Spectral Gravity Formant Estimation for Phonetic Segmentation

Michael S. Yantosca, Albert M. K. Cheng
Department of Computer Science
University of Houston
Houston, TX, USA
msyantosca@uh.edu, amcheng@uh.edu

Abstract

Recent automated transcription systems have focused on end-to-end orthographic approaches driven by deep neural networks and sequence-to-sequence transformers. Growing public interest in transcription at the phonemic or phonetic level has led to re-purposing these systems to segment and identify phones, the basic sounds which comprise human speech. However, they miss the mark on a fundamental component of time-series analysis, namely time. For linguistic applications which require high fidelity in the temporal domain, the loss of timing information is untenable. Our work proposes a deadline-bounded expectation maximization (EM) algorithm with a novel initialization method to estimate formants, i.e., salient speech frequencies, for enhanced phonetic segmentation. Based on the concept of spectral gravity, i.e., treating spectral energy as mass attenuated by the square of frequency distance across the spectrum, our technique outperforms the recent state of the art on key clustering metrics, generating reasonable alignments across multiple languages with no *a priori* training.

1 Introduction

If the saying is true that excluding linguists increases the performance of speech recognizers, one must ask what constitutes performance. Simply targeting character error rate (CER) or word error rate (WER) does not confer understanding of articulatory processes which may be used to extrapolate whence a language has come and whither it is going, phonetically speaking. This holds particularly true for systems based on sequence-to-sequence transformers and Connectionist Temporal Classification (CTC) which purposely lose temporal information to minimize orthographic loss.

End-to-end approaches have achieved success in their targeted benchmarks. However, flattening the hierarchical structure inherent in language to an opaque, time-agnostic stream has made systematic, linguistic analysis difficult: an impoverishment for the field given that human language processing seems to work on multiple temporal scales (Lyon, 2017). Temporal reconstructions offered by or recoverable with the current state of the art provide imprecise or inaccurate timestamps at the phonetic level despite public interest in this level of detail.

A corollary to the aforementioned quip arises: linguistically agnostic state-of-the-art methods offer diminishing utility to the linguist. A recent project sought to eliminate the need for "onerous expertise" (Omnilingual ASR team et al., 2025) and initially misclassified institutional languages as highly endangered. Status information was ultimately erased for all languages. Rather than omitting this meaningful data, revisions from expert linguists could have directed attention to languages with immediate needs. Sharing hard-won expertise from the full spectrum of participants and disciplines would better advance the field.

Given recent achievements of transformers in orthographic transcription, researchers have attempted to pivot to the more exacting task of phonemic transcription. The lack of expert phonemic transcriptions has led to reliance on graph-to-phoneme (G2P) converters for experiment ground truths. These supposedly time- and labor-saving approaches have incurred penalties in accent bias, language bias, and temporal inaccuracy. Identifications from these systems tend to favor the North American English phonemic inventory despite training on other languages, and phonemes unseen in training cannot be generated *ex nihilo*.

All this fails to address a fundamental distinction in multilingual applications: phoneme vs. phone. Phonemic segmentation implies a particular language or variant at the morphophonological level, which in turn can lead to orthography. Phonetic segmentation, its messier cousin, offers practical insight into the production and articulation of sounds (phones) on a universal, human level.

Findings of the Association for Computational Linguistics: ACL 2026, pages 35639–35652
July 2-7, 2026

Detection of formants, or salient frequencies that characterize human speech, has been a staple of phonetic research. While primarily used for vowels, they can also distinguish other phones. This work presents a novel algorithm for estimating human speech formants based on a concept of spectral gravity, that is, the relative strength of clusters of spectral energy attenuated by the square of distance across the spectrum. Our approach suits real-time applications with deadlines in the tens of milliseconds and avoids the typical pitfalls of expectation maximization (EM): slow convergence, heavy initialization dependence, and fragility against sparse sample data. The system ingests raw audio and outputs time-stamped phone segments with descriptive labels based on constituent formants encoded as 64-bit integers per Yantosca and Cheng (2025).

2 Related Work

Numerous methods for the critical task of formant estimation have been developed over the years. Several (e.g., Ma et al., 1993; Alku et al., 2013) improve upon the linear predictive coding (LPC) analyses advanced by Atal (1975) and Atal and Schroeder (1978). Boersma and Weenink's (2023) popular speech analysis program Praat uses Burg's (1975) maximum-entropy LPC method.

Shadle et al. (2016) comprehensively analyzed the measurement errors induced by different techniques noting that most methods suffered "large errors in the direction of the strongest harmonic." Weighted linear prediction-attenuated main excitation (WLP-AME) (Alku et al., 2013) and pruned reassigned spectrograms (RS) (Fulop, 2011) best mitigated these errors.

However, these methods may incur untenable costs for real-time, streaming applications. Shadle et al. (2016) note that all but one of the methods tested were "semiautomatic," and the best performing method, RS, was fully manual. The runner-up, WLP-AME, requires accurate identification of glottal closure instants, a difficult problem in itself.

Gaussian mixture models (GMMs) offer another way to model formants as spectral energy clusters. Many EM-based GMM component estimators have been proposed in the literature (e.g., Melnykov and Melnykov, 2012; Paalanen et al., 2006; Zhang et al., 2004; Figueiredo and Jain, 2002). The simplicity of implementation has popularized this family of maximum likelihood estimation methods. However, convergence time varies greatly, and local, non-global optima can dominate the search depending on initialization. Executing multiple EM runs competitively can mitigate this at the expense of increased computation time.

