Speak Up: Offline /r/ Screening for School Speech-Language Pathologists on a TI SK-AM62B

Godfrey Ponce, Rafael Amador, and Alexis Marsh
The University of Texas at Dallas · HackEdge 2026, Track 3: Edge AI Assistant

Code and documentation: **https://github.com/godfreyponce/speakup-hackedge-2026**

***Abstract*—A school speech-language pathologist (SLP) carries a caseload of about 50 students and still screens new children one at a time, by ear. Speech recognizers cannot help much, because they are built to find the word a person meant: a good one hears "wabbit" and writes "rabbit," which is exactly the information the SLP needs to keep. Speak Up is a screening aid that runs entirely on a Texas Instruments SK-AM62B with no network connection. A child says short words containing /r/, a small model we trained, Arti, rates each r against what trained listeners heard, and the SLP agrees with or overrides every call before it counts. We see Speak Up as the first, smallest piece of a larger system that understands atypical speech: what sounds were made, how they differed from the intended ones, and which errors repeat. We tested four stages of that system on /r/. Training on the share of listeners who heard an error, instead of a yes/no label, raised agreement with listeners on 55 children the model had never heard (Spearman 0.624, against 0.595 for the same network on yes/no labels), and Platt scaling cut calibration error from 0.107 to 0.026. Raising the third formant (F3) inside the r with Praat moved the error score by 0.338; the same change outside the r moved it by 0.005. A side task helped from 25% of the data up. A finer time grid and fine-tuning on a few clips from one child did not help. Arti takes 46.7 ms per clip on one Cortex-A53 core, and in practice mode a coach shows the child a tip for the error just made, worded by a small language model under code checks.**

***Index Terms*—speech sound disorders, /r/, school SLPs, edge AI, pronunciation assessment, calibration, formants.**

I. Introduction

Speak Up is the work of three people who came to the same problem from different sides. Alexis Marsh studies speech, language and hearing sciences and child learning and development, and she brought the clinical view: what a school SLP actually needs from a tool, and what a tool must never do. Rafael Amador built the hardware and the model experiments, and kept asking what a speech system could understand beyond words. Godfrey Ponce built the app, the language-model features and the practice coach, and wrote up the system and its results.

We kept all three voices in this paper instead of flattening them into one. Section II is the clinical problem, told from the SLP's side. Section III is the bigger idea: a speech system that understands atypical speech, and the pipeline it needs. Section IV places us among existing tools. Section V describes what we built, and Section VI is the part we care about most, the tests that check the claims of Section III one stage at a time, on /r/. Section VII covers our limits, our data and where the research goes next.

II. The Clinical Problem: Why School SLPs Need Help

This section is written from the clinical side of the team (A. Marsh).

A school speech therapist (SLP) in Texas carries about 50 students. Every one of them has goals, sessions, progress notes and paperwork, and on top of that the SLP is the person who screens new children who might need help. Screening every child takes hours, because it is done one child at a time, by ear. Those are hours the SLP is not spending in therapy with the children already on the caseload. When we talk about helping SLPs, this is the problem we mean: not that they lack skill, but that they lack time.

The /r/ sound is where a lot of that time goes. It is one of the most common speech sound errors in school-age children and one of the last sounds to develop. ASHA cites Crowe and McLeod [1], who place /r/ at about 5½ years, and ASHA notes that gliding ("wabbit" for "rabbit") is usually gone by 6 to 7 [2]. So the same error means different things at different ages: normal at five, worth watching at six, worth a closer look at seven. Speak Up follows those bands. An r error is normal for age under 60 months, worth keeping an eye on from 60 to 83 months, and flagged from 84 months.

Overload is not the only problem. Children who speak Spanish at home, or African American English, are too often flagged for errors that are really just normal dialect. An accent is not a disorder. A false referral worries a family and takes an evaluation slot that another child needed. That is why Speak Up scores the r at the start of the word, like the r in *red*, where dialects agree, and why fairness across accent groups is the first thing we will measure next (Section VII).

Early on, the team's idea was a coach that would run practice with the child on its own. I objected that a coach like that does the therapist's job. An overloaded SLP does not need a machine making clinical decisions; they need something that does the counting and the record-keeping, shows its evidence, and leaves every call to the person trained to make it. That became the rule the whole system is built around:

"Screening aid, not a diagnosis. The therapist always decides."

It is also why practice mode is "a tracker, not therapy." Levels from word to phrase to sentence are good for screening and practice, and every try lands in the SLP's log, but the SLP sets the goals. I chose the practice word list: rabbit, rain, ring, rocket and

truck for the r at the start; carrot and very for the middle; car, door and bear for the end. We used very instead of cherry and truck instead of giraffe, because “ch” and “j” are also late-developing sounds and would muddy what we are measuring. Any tip a child sees goes through me first.

For an SLP, the value of Speak Up is not that a model agrees with listeners 89% of the time. It is that a screening which took a session can take minutes, that the SLP hears every clip the model is unsure about, that each call comes with evidence they can check, and that the record of every try is already written when the child leaves the room.

III. From Speech Aid to Speech Understanding

This section is written from the research side of the team (R. Amador).

Our project was initially intended to be a speech aid for children, designed to detect speech abnormalities, identify patterns in pronunciation errors, and compare these patterns with those associated with known speech disorders and medical conditions. We also wanted to go beyond simply identifying errors by understanding the articulatory movements that may be producing these sounds and how they affect speech.

However, after speaking with other research students, we began to realize that this technology could serve a much broader and potentially more impactful purpose while still staying true to our original idea. Until now, we had focused nearly all our efforts on developing a tool for speech therapists, without fully considering the capabilities of the underlying speech recognition system. What we originally envisioned as a specialized speech aid could instead become the foundation for a more general-purpose system capable of understanding atypical speech, analyzing its acoustic characteristics, and identifying patterns that conventional speech recognition systems often overlook.

We realized that our original application was only scratching the surface of what this technology could potentially achieve. For this we need to investigate a system that asks:

> *“What acoustic events occurred, how were the sounds produced, how did they differ from the intended sounds, and what patterns emerge across repeated speech?”*

If we could develop a system like this, we would be able to move forward into arguably the most difficult phase of the project: collecting data, having an efficient system to do so in mass, and reaching out to all ethnicities, ages, and those who struggle with speech impediments. Here are the stages of the speech processing pipeline we need to develop in order to have a system that can do this.

1) Input processing. The data we collect from a voice sample, and what we turn it into for the model to process, is one of the most important steps in typical automatic speech recognition (ASR). However, for our application we need to think about it in a different way. Not only do we need the waveform data of the speech to recognize sounds, frequencies, and tones, but most importantly we need representations that preserve potentially important timing and spectral details.

