.669], $p < 0.001$), while MPD and PR achieved very strong correlations ($r = 0.958$ [0.941, 0.970], $p < 0.001$; $r = 0.880$ [0.834, 0.914], $p < 0.001$). These results indicate that DJ-Raw is particularly accurate for silent-pause-based metrics, though less precise for syllable count.

In contrast, M-ASR showed generally lower correlations with SR. AR was moderate ($r = 0.460$ [0.311, 0.586], $p < 0.001$), MPD was weak ($r = 0.324$ [0.159, 0.471], $p < 0.001$), and PR was relatively strong ($r = 0.620$ [0.501, 0.717], $p < 0.001$). These results indicate greater metric-wise variability for M-ASR, with notable limitations in estimating pause duration.

Correlations between M-Corrected and SR were strong for AR and PR ($r = 0.692$ [0.589, 0.773], $p < 0.001$; $r = 0.772$ [0.691, 0.834], $p < 0.001$), and moderate for MPD ($r = 0.572$ [0.442, 0.678], $p < 0.001$).

To assess whether the current sample size provides sufficient sensitivity for the correlation analyses in this section, we conducted a sensitivity power analysis for Pearson's correlation (two-sided $\alpha = 0.05$, $n = 128$). The minimum detectable correlation was $r = 0.281$ for 90 % power. All observed correlations reported above exceed this threshold, indicating adequate power to detect effects of the magnitude found here.

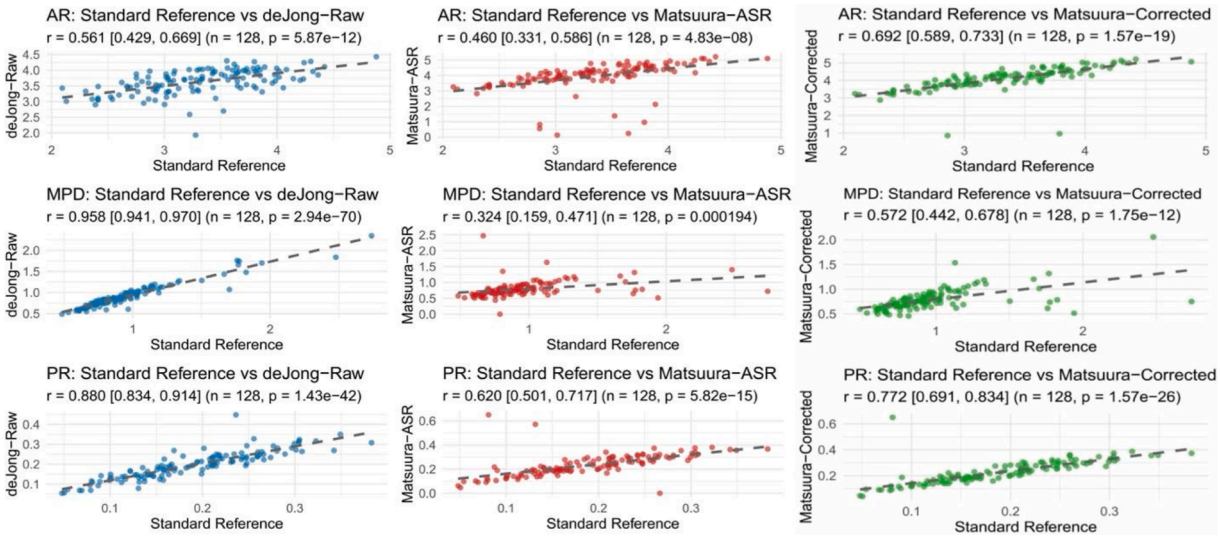

Fig. 1. Correlations with SR (Blue: DJ-Raw, Red: M-ASR, Green: M-Corrected).

4.2. Steiger's tests for pairwise dependent correlations

Beyond reporting correlations with SR, we evaluated dependent and overlapping correlations using two-sided Steiger's tests, with Holm adjustment across AR, MPD, and PR. We report three pairwise contrasts: DJ-Raw vs. M-ASR, M-ASR vs. M-Corrected, and DJ-Raw vs. M-Corrected (Table 2).