Accordingly, some researchers have sought alternatives to EM. Wu and Yang (2020) presented a denoised method of moments estimator, but constraining variance as either known or common fails to account for formant bandwidths as uncommon unknowns. Hosseini and Sra (2020) described an alternative to EM using Riemannian optimization. The method has attractive convergence properties, but a sparse, univariate spectral decomposition may not accommodate the required reformulation of the problem into a smooth manifold.

In light of these difficulties and the recent successes achieved with deep neural networks and sequence-to-sequence transformers on end-to-end speech recognition tasks, several groups have retrofitted and trained their models to support phonemic transcription. Xu et al. (2021) took the orthographically effective Wav2Vec2 framework by Baevski et al. (2020) and trained on a broad corpus of G2P-generated phonemic transliterations of orthographic text for the Wav2Vec2Phoneme model. The phoneme sequences produced were plausible but limited to the phoneme inventories of the training data, favoring phoneme distributions from languages with the most training representation. Timing estimates have to be derived *post hoc* based on the sample rate and each phoneme's single frame of identification in the sequence.

Li et al. (2020) debuted Allosaurus, a universal phone recognizer to produce timestamped phone sequences for many languages. Explicit start times per phone are given with duration estimates, but the authors explicitly relate in their documentation that timestamps are "provided by the CTC model, which might not be accurate in some cases due to its nature" (Li et al., 2021). Empirically, phone durations appear estimated at a constant 45 ms, presumably the observation frame size.

Omnilingual ASR team et al. (2025) released Omnilingual ASR in November 2025 with the promise to extend recognition to new languages with a few in-context samples. The cited CER of below 10% on 1570 languages still leaves a long tail of about 345 languages failing to meet the self-imposed quality threshold. The recommended mode for zero- or few-shot language introduction seems to be predicated on an apparent recency bias toward "context examples with higher text similar-

ity to the target" to avoid out-of-distribution script errors. Although Omnilingual ASR targets orthographic transcription, we may see future phonemic transcription bootstrapping instructions to pair sounds with corresponding International Phonetic Alphabet (IPA) or romanized representations.

Forced-alignment continues to play an active role in transcription. Omnilingual ASR prepared data for languages without sentence-level annotation using a method by Pratap et al. (2024). Linguists and phoneticians use the Montreal Forced Aligner (MFA) (McAuliffe et al., 2017) directly to generate timestamped phone sequences. However, forced alignment tends to misalign long utterances, and some implementations fail to tolerate noise.

3 Methodology

We hypothesized that accurate formant estimation could serve as a basis for linguistically sound phone segmentation and identification. In reviewing Yantosca and Cheng's (2025) recent work on real-time phonetic segmentation, we observed a tendency for formant estimates to cluster below 1 kHz as evidenced in Fig. 5 of the paper. The admitted faults in the formant picker are particularly apparent for sibilants. Despite the majority of spectral energy residing in the higher frequencies, e.g., 3–6 kHz, detected formants fell below 2 kHz.

These low formant values appear to have stemmed from the spectral density heuristic used, which walked up the spectrum in a single pass and averaged areas of high spectral density to derive the formant estimate. A single wide formant band striped with "whitespace" due to a high noise filter would be miscategorized as a set of multiple formants. Since the algorithm exited once the quota of 4 formants was reached, the higher bands were frequently not considered. This was most evident for sibilants and other fricatives.

Our approach sought to build on and improve the work by Yantosca and Cheng (2025) by applying a deadline-bounded expectation maximization algorithm (cf. Algorithms 1, 2, 3) to derive a univariate GMM characterization of each frame of the pyknogram (cf. Potamianos and Maragos, 1995; Shokouhi and Hansen, 2017) generated by the band-estimator stage. The ESTIMATE function takes the following arguments: a deadline d_{EM}, a maximum epoch count E, initial cluster mean hypotheses $\boldsymbol{\mu}$, variance hypotheses $\boldsymbol{\sigma^2}$, mixture probability hypotheses $\mathbf{p}$, pyknogram frequencies $\mathbf{f}$ as sample locations, and pyknogram amplitudes $\mathbf{a}$ as sample weights. The function returns revised hypotheses along with execution metrics.