2) Feature encoding. The next step in the system is finding patterns to encode specific information from the processed input. For example, typical ASR feature encoders are built for the primary purpose of transcription. This is great for having an efficient way to turn sound into a guess of what someone says, but it is not built for recognizing defective patterns and acting on them, to give a much more accurate transcription, or even a description of what is wrong with the sound, at a much higher level.

3) Adaptive resolution. Speech processing takes a finite number of samples for each snapshot of input data, and in the usual implementation this snapshot is constant, at around 20 ms per slice. However, to have a better view of the waveform for transition events that produce quick bursts of energy, you must be able to collect smaller samples of data with the same resolution to dissect the sound more accurately. To keep efficiency without making every slice smaller, you can implement adaptive resolution.

4) Prediction objective. The extent of what a model discerns from the data is called the prediction objective: the end goal of the system, what it wants to get out of the earlier process to create an output from. Instead of being restricted to finding the intended words or text, a system that would work for our architecture would need to produce the sounds made, acoustic features, intended meaning, and recurring deviations. When a system has all of those, it can act upon deviations and adjust itself to fit any user. This would be the first adaptive system of its kind, with a level of accuracy high enough to recreate the physical movements needed to create the sounds it is processing, and a system able to reach such conclusions would be a tool suitable for any need in speech systems.

5) Training. Lastly, an important part of processing is how we view training the model. In many systems, training is viewed as having one objective: minimize the transcription errors and produce the intended words. In our case we can shoot higher. Using multiple losses, we can measure phonetic accuracy, see exactly how sounds deviate from general phonetic sounds and by how much, and learn acoustic behavior, timing and meaningful patterns.

Speak Up is the first, smallest version of this system: one sound, on one small board, for one kind of user. It is small on purpose, because every stage above is a claim we can test on /r/ before we build on it. Section VI does that, stage by stage.

IV. Related Work and Where We Sit

PERCEPT-R [3]–[5] is a corpus of American English /r/ productions from children with and without speech sound disorders, each judged correct or incorrect by multiple listeners (Benway, McAllister, Preston and colleagues). We use it for all training and testing, and we keep its listener ratings instead of collapsing them to one label. Campbell et al. [6] compared acoustic measures against clinician ratings of children's /r/ and found that the F3–F2 distance explained the most variance. It is a recommended acoustic measure of /r/ accuracy, it is our baseline, and Speak Up shows it to the SLP next to every call. For /s/, Wong and Munson [7] guide which measures to use. Our fairness rules follow ASHA: dialect and second-language patterns are not disorders.

Competitors and the gap. Articulation Station helps the SLP, but the SLP scores everything by hand. Amira and Ambiki use AI in the cloud, so audio leaves the school. staRt (NYU) gives /r/ biofeedback, but the child reads a live acoustic display and it covers /r/ only. Speech Blubs and SayIt Labs are home practice games, not school screening tools. We have not tested every feature of every product, but the gap we aim to fill is an offline, clinician-in-the-loop screening aid for public schools, designed from the start to be fair across dialects.

Where we sit vs. ASR. Speech recognition is trained to guess the intended word, so it tends to correct a child's error away. Speak Up does not use a word recognizer at all. The SLP already knows the target word; our model scores how the sound was made against what trained listeners heard, and shows the acoustic evidence (the F3–F2 gap) to the SLP. Our model body is BC-ResNet [8], a small network designed for keyword spotting on low-power hardware.

V. Methodology

Sections V and VI keep the system write-up by G. Ponce, with the stage tests added.

A. System Architecture and Edge Implementation

A session uses two screens, where the child sees each target word on the board's monitor and says it into a USB microphone. The board scores the clip locally and sends the result over a direct cable to the SLP's browser review page. Arti scores the word against what trained listeners heard, and the page shows the clip's F3–F2 gap as evidence. There is no word recognizer, so the SLP confirms or rejects every call, and that answer is saved. The SLP enters the child's age, and the app only flags an error the child is old enough to be expected to have outgrown (Section II). A small on-board language model, Qwen2.5-0.5B-Instruct, writes session notes, and a checker rejects any note that contradicts the facts the code gives it. Since no audio leaves the room, we also stay clear of privacy rules for school records such as FERPA: there is no copy anywhere else to protect.

Because the board has four Cortex-A53 cores, 2 GB of memory and no accelerator, and the demo runs offline, we also report what Arti costs to run. Measured without the rest of the app, it takes a median of 46.7 ms on one thread for 1 s of audio, peaks at 72 MB and adds about 11 mJ per clip on the SoC rails. We use one thread on purpose: it is slightly faster than four on an idle board and twice as fast while Qwen and the browser are running (Fig. 1, Table I). Recordings stay on the board, but there is no delete button or login yet.

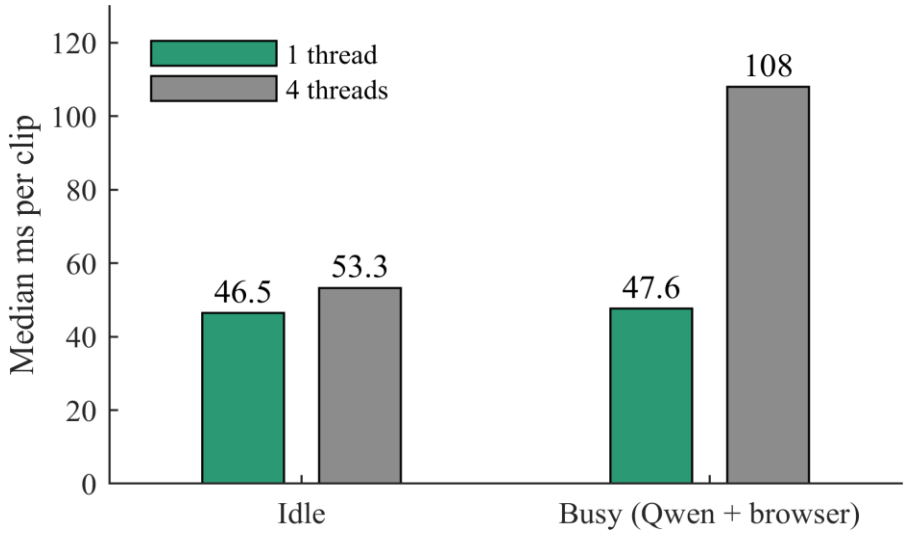


Fig. 1. Median time per clip for the v2 network on the SK-AM62B, 100 runs. On an idle board one thread is faster than four; with Qwen and the browser running, four threads more than double the time.