First, we compared DJ-Raw with M-ASR. For AR, the difference between r (SR, DJ-Raw) and r (SR, M-ASR) was not significant ($\Delta r = 0.101$ [−0.050, 0.252], $z = 1.140$, $p = 0.254$, Holm-adjusted $p = 0.254$). For MPD, DJ-Raw aligned more closely with SR than M-ASR ($\Delta r = 0.634$ [0.483, 0.786], $z = 13.249$, $p < 0.001$, Holm-adjusted $p < 0.001$). For PR, DJ-Raw again outperformed M-ASR ($\Delta r = 0.260$ [0.168, 0.351], $z = 6.144$, $p < 0.001$, Holm-adjusted $p < 0.001$). Overall, DJ-Raw significantly outperformed M-ASR on silent-pause metrics, whereas the evidence for AR was non-significant.

We then compared M-ASR with M-Corrected. For AR, r increased from 0.460 to 0.692 ($\Delta r = -0.232$ [−0.338, −0.127], $z = -3.649$, $p < 0.001$, Holm-adjusted $p < 0.001$). For MPD, r rose from 0.324 to 0.572 ($\Delta r = -0.248$ [−0.369, −0.128], $z = -3.241$, $p < 0.001$, Holm-adjusted $p < 0.001$). For PR, r increased from 0.620 to 0.772 ($\Delta r = -0.152$ [−0.212, −0.091], $z = -4.423$, $p < 0.001$, Holm-adjusted $p < 0.001$). These results confirm that transcript correction yields statistically significant correlation gains across metrics.

Lastly, we compared DJ-Raw with M-Corrected. For AR, r (SR, DJ-Raw) = 0.561 vs. r (SR, M-Corrected) = 0.692, $\Delta r = -0.131$ [−0.249, −0.013], $z = -1.874$, $p = 0.061$, Holm-$p = 0.061$, indicating a non-significant numerical advantage for M-Corrected. For MPD, DJ-Raw was markedly superior: 0.958 vs. 0.572 ($\Delta r = 0.386$ [0.276, 0.496], $z = 12.038$, $p < 0.001$, Holm-$p < 0.001$). For PR, DJ-Raw again outperformed: 0.880 vs. 0.772 ($\Delta r = 0.108$ [0.056, 0.161], $z = 3.782$, $p < 0.001$, Holm-$p < 0.001$). Overall, DJ-Raw significantly outperformed M-Corrected on MPD and PR. For AR, the advance of M-Corrected was not significant.

To contextualize non-significant Steiger contrasts, we conducted a simulation-based sensitivity analysis for differences between dependent correlations (two-sided, $n = 128$). Because Holm correction was applied across AR/MPD/PR within each contrast, sensitivity was evaluated conservatively at $\alpha = 0.017$ (0.05/3). Using the empirically observed inter-system correlations and SR-alignment levels, 90 % power required correlation differences of approximately $\Delta r_{min} = 0.075$–0.265 across metrics and contrasts. Consistent with this sensitivity, the two non-significant AR contrasts (DJ-Raw vs. M-ASR and DJ-Raw vs. M-Corrected) involved observed differences ($\Delta r = 0.101$ and 0.131) below Δr_{min}, whereas all significant MPD/PR contrasts exceeded the corresponding sensitivity thresholds.

4.3. Leave-one-subject-out robustness

To assess whether our findings are driven by any individual speaker and to account for within-speaker dependence, we conducted a leave-one-subject-out (LOSO) analysis. For each iteration, we excluded one speaker and recomputed the item-level Pearson correlations between the human reference and each tool using the same preprocessing and missing-data handling as in the main analysis. Correlations were highly stable across speakers: for every metric-by-tool combination, the LOSO median r closely matched the full-sample estimate, with median absolute deviations ≤ 0.021 and narrow interquartile ranges (typically 0.007–0.037). For example, PR (M-Corrected) yielded $r\_full = 0.772$ with a LOSO median of 0.768 (IQR = 0.007; range = 0.733–0.927), and AR (M-Corrected) yielded $r\_full = 0.692$ with a LOSO median of 0.687 (IQR = 0.016; range = 0.649–0.811). These results indicate that the observed effects are consistent across speakers and are not driven by any single participant (see Table 3).

Table 2
Pairwise Steiger tests vs. SR across metrics (with Pearson r).