Algorithm 1 Deadline-Bounded Univariate EM

procedure ESTIMATE($d_{\text{EM}}, E, \boldsymbol{\mu}, \boldsymbol{\sigma^2}, \mathbf{p}, \mathbf{f}, \mathbf{a}$)
  ▷ NB: Log space variables underlined for clarity.
  $\delta_T \leftarrow 0$ ▷ Elapsed time delta.
  $c \leftarrow 0$ ▷ Worst Observed Execution Time.
  $\underline{\lambda} \leftarrow \infty$ ▷ Log likelihood.
  $\underline{\delta_\lambda} \leftarrow \infty$ ▷ Log likelihood delta.
  $e \leftarrow 0$ ▷ Epoch counter.
  while $e < E$
  $\wedge \quad \delta_T + c < d_{\text{EM}}$
  $\wedge \quad \underline{\delta_\lambda} > 10^{-10}$
  $\wedge \quad$ ISNORMAL($\min(\boldsymbol{\sigma^2})$) **do**
    $\underline{\lambda_0} \leftarrow \underline{\lambda}$
    $t_0 \leftarrow$ GETMICROS()
    $\underline{\lambda}, \boldsymbol{\Gamma} \leftarrow$ ESTEP($\boldsymbol{\mu}, \boldsymbol{\sigma^2}, \mathbf{p}, \mathbf{f}, \mathbf{a}$)
    $\underline{\delta_\lambda} \leftarrow |\underline{\lambda} - \underline{\lambda_0}|$
    $\boldsymbol{\mu}, \boldsymbol{\sigma^2}, \mathbf{p} \leftarrow$ MSTEP($\boldsymbol{\mu}, \boldsymbol{\sigma^2}, \mathbf{p}, \mathbf{f}, \mathbf{a}$)
    $\delta_t \leftarrow$ GETMICROS() $- t_0$
    $c \leftarrow \max(\delta_t, c)$
    $\delta_T \leftarrow \delta_T + \delta_t$
    $e \leftarrow e + 1$
  end while
return $\underline{\lambda}, e, \delta_T, \boldsymbol{\mu}, \boldsymbol{\sigma^2}, \mathbf{p}$
end procedure

The ESTEP function uses the amplitude weight vector $\mathbf{a}$ to avoid numerical instability from non-contributing samples, returning the estimate's overall log-likelihood $\underline{\lambda}$ and sample responsibilities $\boldsymbol{\Gamma}$.

The MSTEP function takes the responsibilities derived from ESTEP and applies the amplitude vector $\mathbf{a}$ per Frisch and Hanebeck (2021) to a weighted maximization step, returning the mixture means $\boldsymbol{\mu}$, variances $\boldsymbol{\sigma^2}$, and membership probabilities $\mathbf{p}$.

Algorithm 3 Weighted Maximization Step

procedure MSTEP($\boldsymbol{\mu}, \boldsymbol{\sigma^2}, \mathbf{p}, \boldsymbol{\Gamma}, \mathbf{f}, \mathbf{a}$)
  for $k \in [0, |\boldsymbol{\mu}|)$ **do**
    $\mu_k \leftarrow 0$
    $\sigma_k^2 \leftarrow 0$
    if $p_k > 0$ **then**
      $p_k \leftarrow 0$
      for $i \in [0, |\mathbf{f}|)$ **do**
        if $a_i > 0$ **then**
          $r \leftarrow f_i - \mu_k$
          $p_{k,i} \leftarrow \Gamma_{k,i} a_i$
          $p_k \leftarrow p_k + p_{k,i}$
          if $p_k > 0$ **then**
            $\mu_k = \mu_k + p_{k,i} r / p_k$
            $\sigma_k^2 = \sigma_k^2 + p_{k,i} r (f_i - \mu_k)$
          end if
        end if
      end for
      $\sigma_k^2 = \sigma_k^2 / p_k$
    end if
  end for
return $\boldsymbol{\mu}, \boldsymbol{\sigma^2}, \mathbf{p}$
end procedure

Algorithm 2 Expectation Step

procedure ESTEP($\boldsymbol{\mu}, \boldsymbol{\sigma^2}, \mathbf{p}, \mathbf{f}, \mathbf{a}$)
  ▷ NB: Log space variables underlined for clarity.
  $\underline{\lambda} \leftarrow 0$ ▷ Log likelihood.
  $\boldsymbol{\Gamma} \leftarrow \mathbf{0}$ ▷ Sample responsibilities.
  for $k \in [0, |\boldsymbol{\mu}|)$ **do**
    $\underline{\xi_k} \leftarrow \log \sqrt{2\pi\sigma_k^2}$
    $\underline{p_k} \leftarrow \log p_k$
    for $i \in [0, |\mathbf{f}|)$ **do**
      if $p_k > 0 \wedge a_i > 0$ **then**
        $r \leftarrow f_i - \mu_k$
        $\underline{\Gamma_{k,i}} \leftarrow \underline{\xi_k} + \underline{p_k} - r^2/2\sigma_k^2$
      end if
    end for
  end for
  for $i \leftarrow 0 \rightarrow |\mathbf{f}|$ **do**
    if $a_i > 0$ **then**
      $\underline{\gamma} \leftarrow \max_{k=1}^{|\boldsymbol{\mu}|} \underline{\Gamma_{k,i}}$
      $\underline{\lambda_i} \leftarrow \underline{\gamma} + \log \sum_{k=1}^{|\boldsymbol{\mu}|} \exp\left(\underline{\Gamma_{k,i}} - \underline{\gamma}\right)$
      $\underline{\lambda} \leftarrow \underline{\lambda} + \underline{\lambda_i}$
      for $k \in [0, |\boldsymbol{\mu}|)$ **do**
        $\Gamma_{k,i} \leftarrow \exp\left(\underline{\Gamma_{k,i}} - \underline{\lambda_i}\right)$
      end for
    end if
  end for
return $\underline{\lambda}, \boldsymbol{\Gamma}$
end procedure

These functions returned closely matching centroids with simulated GMM components, but numerical collapse against sparse pyknogram data required additional guards (cf. Algorithm 1, line 10). To avoid spurious centroids, we took inspiration from Melnykov and Melnykov's (2012) method of progressively removing points during initialization. However, their algorithm required running multiple EM or EM-like steps to split and merge proposed clusters, which our time budget could not accommodate due to the attendant latency. Consequently, we replaced the closest neighbors metric with spectral gravity measures taken at each contributing point with respect to all the other points in the system.