TABLE I. COST ON THE BOARD (IDLE, ONE THREAD)

Model	Test /300	Params	ONNX	ms/clip	RAM
Arti v0 (CNN)	258	60,706	0.95 MB	20.2	68 MB
v2, yes/no labels	267	~81k	1.27 MB	46.5	71 MB
v2, soft labels	268	~81k	1.27 MB	46.7	72 MB
v2, 5 ms hop	262	~81k	1.27 MB	95.5	–

B. Practice Mode and the Adaptive Coach

Practice mode uses the same loop, and it is "a tracker, not therapy." The child starts with a single word. A correct r moves them up to a short phrase, then a sentence, and an error moves them back down. Arti's call, as checked by the SLP, sets the level; the language model never does. At startup Qwen writes the phrase and sentence lines under a grammar that fixes their shape, and every line has to pass a check before a child can see it: real words only, the target word exactly once, no other r in the line, and the r in the right place in the word. If Qwen fails, the app uses a fixed bank of lines that our team wrote.

After a miss, the coach finds the r position (start, middle or end) the child misses most, once it has two tries there, and the next word comes from that position. The child's screen shows a coach panel beside the buddy (Fig. 2): the word, phrase and sentence steps with the current one lit, the word with its r in red, where the r sits, and a tip. After a miss the tip matches the error the SLP marked, gliding or distortion; otherwise it fits the r position. The panel never says right or wrong. On the SLP's page, a miss with low confidence (below 0.65) is marked "almost there." Qwen gets one more job: once at startup it rewords the approved tips for a six-year-old under a JSON grammar, and a rewording replaces a tip only if it has at most 16 words, uses only words from that tip plus a short allowed list, and ends a sentence. Code makes every choice; Qwen only changes wording.

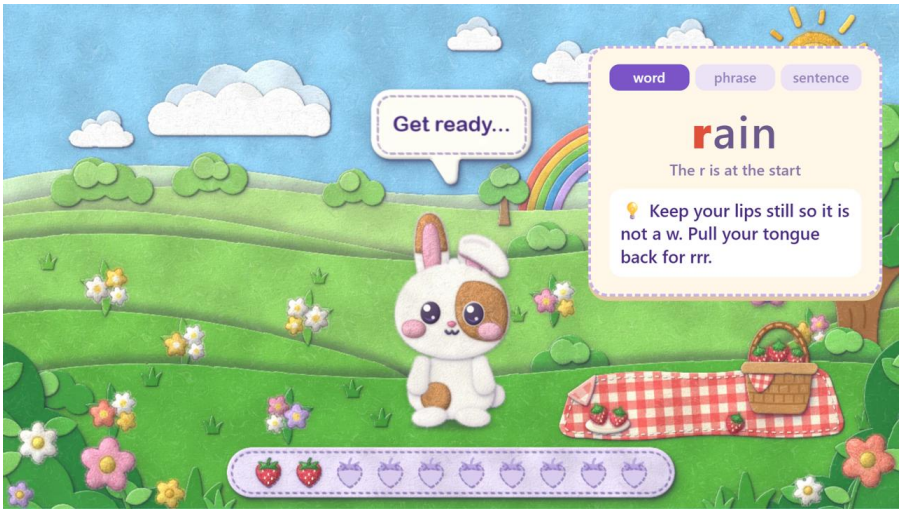


Fig. 2. The child's screen in practice mode, after a gliding miss on "rabbit." The coach panel shows the level steps, the next word with its r highlighted, and the gliding tip. No result is shown to the child. (Demo: recorded with a computer-made test clip, not a real child.)

C. The App, Screen by Screen

During a screening the child sees only the word, its picture and the buddy (Fig. 3). The buddy says "I'm listening!" while the board records, and a row of strawberries fills in as the child moves through the ten words. The child's screen never shows a result, a score or a flag; whatever the r sounded like, the child simply goes on to the next word.



Fig. 3. The child's screen during a screening, on the word "rabbit." The buddy shows it is listening; the strawberries count the words. No result is ever shown to the child. (Demo: recorded with a computer-made test clip, not a real child.)

The SLP works on a second screen (Fig. 4). On the left is the word list with each word's state. In the middle, Arti's suggestion is written in plain words ("the r sounded off"), with the error type it heard, the age check that decided whether to flag it, and the F3–F2 measurement as evidence. When the measurement and Arti disagree, the page says so and asks the SLP to listen first. Three large buttons, Disagree, Listen again and Agree (keys D, L and A), record the SLP's decision. On the right, the page keeps a running count of right and off by r position and repeats the rule: screening aid, not a diagnosis.

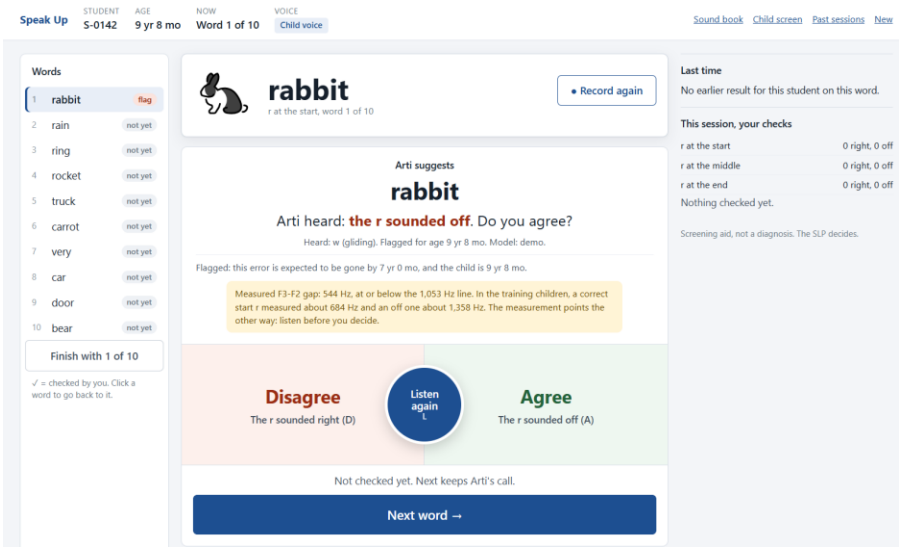


Fig. 4. The SLP's page for one clip. Arti's call, the error type, the age check and the F3–F2 evidence sit above the Disagree / Listen again / Agree buttons; nothing counts until the SLP answers. (Demo: recorded with a computer-made test clip, not a real child.)