Metrics	DJ-Raw: r [CI], p	M-ASR: r [CI], p	M-Corrected: r [CI], p	DJ-Raw vs. M-ASR	M-ASR vs. M-Corrected	DJ-Raw vs. M-Corrected
AR	0.561 [0.429, 0.669], $p < 0.001$	0.460 [0.331, 0.586], $p < 0.001$	0.692 [0.589, 0.733], $p < 0.001$	$\Delta r = 0.101$ [−0.050, 0.252], $z = 1.140$, $p = 0.254$, Holm $p = 0.254$	$\Delta r = -0.232$ [−0.338, −0.127], $z = -3.649$, $p < 0.001$, Holm $p < 0.001$	$\Delta r = -0.131$ [−0.249, −0.013], $z = -1.874$, $p = 0.061$, Holm $p = 0.061$
MPD	0.958 [0.941, 0.970], $p < 0.001$	0.324 [0.159, 0.471], $p < 0.001$	0.572 [0.442, 0.678], $p < 0.001$	$\Delta r = 0.634$ [0.483, 0.786], $z = 13.249$, $p < 0.001$, Holm $p < 0.001$	$\Delta r = -0.248$ [−0.369, −0.128], $z = -3.241$, $p < 0.001$, Holm $p < 0.001$	$\Delta r = 0.386$ [0.276, 0.496], $z = 12.038$, $p < 0.001$, Holm $p < 0.001$
PR	0.880 [0.834, 0.914], $p < 0.001$	0.620 [0.501, 0.717], $p < 0.001$	0.772 [0.691, 0.834], $p < 0.001$	$\Delta r = 0.260$ [0.168, 0.351], $z = 6.144$, $p < 0.001$, Holm $p < 0.001$	$\Delta r = -0.152$ [−0.212, −0.091], $z = -4.423$, $p < 0.001$, Holm $p < 0.001$	$\Delta r = 0.108$ [0.056, 0.161], $z = 3.782$, $p < 0.001$, Holm $p < 0.001$

Note. AR = articulation rate; MPD = mean pause duration; PR = pause ratio.

5. Discussion

5.1. Tool comparison: strengths and limitations

In our sample, de Jong-Raw (DJ-Raw) outperformed Matsuura-ASR (M-ASR) on silent pause metrics with large effects. For mean pause duration (MPD), DJ-Raw showed near-perfect alignment ($r = 0.958$) vs. M-ASR ($r = 0.324$), a very large difference ($\Delta r = 0.634$). For pause ratio (PR), DJ-Raw also had a substantial advantage ($r = 0.880$ vs. 0.620, $\Delta r = 0.260$). A likely reason lies in how pauses are identified from the signal. The two tools utilize different methods, which likely have influenced the results. DJ-Raw applies Praat's To TextGrid (silences) function, which yields boundaries directly from the waveform, without relying on lexical content. As a result, counts and durations of silences can be captured more consistently. By contrast, M-ASR infers pause timing from transcripts: a forced-alignment model maps each word in transcripts to when it was spoken in recordings. Since ASR typically performs worse and unevenly on L2 speech (Feng et al., 2024; Knill et al., 2018), following Matsuura et al.'s (2025) recommendation to use ASR for transcription can render the alignment unreliable and cause shifts in speech-gap boundaries. In addition, even with a correct transcript, the forced-alignment component in Matsuura et al. (2025), which is based on an ASR/CTC-style model, may yield imprecise localization of speech-silence boundaries, resulting in pause detection errors (e.g., missed pauses, spurious pauses, or biased pause durations) (Zhang & Hira, 2025). Therefore, pause estimation in Matsuura et al. (2025) can be less precise due to lower precision of boundary placement between speech and gaps.

For articulation rate (AR), DJ-Raw was slightly ahead of M-ASR; this comparison did not reach statistical significance, and both tools reached only moderate alignment with the Standard Reference (SR). DJ-Raw showed $r = 0.561$ vs. M-ASR $r = 0.460$, a small difference $\Delta r = 0.101$. One likely reason is how each tool handles the components of AR: phonation time and syllable counts. For phonation time, DJ-Raw benefits from relatively accurate silent-pause detection, yielding a more stable estimate of phonation time. By contrast, M-ASR may yield less reliable pause timing because its pause estimates are based on ASR-driven forced alignment, which can place speech boundaries imprecisely. This issue is particularly relevant when phonation time is computed by subtracting detected pauses from the total speaking time. For syllable counts, M-ASR leverages a RoBERTa-based disfluency filter to remove non-fluent tokens from the transcript, giving it an advantage over DJ-Raw, whose syllable-based counting cannot detect broader disfluency phenomena, which can inflate this metric. Consequently, both pipelines have critical gaps in the components of AR estimation, and taken together, these shortcomings result in suboptimal AR calculation for both tools.