In early development, we found that random or equidistant initialization tended to gravitate toward the area around F0/F1 with the highest concentrations of absolute energy. The spectral gravity initialization method came out of our best intuition on what might constitute a global view of the spectral energy landscape. We wanted to give the best opportunity for mid-spectrum formants (e.g., F2, F3) in vowels like /i/ and /ɪ/ to be recognized despite the relative post-filtration scarcity of neighbors.

We modeled our system on Newton's law of universal gravitation (Newton, 2009 [orig. 1686]) in order to assign the most weight to concentrated clusters of spectral energy and quickly diminish the impact of spectrally distant points. While quadratic in time and space complexity, time savings can be gained in the average case due to the sparsity of a pyknogram appropriately filtered to remove undesired noise.

To avoid missing critical formant centers whose neighbors had been filtered out, we oversampled the maximum cluster count $K = 4$ by a factor of 2 and selected the most probable clusters, merging and updating candidate probabilities per Algorithm 4, the gist of which follows.

Algorithm 4 Spectral Gravity Initialization for EM

procedure SPECTRALGRAVITYINIT($\theta_g, \boldsymbol{f}, \boldsymbol{a}$)
  ▷ NB: Log space variables underlined for clarity.
  $\underline{\boldsymbol{a}} \leftarrow \log \boldsymbol{a}$
  $\boldsymbol{\mu}, \boldsymbol{\sigma^2}, \mathbf{p}, \mathbf{g}, \mathbf{G} \leftarrow \mathbf{0}$
  for $i \in [0, |\boldsymbol{f}|)$ **do**
    for $j \in [0, |\boldsymbol{f}|)$ **do**
      if ISNORMAL(a_i) $\wedge$ ISNORMAL(a_j) **then**
        if $i = j$ **then**
          $G_{i,j} \leftarrow \exp \underline{a_i}$
        else if ISNORMAL($f_i - f_j$) **then**
          $G_{i,j} \leftarrow \exp(\underline{a_j} 2 \log |f_i - f_j|)$
        end if
      end if
    end for
    $g_i \leftarrow g_i + G_{i,j}$
  end for
  $k \leftarrow 0$
  while $k < 2K$ **do**
    $\omega \leftarrow \text{argmax } \mathbf{g}$
    if $g_\omega = 0$ **then break**
    $p_k, \mu_k, \sigma_k^2 \leftarrow 0$
    for $i \in [0, |\boldsymbol{f}|)$ **do**
      if $\neg$ISNORMAL(a_i) $\vee\ g_i = 0$ **then continue**
      if $G_{\omega,i} < \theta_g$ **then continue**
      for $j \in [0, |\mathbf{f}|)$ **do**
        if $g_j > 0$ **then** $g_j \leftarrow g_j - G_{j,i}$
      end for
      $p_k \leftarrow p_k + \exp \underline{a_i}$
      $\mu_{k0} \leftarrow \mu_k$
      $\mu_k \leftarrow \exp(\underline{a_i}) |f_i - \mu_k| / p_k$
      $\sigma_k^2 \leftarrow \exp(\underline{a_i}) |f_i - \mu_k| |f_i - \mu_{k0}|$
    end for
    if $p_k = 0$ **then continue**
    if $|\mu_k - \mu_j| < \sigma_j \mid j < k$ **then**
      Merge with the closest cluster.
    else
      Create a new cluster and increment k.
    end if
  end while
  Sort clusters by descending $p_m | 0 \le m < k$.
  Zero out clusters with index $m > \min(K, k)$.
  Reassign equal p_m to valid clusters.
  Reorder valid clusters by ascending μ_m.
end procedure

1. Spectral gravity is measured at each contributing pyknogram sample from the perspective of

an observer with unit "mass." These pairwise gravity contributions are memoized.

2. Until the quota of $2K$ centroids (twice the desired number) has been reached or all samples have been exhausted, the sample with the maximum remaining gravity is selected.

3. All samples with gravity "pull" from this selected sample greater than the threshold θ_g are removed from consideration and add their "mass" to the cluster.

4. The cluster around the selected sample is merged if it falls within the standard deviation radius of an extant cluster. Otherwise, it forms a new cluster and increments k.

5. Once the search finishes, the most probable K or fewer clusters are designated as formants and reordered by ascending mean and given equal probability to prevent stagnation when the EM algorithm is applied. Each bandwidth radius is normalized as $\max(\sigma_k, \mu_k/4)$ to cover singleton clusters with zero variance.

The choice of $K = 4$ stems from practical considerations. $K < 4$ would yield lower homogeneity with identifications concentrating around F0 and F1, e.g., at $K = 2$. $K > 4$ would likely suffer more oversegmentation, possibly at every frame.