Below each call, the SLP can open the evidence (Fig. 5). The recording panel shows the waveform with the part Arti scored shaded; the SLP can listen to that box, play the whole clip, or drag the box's edges, and Arti scores the new window as soon as it is let go. The sound picture is a spectrogram of the word with the first three formants drawn as colored dots and the F3–F2 gap marked where the r should be. It is labeled an estimate and the gap "experimental," because formant tracking on children's voices can fail.

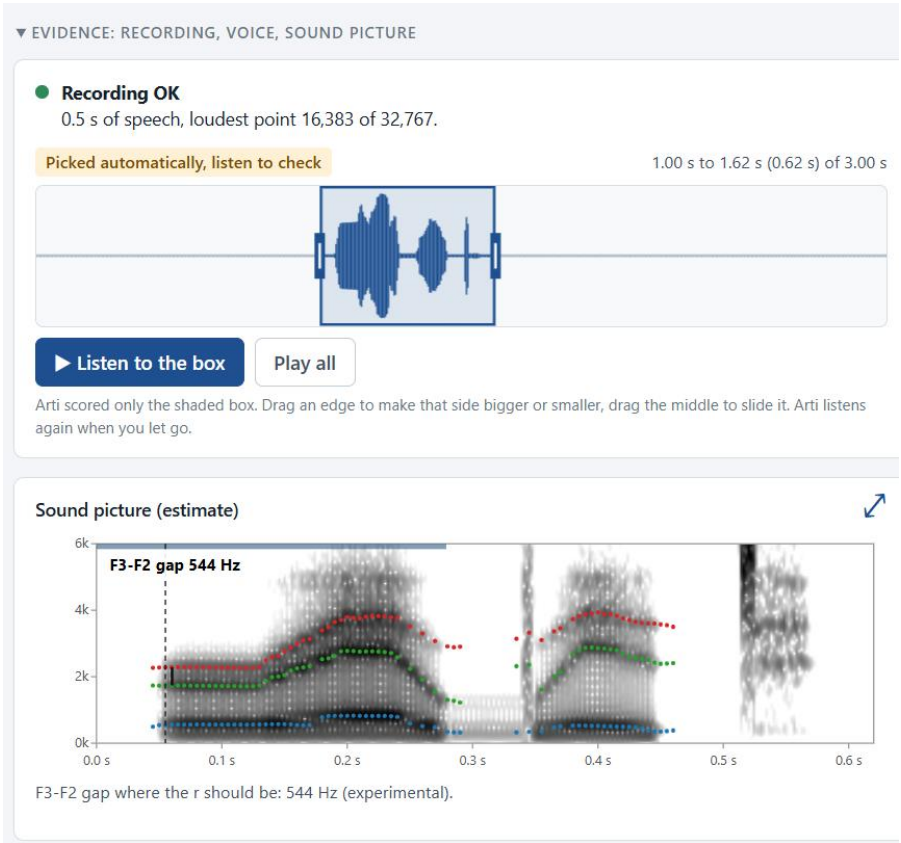


Fig. 5. The evidence panel: the scored window on the waveform (draggable, with Listen to the box), and the sound picture with F1, F2 and F3 tracks and the F3–F2 gap at the r. (Demo: recorded with a computer-made test clip, not a real child.)

When the SLP finishes, the session summary (Fig. 6) lists the words that were flagged after the SLP's checks, compares Arti's call with the SLP's for every word, and gives a short practice note for the family in English and Spanish. Past sessions and each child's history are kept on the board, so the SLP can see progress across visits.

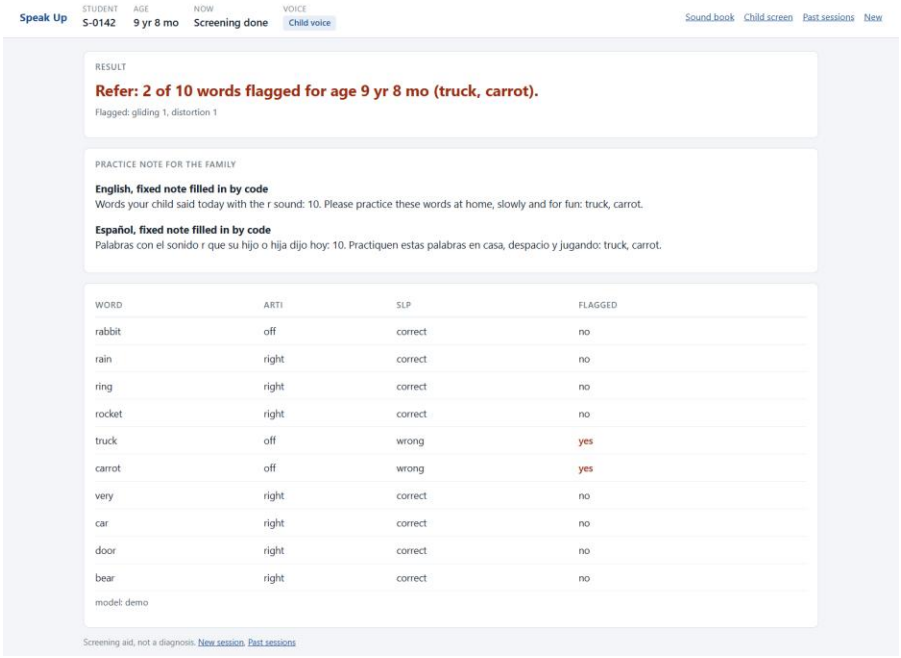


Fig. 6. The end-of-session summary: words flagged after the SLP's checks, Arti's call next to the SLP's for each word, and a family practice note in English and Spanish. (Demo: recorded with a computer-made test clip, not a real child.)

D. The Sound Book

A spectrogram is only useful to an SLP who can read one, and most school SLPs were not trained on acoustics. So Speak Up includes a Sound Book (Fig. 7), a reference written for SLPs that opens from the app at /book. It explains every part of the SLP screen (the waveform, the spectrogram, the formant dots, pitch, intensity and the F3–F2 gap) and then has an entry for each English sound. Every entry has the same parts: what the sound should look like on the spectrogram, why those frequencies (what the lips, tongue, jaw and soft palate are doing), typical values for children and adults with their ages and sources, and what common errors look like and why they happen. A search bar finds terms such as "gliding," "F3" or "lisp," and when the book is opened from a word page, the sounds in that word are listed first. There is also a printable version.