5.2. Impact of human refinement: why manual correction helps

Manual correction (M-Corrected) was associated with higher agreement with the SR across metrics. Relative to M-ASR, correlations increased for AR (r: 0.460 → 0.692; $\Delta r = -0.232$; $z = -3.649$; Holm $p < 0.001$), MPD (r: 0.324 → 0.572; $\Delta r = -0.248$; $z = -3.241$; Holm $p = 0.001$), and PR (r: 0.620 → 0.772; $\Delta r = -0.152$; $z = -4.423$; Holm $p < 0.001$). Two aspects likely contributed. First, edits to the ASR output — such as correcting substitutions (e.g., cause → because), reinstating words missed under low volume or background noise, and tidying obvious tokenization issues — seem to reduce lexical noise that would otherwise affect syllable counts and the anchors used for alignment. Second, with fewer textual errors, the word-to-audio alignment appears more stable, which can sharpen pause boundaries and improve the estimation of phonation time, thereby benefiting the computation of MPD, PR, and AR. As an indication of editing scope, the manually corrected transcripts and the ASR output in our sample differed by 7.23 % WER (word error rate, word-level substitutions, insertions, and deletions).

5.3. Practical implications and recommendations

Overall, two complementary patterns emerge. For silent pauses, DJ-Raw appears more robust ($r = 0.958/0.880$ vs. $r = 0.572/0.772$, $\Delta r = 0.386/0.108$, both $p < 0.001$). For AR, M-Corrected yields a slightly higher correlation (0.692 vs. 0.561) but not significant ($\Delta r = -0.131$, Holm $p = 0.061$). Given the overlapping CIs and non-significant Δr, the apparent advantage for M-Corrected on AR should be regarded as tentative rather than definitive.

Therefore, the data from our sample suggest that for silent-pause metrics (MPD, PR), de Jong et al. (2021) is more reliable for stable, accurate results. For AR, if maximum accuracy is desired, manually correct the TextGrids generated by de Jong et al. (2021); however,

Table 3
LOSO robustness summary ($n = 17$ speakers).

Metric	Dataset	LOSO median r	IQR	Range [min, max]	$\lvert\Delta\rvert$ median	$\lvert\Delta\rvert$ max
AR	DJ-Raw	0.562	0.037	[0.514, 0.666]	0.015	0.105
	M-ASR	0.457	0.014	[0.413, 0.520]	0.011	0.060
	M-Corrected	0.687	0.016	[0.649, 0.811]	0.010	0.120
MPD	DJ-Raw	0.958	0.002	[0.948, 0.971]	0.001	0.013
	M-ASR	0.318	0.031	[0.255, 0.444]	0.021	0.120
	M-Corrected	0.569	0.016	[0.453, 0.668]	0.015	0.119
PR	DJ-Raw	0.880	0.005	[0.851, 0.923]	0.003	0.043
	M-ASR	0.615	0.015	[0.566, 0.705]	0.011	0.085
	M-Corrected	0.768	0.007	[0.733, 0.927]	0.005	0.156

this is time-consuming (typically 30–45 min for correcting a 3-minute monologic recording in our experience) and requires coding skills, so it may not be suitable for large samples or researchers without the requisite coding knowledge. For larger datasets, a manual correction of ASR transcripts + Matsuura et al.'s (2025) tool workflow is more efficient and easier to adopt, because transcript correction can be performed with solely text-editing interfaces, and entails simpler coder training. Note, though, that Matsuura et al.'s (2025) tool has higher hardware demands. Based on our experience, this tool imposes limits on individual audio length (≤2.5 min) and the total duration processed per batch (≤~38 min), necessitating batch scheduling.

6. Conclusion

In this study, we compared two automated tools for L2 fluency analysis and explored targeted adjustments to improve their performance. Within our sample, de Jong et al.'s (2021) acoustic pause-based tool produced relatively stable estimates for silence-oriented metrics (e.g., pause duration and ratio). Matsuura et al.'s (2025) ML-based tool supplied the textual alignment needed for syllable-related metrics, but appeared more sensitive to disfluency patterns. A hybrid arrangement that couples acoustic pause metrics with textual alignment for disfluency elements showed promise as a practical compromise, though it requires careful coordination and checking.