Additionally, we simplified Yantosca and Cheng's (2025) one-hot similarity tactic for reducing oversegmentation by applying the following inequality against the similarity floor θ_1:

$$\theta_1 \leq 1 - \frac{\sum_{b=1}^{B} (f_{b,t-1} > 0) \oplus (f_{b,t} > 0)}{\max_{b=1}^{B} b \mid f_{b,t-1} > 0, f_{b,t} > 0} \quad (1)$$

If the count of band flips between frames drives the quantity below θ_1, a formant check decides similarity. In the formula, the numerator sums the band flips from hot to cold or vice versa. The denominator is the index of the highest hot band. Using the highest hot band across both frames as the denominator helps distinguish voiced sounds whose lack of high frequency energy might mask salient differences at lower frequencies.

Yantosca and Cheng's (2025) original method counted one-hot similarity across all bands without distinction, which led to an almost guaranteed minimum 50% similarity at 8 KHz Nyquist frequency for voiced sounds. Using the highest hot frequency as the cutoff for similarity comparison picks up on more minute differences between neighboring voiced sounds.

4 Experimental Setting

Table 1 describes the environment used for testing and comparing Xu et al.'s (2021) Wav2Vec2Phoneme (W2V2P) `wav2vec2-xlsr-53-espeak-cv-ft` model derived from XLSR-53 (Conneau et al., 2021), Li et al.'s (2020) Allosaurus (Allo) model, McAuliffe et al.'s (2017) Montreal Forced Aligner (MFA) v3.3.8, and Yantosca and Cheng's (2025) ARTIC/Phonotomizer (A/P) v3.0.13 augmented with the formant estimation algorithms described in this paper.

Parameter	Value
OS	Ubuntu 24.04
CPU	Intel® Core™ i9-10900X, 3.70 GHz
Cores	10 cores × 2 threads
Memory	128 GB
Build System	GNU `make` + `g++` 13.3.0

Table 1: Experimental System Specifications

The Mozilla Common Voice project (version: 22.0-2025-06-20) supplied Irish, Twi, and Votic sound samples (The Mozilla Foundation, 2025). The project's public availability and commitment to data sovereignty of contributors (e.g., the ability to withdraw contributions upon request) coupled with the ease of use and organization of data led us to select these datasets for our experiments.

To support post-hoc evaluation, gold labels were manually transcribed as TextGrids in Praat (Boersma and Weenink, 2023). Irish transcriptions covered at least one clip per speaker of the "train" and "dev" partitions. Twi transcriptions fully covered the "train" and "other" partitions of the 2022 version of the corpus, though the 2025 version altered the clip partitioning and added new speakers. Votic transcriptions nearly covered the "train" and "test" partitions excepting two "train" clips which induced physical pain in the listener due to distortion artifacts and one "test" clip which was inadvertently omitted. Table 2 quantifies the transcription distribution.

For Phonotomizer, we varied one-hot similarity and pairwise gravity thresholds and EM deadlines and fixed all other hyperparameters per Table 3. Parameter fixations were based on the results and discussion from Yantosca and Cheng (2025) and observations made during development. Focusing on the formant estimation performance, we tested

Language	Partition	Clips	Speakers
Irish	train	56 / 916	37 / 37
Irish	dev	55 / 739	51 / 51
Twi	train	204 / 205	1 / 1
Twi	test	8 / 20	2 / 8
Twi	other	16 / 47	1 / 10
Votic	train	94 / 96	1 / 1
Votic	test	6 / 7	3 / 3

Table 2: Manual Transcription Coverage

Phonotomizer's online-training, zero-shot mode but did not train language-specific models.

Parameter	Value
Band spacing	Logarithmic
Band count (B)	160
Filter order (O)	4
DESA Algorithm	DESA-1
Confidence threshold (F_b)	implicitly derived
Noise floor (ν)	0.01
Sample Rate (r)	16 kHz
Data frame size (N_t)	160 samples (10 ms)
One-hot similarity (θ_1)	0.75, 0.85, 0.99
Pairwise gravity (θ_g)	$10^{-6}, 10^{-7}, 10^{-8}$
EM deadline (d_{EM})	200, 500, 1000 (μs)

Table 3: Phonotomizer Evaluation Parameters

Because Wav2Vec2Phoneme does not provide explicit timing information, we calculated the timing from the model's sample rate (16 kHz) and frame size (20 ms) and correlated the frame identifiers with frame index time derivations. Allosaurus provides timestamps, but some segments may overlap, which the Praat TextGrid format does not admit. Because the durations reported by Allosaurus are constant (45 ms) regardless of actual phone length, we favored the start timestamp and truncated durations which bled into subsequent phones. Otherwise, we used the given duration since we had no alternative basis for calculating phone duration.

For MFA, we trained on the given Mozilla Common Voice dataset partitions separately and evaluated the training alignments, skipping cross-validation to provide steel man baselines. Similarly, we used the Allosaurus `aka`, `gle`, and `vot` language-specific models on Twi, Irish, and Votic, respectively. Although we used the base model for W2V2P, the training of XLSR-53 included an earlier release of the Irish corpus.

5 Results

5.1 Model Performance

In order to compare the generated alignments, we followed Yantosca and Cheng (2025) in using the clustering metrics of completeness (c), homogeneity (h), and normalized mutual information (NMI). An overview of performance is given in Table 4.