The book is a draft. Every number in it has a citation, and anything we could not confirm with the publisher, PubMed, Crossref or Semantic Scholar is marked UNVERIFIED in red. The example pictures are computer-made spectrograms built from published average formant values, not recordings of children, and each is tagged "From published averages" or "Estimated." Fig. 8 shows one: a correct r in "rabbit" next to the gliding "wabbit," where F2 drops and F3 stays high. Before an SLP relies on it, we want a licensed speech-language pathologist to review every entry, remove or correct each unverified number, and replace the computer-made pictures with real child recordings collected with consent. The finished version would be a printable booklet and an in-app guide that sits next to each call: an SLP who sees a high F3 in a child's "rabbit" could tap the r and read, on the same screen, what a typical r looks like, what gliding looks like, and which way to listen.

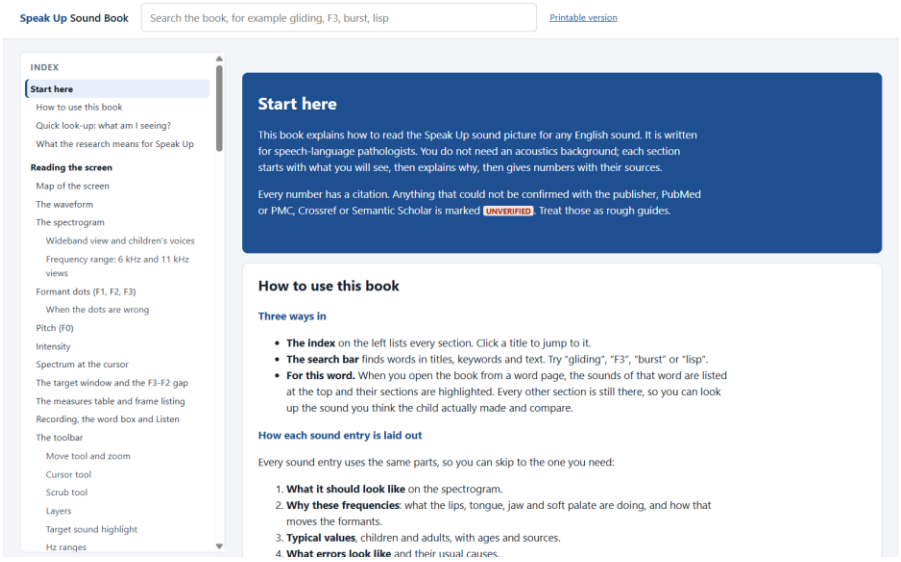


Fig. 7. The Sound Book in the app: an index of every section, search, and a printable version. Each sound entry shows what it should look like, why, typical values with sources, and what errors look like.

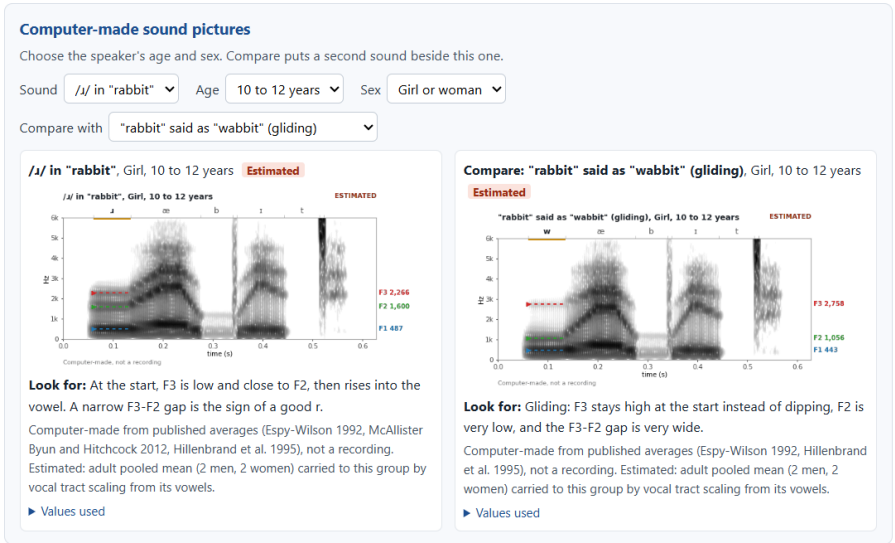


Fig. 8. A Sound Book comparison: a correct /r/ in "rabbit" (F3 about 2,266 Hz, F2 about 1,600 Hz) beside the gliding "wabbit" (F3 about 2,758 Hz, F2 about 1,056 Hz), each with what to look for and its sources. (Demo: computer-made pictures from published averages, not recordings.)

E. Acoustic Processing and Model Representation

Input processing is one of the most important steps, since a typical recognizer compresses the waveform to what it needs to predict words, which suits transcription. In our case, pronunciation evidence sits in short transitions, formant structure and timing, so the architecture we propose keeps those details, for example with adaptive time resolution, which uses small slices around quick changes and larger ones elsewhere. Today Arti uses a fixed input, 40 log-mel bands from 16 kHz audio with 25 ms windows every 10 ms, taken from the loudest 1.5 s of the recording (which holds the word in 95% of corpus clips), followed by a BC-ResNet with about 81,000 parameters. It is a 1.27 MB ONNX file. Arti was trained on children's voices, so for teenagers and adults the app first stretches the frequency axis so the voice looks more like a child's; that is how we can demo it on ourselves.

To test this, we retrained Arti with a 5 ms frame step instead of 10 ms. /r/ is a slow sound whose F3 drop spans many frames, so quick events such as stop bursts are where targeted finer resolution may still help.

F. Training Strategy and Prediction Objectives

We trained Arti on listener judgments from PERCEPT-R [3], split by speaker into 106 training, 12 validation and 55 test children. Our first model, v0, dropped clips where listeners disagreed, even though real children produce those in-between sounds. Instead of a correct or incorrect target, v2 trains each clip toward the share of listeners who heard the /r/ as wrong, which keeps the disagreement and adds about 8,700 clips. The raw score still needs calibration before it can be read as a probability. We used Adam (learning rate 0.001, batch 256, 20 epochs) and kept the best validation epoch.

In many systems training has one objective, but in one experiment our model also learned to predict whether the /r/ was at the start, middle or end of the word, with an auxiliary loss weight of 0.3. Four data fractions (10%, 25%, 50% and 100%), five seeds each, with and without the auxiliary task, gave 40 runs. Lastly, we fine-tuned the last block and output layer on 5, 10 or 20 clips from one child (20 test children, three random draws each) to see whether that child's speech helps.