There are limitations to both tools. An acoustics-only tool cannot handle text-dependent metrics, such as excluding disfluency words when computing syllable counts. An ML-based tool can misplace boundaries and influence timing metrics — especially for L2 speakers. Practical constraints also matter: an acoustics-based tool scales more easily than an ML-based tool, and institutional or technical conditions may shape feasibility. These methodological and practical constraints are compounded by a modest recording sample size ($n = 128$) and the fluency metrics included in this study. In addition, based on the cloze scores (range = 17–50, Q1 = 29, *Mdn* = 38, Q3 = 48), the English proficiency of participants in our study spanned a moderate range with some concentration toward the higher end. Moreover, the Standard Reference (SR) was derived from manual correction of auto-initialized TextGrids; because the pre-existing tiers were visible during editing, some residual anchoring effects cannot be ruled out. These factors constrain the generalizability of the findings and underscore the need for additional research to validate these instruments across different learner populations and contexts. In light of these factors, manual coding remains important. Rather than replacing automation, focused manual correction — transcript correction, boundary verification, and checks on disfluency labels — can support more dependable annotation with manageable effort.

Funding sources

This project was supported by a General Research Fund Grant awarded by the University Grants Council, Hong Kong (Ref: 15601124)

Statements on open data

Data can be provided upon request.

CRediT authorship contribution statement

Jueyu Lu: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **John Rogers:** Writing – review & editing, Supervision, Resources, Project administration, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

We have nothing to declare.

References

Anwyl-Irvine, A. L., Massonnié, J., Flitton, A., Kirkham, N., & Evershed, J. K. (2020). Gorilla in our midst: An online behavioral experiment builder. *Behavior Research Methods, 52*(1), 388–407. https://doi.org/10.3758/s13428-019-01237-x

Boersma, P., & Weenink, D. (2022). *Praat: Doing Phonetics by Computer [Computer program]*. Version 6.2.23, retrieved 15 October 2023 from https://praat.org/.

Brown, J. D. (1980). Relative merits of four methods for scoring cloze tests. *The Modern Language Journal, 64*(3), 311–317. https://doi.org/10.2307/324497

Bui, G., Ahmadian, M. J., & Hunter, A.-M. (2019). Spacing effects on repeated L2 task performance. *System, 81*, 1–13. https://doi.org/10.1016/j.system.2018.12.006

de Jong, N. H. (2016). Predicting pauses in L1 and L2 speech: The effects of utterance boundaries and word frequency. *International Review of Applied Linguistics in Language Teaching, 54*(2), 113–132. https://doi.org/10.1515/iral-2016-9993

de Jong, N. H., & Bosker, H. R. (2013). Choosing a threshold for silent pauses to measure second language fluency. In *The 6th Workshop on Disfluency in Spontaneous Speech (Diss)* (pp. 17–20).

de Jong, N. H., Groenhout, R., Schoonen, R., & Hulstijn, J. H. (2015). Second language fluency: Speaking style or proficiency? Correcting measures of second language fluency for first language behavior. *Applied Psycholinguistics, 36*(2), 223–243. https://doi.org/10.1017/S0142716413000210

de Jong, N. H., Pacilly, J., & Heeren, W. (2021). PRAAT scripts to measure speed fluency and breakdown fluency in speech automatically. *Assessment in Education: Principles, Policy & Practice, 28*(4), 456–476. https://doi.org/10.1080/0969594X.2021.1951162

de Jong, N. H., & Wempe, T. (2009). Praat script to detect syllable nuclei and measure speech rate automatically. *Behavior Research Methods, 41*(2), 385–390. https://doi.org/10.3758/BRM.41.2.385

Feng, S., Halpern, B. M., Kudina, O., & Scharenborg, O. (2024). Towards inclusive automatic speech recognition. *Computer Speech & Language, 84*, Article 101567. https://doi.org/10.1016/j.csl.2023.101567

Hanzawa, K. (2024). Development of second language speech fluency in foreign language classrooms: A longitudinal study. *Language Teaching Research, 28*(3), 816–838. https://doi.org/10.1177/13621688211008693

Heaton, J. B. (1966). *Composition through pictures*. Longman.