Model	c	h	NMI
Allo	0.388±0.095	0.608±0.287	**0.811**±0.369
W2V2P	0.540±0.046	0.133±0.026	0.216±0.033
MFA	**0.773**±0.079	0.604±0.090	0.678±0.081
A/P	0.637±0.082	**0.746**±0.154	0.677±0.080

Table 4: Average Clustering Metrics Overview

One sees the detriment to alignment of discarding timing information. Wav2Vec2Phoneme's poor homogeneity is likely a penalty for marking so many speech frames as padding. In general, we found that the identified phone frames seemed to correspond with the end of the identified phone, but there was no reference back to its start. While a heuristic method might be devised to derive the start times, the information lost due to conflation of silence with speech frames cannot be systematically recovered. In any case, Wav2Vec2Phoneme did not employ such a heuristic.

Allosaurus favors homogeneity and has the best NMI score, but its standard deviation exceeds the others by an order of magnitude, suggesting inconsistent performance. The starting timestamp and constant duration provided by Allosaurus seemed to yield better results than Wav2Vec2Phoneme, but phones do not have uniform duration, even within the same phone. In the Votic samples, we frequently observed epenthesis, i.e., insertion of barely perceptible "helper" vowels to aid articulation. Fast or truncated speech was also common in the Irish samples. Such speech features may get missed at a 45 ms or even 20 ms granularity.

After establishing optimal values for the variable parameters, we examined the performance across all four models. Fig. 1 corroborates Table 4. Phonotomizer's performance clusters around a high homogeneity with comparatively better completeness than Allosaurus or Wav2Vec2Phoneme.

Allosaurus appears slightly more consistent on completeness but shows volatility on homogeneity. Wav2Vec2Phoneme exhibits extremely poor homogeneity and consistent but mediocre completeness.

The linguistically oriented models have the best alignment coherence, though Phonotomizer and MFA optimize for different trade-offs as evidenced by the mutual reflection across the diagonal. MFA's performance depended heavily on the training partition size with a break point requiring around 100 samples to achieve best performance. Phonotomizer's zero shots required no pre-training.

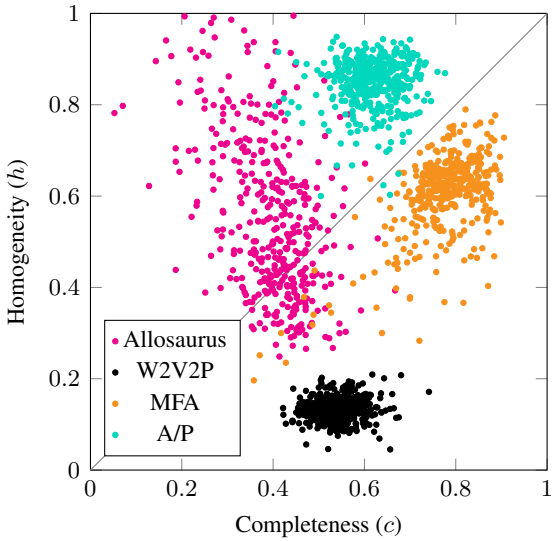

Figure 1: Completeness (c) vs. Homogeneity (h) Across All Models. Upper right corner is best. A/P and MFA outperform Allosaurus and W2V2P in alignment coherence but differ in trade-off philosophy.

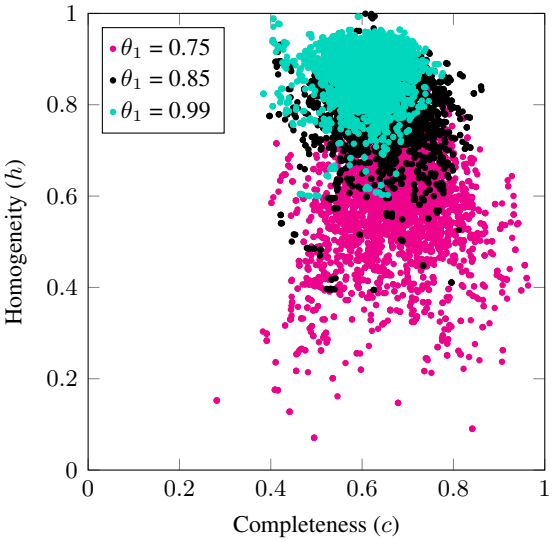

Figure 2: A/P Completeness (c) vs. Homogeneity (h) Varying the One-Hot Similarity Threshold (θ_1). Setting $\theta_1 = 0.99$ scores highest most consistently.

5.2 Optimal Parameter Selection

To inform our optimal parameter selection, we visualized the hyperparameter variation's trade-off between over- and undersegmentation as a 2D plot of corresponding pairs of completeness and homogeneity scores. Completeness (c), which diagnoses oversegmentation, occupies the horizontal axis, and homogeneity (h), which diagnoses undersegmentation, occupies the vertical. Toward the upper right corner indicates better performance with (1,1) indicating perfect completeness and homogeneity.

We observed the greatest distinction when varying the one-hot similarity threshold θ_1 as depicted in Fig. 2. The tightest, most performant grouping occurs when $\theta_1 = 0.99$. The centers of the other clusters have slightly better completeness, but the lack of consistency and significantly poorer homogeneity recommends the usage of $\theta_1 = 0.99$. The highness of this optimal value seems to indicate an improved robustness in the formant similarity check in addition to the more adaptable formula this paper introduces.