G. Experimental Evaluation

Every result is scored on children the model never heard in training, and the main test set has 300 clips from 55 children, 150 heard as correct and 150 as wrong, comparing the shipped model with v0 and with the F3–F2 gap alone. Those clips are all clear cases, so we also scored all 21,591 labeled clips from the same children, using Spearman correlation for agreement with listeners and expected calibration error (ECE). We fitted temperature and Platt scaling on validation children and resampled children for intervals.

We used Praat [9] to raise F3 in steps up to 1,250 Hz in 150 correct clips from 35 test children, in the low-F3 /r/ frames and, as a control, in the high-F3 frames. Only the first change should move a score tracking the right cue, and since we edit audio, this cannot test tongue movement.

VI. Results and Discussion: Testing the Pipeline, Stage by Stage

Section III made claims about how a speech system for atypical speech should work. Before building any further, we tested four of those stages with our /r/ model, always scored on children it never saw in training. Table II is the short version; the subsections give the detail. We report the stages that did not pay off as plainly as the ones that did, because for /r/ a negative result tells us where not to spend effort, and it costs less than an unsupported claim in front of an SLP.

TABLE II. Each stage of the proposed pipeline, tested on /r/

Stage (Sec. III)	Test	Result	Verdict
Prediction objective	Listener share (soft) vs. yes/no labels	Spearman 0.624 vs. 0.595; ECE 0.026 after Platt	Supported; strongest evidence
Feature encoding	Raise F3 inside vs. outside the r	+0.338 vs. +0.005	Uses the right cue
Multiple losses	Position side task, 40 runs	Better from 25% of data up; not at 10%	Promising; needs CIs
Adaptive resolution	5 ms vs. 10 ms frame step	262 vs. 267 of 300; 2× time	Not needed for /r/
Per-user adaptation	Fine-tune on 5–20 clips	0.768 → 0.773	Null; needs varied clips

A. Prediction Objective: Listener Agreement and Calibration

On the 300 held-out clips, Arti v2 with soft labels agreed with listeners on 268, against 258 for v0 and 214 for the F3–F2 gap alone (Fig. 9). On all 21,591 clips from the same children, the soft-label model's Spearman correlation with the listener share was 0.624, against 0.593 for v0, and its mean absolute error fell from 0.239 to 0.208. This is the result we care about most, because the score rises with the share of listeners who heard an error. It is our strongest evidence for the richer prediction objective of Section III: a model trained on what listeners heard, not on a yes/no answer, follows listeners best on every test set (Table III).

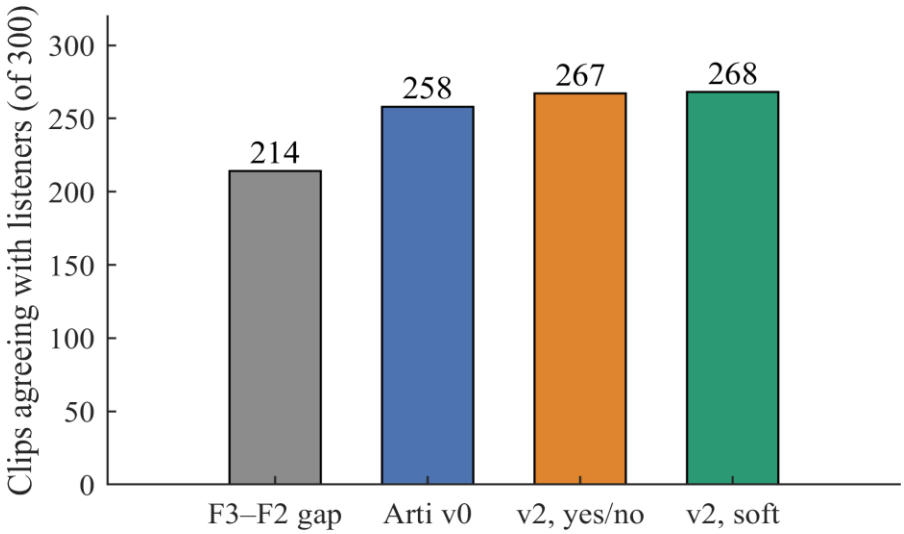


Fig. 9. Clips on which each method agrees with the listener majority, out of 300 clear test clips (150 correct, 150 incorrect).

TABLE III. AGREEMENT WITH THE LISTENER SHARE (SPEARMAN / PEARSON / MEAN ERROR)

Model	300 clear clips	55 test children	Val. children
Arti v0	0.636 / 0.748 / 0.189	0.593 / 0.634 / 0.239	0.591 / 0.626 / 0.221
v2, yes/no	0.734 / 0.805 / 0.159	0.595 / 0.630 / 0.237	0.610 / 0.645 / 0.217
v2, soft	0.747 / 0.827 / 0.156	0.624 / 0.676 / 0.208	0.684 / 0.698 / 0.187

The 95% confidence interval on that correlation is [0.531, 0.697], and the gain was +0.031 [0.001, 0.064] over v0 and +0.029 [0.001, 0.053] over the same network trained with hard labels. Both intervals just clear zero, so we read the gain as real but small. Platt scaling cut ECE from 0.107 to 0.026, and to about 0.02 for every model, from 0.10 to 0.21 raw. In plain terms: after Platt scaling, a score of 0.7 means about 70% of listeners would call the clip wrong, so the "Unsure, SLP please listen" band rests on real probabilities. On the validation children, a band of ±0.2 around the threshold holds about 19% of clips, and Arti is right on 92% of the rest. The clinical call stays with the SLP. Fig. 10 shows the raw scores before scaling: soft labels already sit closer to the listeners at low scores, where the older models call too many clips clean.

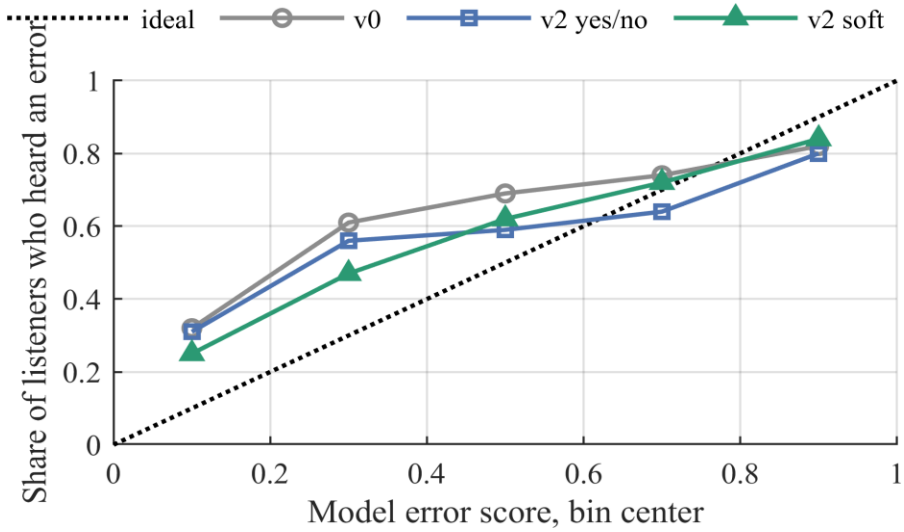


Fig. 10. Reliability on all 21,591 clips from the 55 test children, before calibration. Each point is the share of listeners who heard an error, for clips in one bin of the model's error score.