Kakitani, J., & Kormos, J. (2024). The effects of distributed practice on second language fluency development. *Studies in Second Language Acquisition, 46*(3), 770–794. https://doi.org/10.1017/S0272263124000251

Knill, K., Gales, M., Kyriakopoulos, K., Malinin, A., Ragni, A., Wang, Y., & Caines, A. (2018). Impact of ASR performance on free speaking language assessment. *Proceedings of Interspeech 2018*, 1641–1645. https://eprints.whiterose.ac.uk/id/eprint/152761/.

Matsuura, R., Suzuki, S., Takizawa, K., Saeki, M., & Matsuyama, Y. (2025). Gauging the validity of machine learning-based temporal feature annotation to measure fluency in speech automatically. *Research Methods in Applied Linguistics, 4*(1), Article 100177. https://doi.org/10.1016/j.rmal.2024.100177

Peltonen, P. (2017). Temporal fluency and problem-solving in interaction: An exploratory study of fluency resources in L2 dialogue. *System, 70*, 1–13. https://doi.org/10.1016/j.system.2017.08.009

Riggenbach, H. (1991). Toward an understanding of fluency: A microanalysis of nonnative speaker conversations. *Discourse Processes, 14*(4), 423–441. https://doi.org/10.1080/01638539109544795

Segalowitz, N. (2010). *Cognitive bases of second language fluency*. Routledge. https://doi.org/10.4324/9780203851357

Suzuki, S., & Kormos, J. (2023). The multidimensionality of second language oral fluency: Interfacing cognitive fluency and utterance fluency. *Studies in Second Language Acquisition, 45*(1), 38–64. https://doi.org/10.1017/S0272263121000899

Suzuki, S., & Kormos, J. (2025). The moderating role of L2 proficiency in the predictive power of L1 fluency on L2 utterance fluency. *Language Testing, 42*(1), 73–99. https://doi.org/10.1177/02655322241241851

Suzuki, S., Kormos, J., & Uchihara, T. (2021). The relationship between utterance and perceived fluency: A meta-analysis of correlational studies. *The Modern Language Journal, 105*(2), 435–463. https://doi.org/10.1111/modl.12706

Suzuki, S., & Révész, A. (2023). Measuring speaking and writing fluency: A methodological synthesis focusing on automaticity. In Y. Suzuki (Ed.), *Practice and automatization in second language research* (pp. 247–266). Routledge.

Suzuki, Y. (2021). Optimizing fluency training for speaking skills transfer: Comparing the effects of blocked and interleaved task repetition. *Language Learning, 71*(2), 285–325. https://doi.org/10.1111/lang.12433

Suzuki, Y., & Hanzawa, K. (2022). Massed task repetition is a double-edged sword for fluency development: An EFL classroom study. *Studies in Second Language Acquisition, 44*(2), 536–561. https://doi.org/10.1017/S0272263121000358

Takizawa, K., & Suzuki, S. (2025). The role of multiword sequences in fluent speech: The case of listener-based judgment in L2 argumentative speech. *Studies in Second Language Acquisition*, 1–21. https://doi.org/10.1017/S0272263125000051

Tavakoli, P., Kendon, G., Mazhurnaya, S., & Ziomek, A. (2023). Assessment of fluency in the test of English for educational purposes. *Language Testing, 40*(3), 607–629. https://doi.org/10.1177/02655322231151384

Tavakoli, P., & Skehan, P. (2005). Strategic planning, task structure, and performance testing. In R. Ellis (Ed.), *Planning and task performance in a second language* (pp. 239–276). Amsterdam: John Benjamins.

The TwiLex Group. (2024). First language effects on incidental vocabulary learning through bimodal input: A multisite, preregistered, and close replication of Malone (2018). *Studies in Second Language Acquisition, 46*(5), 1413–1438. https://doi.org/10.1017/S0272263124000275

Zeyer, A., Schlüter, R., & Ney, H. (2021). *Why does CTC result in peaky behavior?* (No. arXiv:2105.14849). arXiv. https://doi.org/10.48550/arXiv.2105.14849.

Zhang, X., & Hira, M. (2025). *CTC forced alignment api tutorial—Torchaudio 2.9.0 documentation*. Torchaudio Documentation [Documentation] https://docs.pytorch.org/audio/stable/tutorials/ctc_forced_alignment_api_tutorial.html#inconsistent-treatment-of-blank-token.