Examining the EM algorithm deadline variance shown in Fig. 3, one has difficulty distinguishing an advantage for any particular value. This suggests that the spectral gravity initialization method picks the targets well enough that the EM converges quickly in the average case. However, the tested dataframe size of 10 ms leaves little overhead room, so in the interest of timeliness, the 200 μs deadline would probably be best, especially since the algorithm has to run twice: once for the incoming frame and again for the accumulated candidate segment.

Drilling down further, Fig. 4 illustrates the effect of varying the pairwise gravity threshold θ_g, controlling for θ_1 and d_{EM} at their respective optimal values. Distinction is again difficult, but a slight trend favors completeness as θ_g diminishes to increase inclusion and favors homogeneity as θ_g increases. Intuition would commend the median $\theta_g = 10^{-7}$, but given Phonotomizer's struggle with oversegmentation, biasing toward the slightly better completeness of $\theta_g = 10^{-8}$ produces the best visual coherence (cf. Figs. 5, 6, and 7).

The formant selections show dramatic improvement compared to Yantosca and Cheng (2025). The relative emptiness due to filtration of the 4–8 kHz range for voiced sounds supports the changes to the one-hot similarity formula in this paper. The observed oversegmentation primarily seems to impact nasals, rhotics, and approximants, which can traverse critical bands of hearing and manifest as diagonal spectral bands (e.g., between 1–2 kHz around 1.75 s for the /w/ in "asamoa" in Fig. 5). One of the challenges for nasals seems to be the relative absence of spectral energy and consequently distinguishable formants, especially in word-initial position (cf. the /m/ and /n/ in "minim" between 0.8–1.1 s in Fig. 5). In these cases, the revised one-hot similarity metric may be working against our intent by drastically the reducing the envelope of

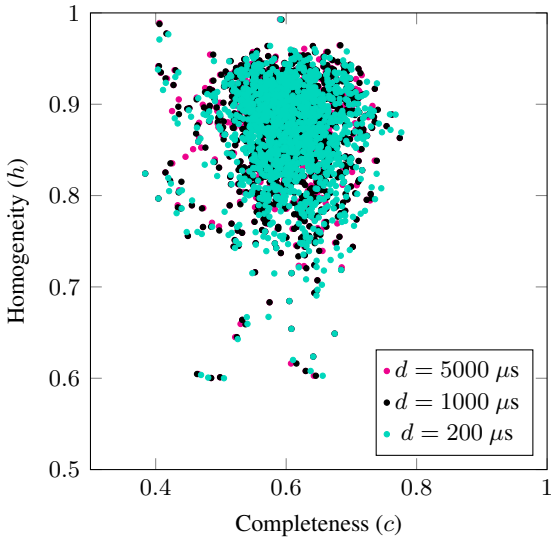

Figure 3: A/P Completeness (c) vs. Homogeneity (h) Varying the EM Algorithm Deadline. Deadline setting seems to have little discernible effect on outcome when controlling for θ_1 at its optimal value, suggesting initialization places the formant centers close to convergence. $d_{\text{EM}} = 200\ \mu$s offers more headroom for 10 ms frames.

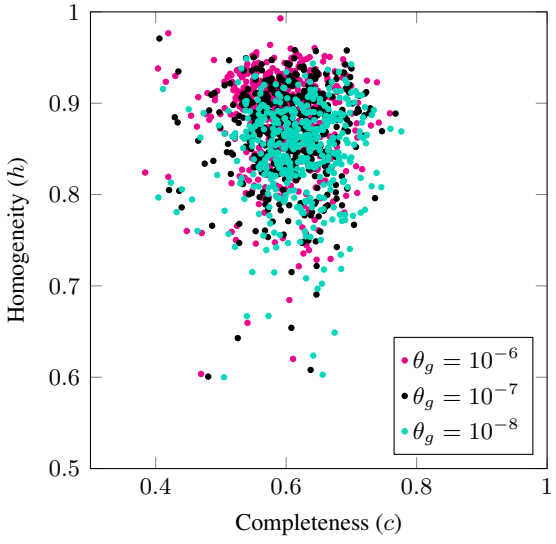

Figure 4: A/P Completeness (c) vs. Homogeneity (h) Varying the Pairwise Gravity Threshold θ_g. In terms of scoring, $\theta_g \propto 1/c$ and $\theta_g \propto h$ when controlling for θ_1 and d_{EM}. $\theta_g = 10^{-8}$ gives Phonotomizer the best completeness boost to combat oversegmentation.

consideration. Characterizing formants as bivariate Gaussian mixtures whose 2nd dimension measures the formant frequency center step change might offer some improvement.

6 Conclusions

The spectral gravity-based algorithm introduced in this paper enhanced the formant estimation and consequently the segmentation alignment compared to the results achieved by Yantosca and Cheng (2025) and the current state of the art. Some improvements remain for phones with unstable or traveling formants which defy simple, scalar characterization. In future work, we intend to investigate methods for tracking and incorporating these step changes.

Despite the recent interest in making this problem space tractable, research efforts in this vein seem to have flagged. Allosaurus has not seen updates in 5 years, and public inquiries into tools for phonemic and phonetic segmentation remain unanswered (e.g., Anonymous, 2022a; Anonymous, 2022b; Anonymous, 2023).

The accent bias of solutions that rely on G2P converters to establish ground truths indicates a need for further research into phonetic segmentation and classification. The literature does not discuss mitigation strategies for this issue, and without strong interdisciplinary collaboration with linguists and other experts, computer scientists working in this space may not be aware of these sources of bias.