B. Feature Encoding: Does Arti Listen to the Right Cue?

A good /r/ pulls the third formant (F3) down; a derhotic /r/ leaves it high. Raising F3 in the low-F3 /r/ frames moved the soft-label model's probability of "wrong" from 0.312 to 0.650, a change of +0.338 [0.255, 0.420], while the control moved only +0.005 [−0.006, 0.015] (Fig. 11). The old model also reacted, +0.373 [0.311, 0.436], but its control moved too, +0.112 [0.089, 0.131]. Since only the new model's control stayed flat, it is more selective for the F3 cue: it responds to the acoustic cue of a derhotic /r/, not to any F3 change. That does not show it recovers tongue movement.

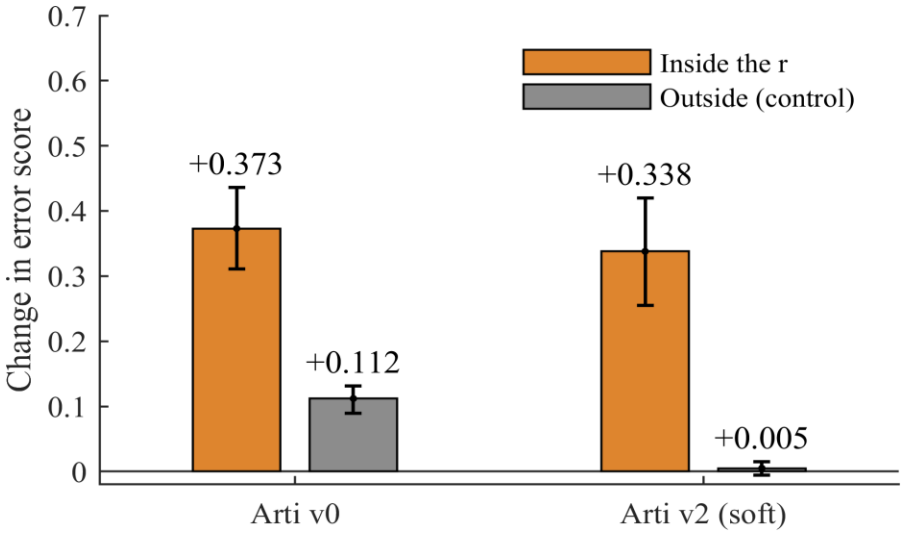


Fig. 11. Change in the error score when F3 is raised by Praat resynthesis, inside the r and in a control region outside it (150 correct clips, 35 test children; bars are 95% CIs over children).

C. Multiple Losses and Per-User Adaptation

Our first two-seed run suggested the side task helped when training data was scarce (+6 clips at 10% of the data). However, the five-seed rerun (Tables IV and V, Fig. 12) shows it did not help at 10% of the data; that earlier result came from too few seeds and was noise. Its means were higher at 25%, 50% and 100%, so the side loss helps once we have at least a quarter of the data. At full data, the mean absolute error also fell from 0.214 to 0.207. We have not yet run the uncertainty analysis these gains need, and the shipped model has no side task: its 268 comes from one run, and over five seeds the same setup averages 264.8 ± 4.1.

TABLE IV. CORRECT CLIPS OUT OF 300 (WITH VS. WITHOUT AUXILIARY TASK)

Training data	With vs. without
10%	243.4 ± 6.5 vs. 248.4 ± 7.0
25%	259.8 vs. 255.0
50%	266.6 vs. 259.8
100%	268.2 vs. 264.8

TABLE V. SPEARMAN CORRELATION (WITH VS. WITHOUT AUXILIARY TASK)

Training data	With vs. without
10%	0.521 vs. 0.503
25%	0.599 vs. 0.580
50%	0.624 vs. 0.602
100%	0.654 vs. 0.628

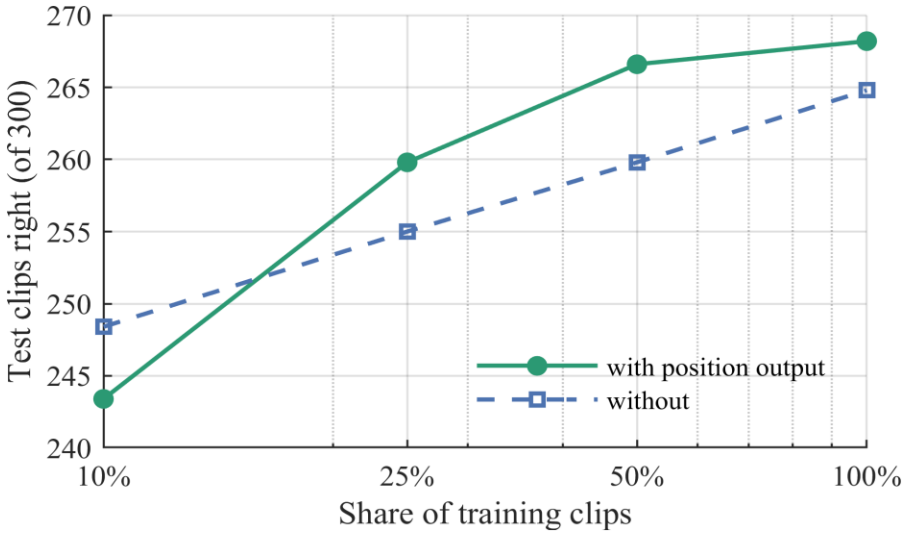


Fig. 12. Test clips right (of 300) with and without the position output, mean of five seeds per point.

Adapting to one child gave no meaningful average gain, since with 20 clips balanced accuracy only went from 0.768 to 0.773. It helped 11 of 20 children, the weakest most, one going from 0.52 to 0.64, while some strong children dropped. A child's few clips are mostly the same words, so the model adapts to the words more than to the voice, and testing this needs more varied clips per child.

D. Input Processing: Resolution and Cost

Board time doubled with the 5 ms frame step, from 46.5 to 95.5 ms per clip, and agreement went from 267 to 262 of 300, inside our run-to-run spread. This fits the theory: /r/ is a slow sound, so finer resolution should matter for quick bursts like stop consonants, not for /r/. Arti v2 takes a median of 46.7 ms on one thread for 1 s of audio and adds about 11 mJ per clip, far less than the second a child takes to say a word. With a model this cheap, an extra input must raise accuracy enough to pay for itself. For example, adding the F3–F2 gap as a second input gave 247 correct clips against 243 in one run on 25% of the data, inside run-to-run noise, while measuring the gap took about 0.2 s of CPU per clip on our laptop.

VII. Limits, Data and Where the Research Goes Next

A. Limitations

These results cover only /r/ and our held-out test sets, not other sounds, adult speakers, dialect groups or real schools. Dialect-fair screening is a design goal we have not measured, and we have not timed the full app with v2 on the board or measured its sensitivity and specificity against clinical decisions. The coach is a first version, untested with children, and its tips are drafts until A. Marsh approves them and, ideally, a licensed SLP reviews them. Qwen should never invent clinical advice.

B. Data and the Live Test

The next step is more sounds and more varied, annotated data, and free sets available now include JIBO Kids (with the DrivenData IPA transcripts) and Speechocean762, both CC BY, and PhonBank sets for non-commercial work. They label speech in different ways, so they cannot simply be merged. PERCEPT-R is distributed by TalkBank under its own rules, so we asked its authors for permission, and on Nov. 13, 2026, TalkBank told us that models trained on its data must not be shared publicly, so our public repository holds the training scripts but no weights. Long term we need our own IRB-approved recordings, and we plan to measure the false-alarm rate per accent group, aiming for a gap under five percentage points.

At the final, a judge says "red," "wed" and "thun" into a headset microphone with the internet unplugged, and the SLP's page shows each call and its evidence.

C. Scaling and Future Research

Beyond /r/, the architecture we propose would learn the intended content, the sounds actually produced, their timing and spectral features, and each speaker's recurring deviations. For this, we will next try adaptive resolution for quick sounds and more auxiliary tasks, then personalize once we have more varied clips per speaker. For the coach, the next steps are pictures of what to do with the mouth next to each tip, and a study of whether drilling the weak position helps more than the fixed order. So far our /r/ results do not show general atypical speech understanding or the recovery of articulation.

Two hardware additions should make screening more accurate. The first is a camera that watches the mouth. Some errors are hard to hear on a noisy classroom recording but easy to see: gliding ("wabbit" for "rabbit") usually brings stronger lip rounding and lip push than a good r, and a distorted r shows in the shape of the lips and jaw. A small camera, for example on a Raspberry Pi 5,

would track mouth landmarks (lip rounding, lip spread, jaw opening) during each word and keep only those numbers, never the video frames. Paired with Arti, this gives a second, independent signal. When Arti and the mouth movements agree, the SLP can trust the call more; when they disagree, or Arti is unsure, the clip is marked for the SLP to watch and listen to first. The same movements could tell gliding apart from other r errors and show the coach what the child's mouth actually did, so a tip can match it. The SLP still makes every decision.

The second is a board with an AI accelerator, such as TI's AM62A, which adds about 2 TOPS of dedicated AI compute next to the same kind of Arm cores as our SK-AM62B. Today Arti runs on the SK-AM62B's processor in 46.5 ms per clip, but every model shares those four cores: with Qwen and the browser running, time per clip more than doubles (Fig. 1). An accelerator would run the audio model and the camera model side by side, in real time, on one board and still fully offline, so no audio or video leaves the school. Neither the camera nor the AM62A is in this system yet; both are next steps we will test.

VIII. Conclusion

Speak Up started from three questions: what an overloaded school SLP needs, what a speech system could understand beyond words, and what a small board can actually run. Our answer so far is a /r/ screener that runs offline on a TI SK-AM62B and leaves every decision to the SLP. Soft labels improved agreement with listeners on children the model had never heard, calibration made its scores readable as probabilities, and the F3 test showed it responds to the cue it should. A side task helped once we had enough data. A finer time grid and few-shot fine-tuning did not help in our tests, and we report them so others can skip them or test them with better data. Each of these is one tested stage of the larger system in Section III, and each one keeps the therapist in charge.

References

[1] K. Crowe and S. McLeod, "Children's English consonant acquisition in the United States: A review," *American J. Speech-Language Pathology*, vol. 29, no. 4, pp. 2155–2169, 2020. doi:10.1044/2020_AJSLP-19-00168.

[2] American Speech-Language-Hearing Association, "Selected phonological patterns," Practice Portal: Articulation and Phonology. https://www.asha.org/practice-portal/clinical-topics/articulation-and-phonology/selected-phonological-patterns/

[3] N. R. Benway, J. L. Preston, E. R. Hitchcock, A. Salekin, H. Sharma, and T. McAllister, "PERCEPT-R: An open-access American English child/clinical speech corpus specialized for the audio classification of /ɹ/," in *Proc. Interspeech 2022*, pp. 3648–3652. doi:10.21437/Interspeech.2022-10785.

[4] N. R. Benway, J. L. Preston, E. Hitchcock, Y. Rose, A. Salekin, W. Liang, and T. McAllister, "Reproducible speech research with the artificial intelligence-ready PERCEPT corpora," *J. Speech, Language, and Hearing Research*, vol. 66, no. 6, pp. 1986–2009, 2023. doi:10.1044/2023_JSLHR-22-00343.

[5] N. R. Benway, *PhonBank Clinical PERCEPT-R Corpus*, TalkBank, 2022. doi:10.21415/0JPJ-X403.

[6] H. Campbell, D. Harel, E. Hitchcock, and T. McAllister Byun, "Selecting an acoustic correlate for automated measurement of American English rhotic production in children," *Int. J. Speech-Language Pathology*, vol. 20, no. 6, pp. 635–643, 2018. doi:10.1080/17549507.2017.1359334.

[7] E. Wong and B. Munson, "Beyond spectral moments: Validating alternative measures of sibilant fricatives using listener ratings of children's speech," *JASA Express Letters*, vol. 5, no. 10, 105202, 2025. doi:10.1121/10.0039497.

[8] B. Kim, S. Chang, J. Lee, and D. Sung, "Broadcasted residual learning for efficient keyword spotting," in *Proc. Interspeech 2021*, pp. 4538–4542. doi:10.21437/Interspeech.2021-383.

[9] P. Boersma, D. Weenink, and A. Shchupak, *Praat: doing phonetics by computer* [software]. https://www.praat.org