As a case in point, Wav2Vec2Phoneme never identified Twi's voiceless alveolo-palatal fricative /ɕ/ as such out of the 52 instances of the phone we observed in our gold manual transcriptions (105 including the affricate /tɕ/ in the count). It consistently substituted some other phone, most frequently /ʃ/. Similarly, we noticed that Wav2Vec2Phoneme favored /o/ over /ɔ/ across the board in both Twi and Votic, despite the latter's phonological importance in Twi and our observation of at least one Votic speaker who consistently rendered the "oo" orthographic digraph as /oɔ/.

We consider the problem space vital to increasing scientific understanding of human speech and linguistic processes and predicting future language trends, e.g. rhotacization, metathesis, and compensatory vowel lengthening. Both human and machine audition depend on accurate timing, and as this work shows, advancement in speech analysis occurs when the research question asks not only what but also when.

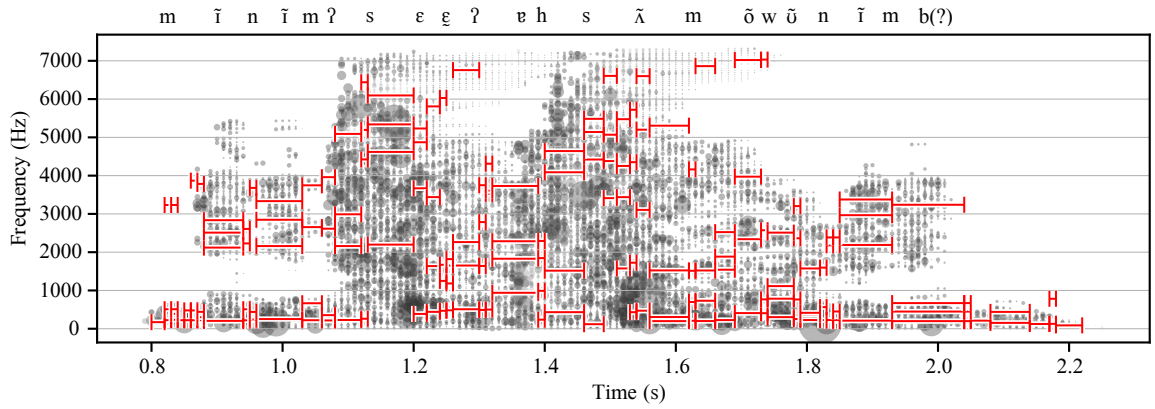

Figure 5: Pyknogram of Twi utterance “minim sɛ asamoa nim” (English: “I know that heaven knows.”) in grayscale. $\theta_1 = 0.99$, $\theta_g = 10^{-8}$, $d_{\text{EM}} = 200$ μs. Formant selections at segmentation intervals in red. The segmentation and formant selection exhibit material improvement compared to Yantosca and Cheng (2025), and the revised formant estimation permitted the noise floor (ν) to drop from 0.05 to 0.01 for a more detailed spectrogram.

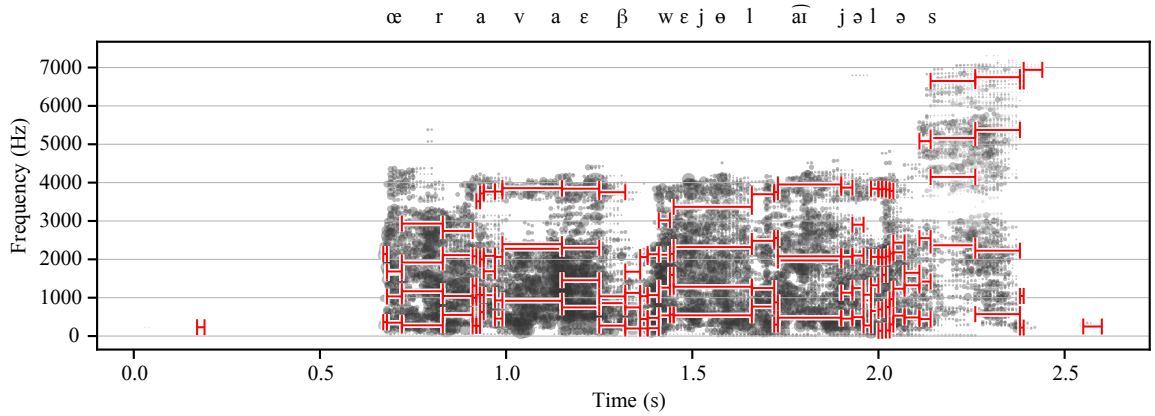

Figure 6: Pyknogram of Votic utterance “õravõ eb õõ laivõz” (English: “There is no squirrel in the boat.”) in grayscale. $\theta_1 = 0.99$, $\theta_g = 10^{-8}$, $d_{\text{EM}} = 200$ μs. Formant selections at segmentation intervals in red. English translation pieced together from grammatical notes by Markus and Rozhanskiy (2022) and the online machine-readable Votic dictionary (Ylonen, 2025) generated by Wiktextract (Ylonen, 2022). The hypothesized second /l/ in “laivõz” may in fact be /β/, but the clip quality makes determination difficult. Spurious formant estimates outside the primary area of interest indicate noise which exceeded the noise floor.

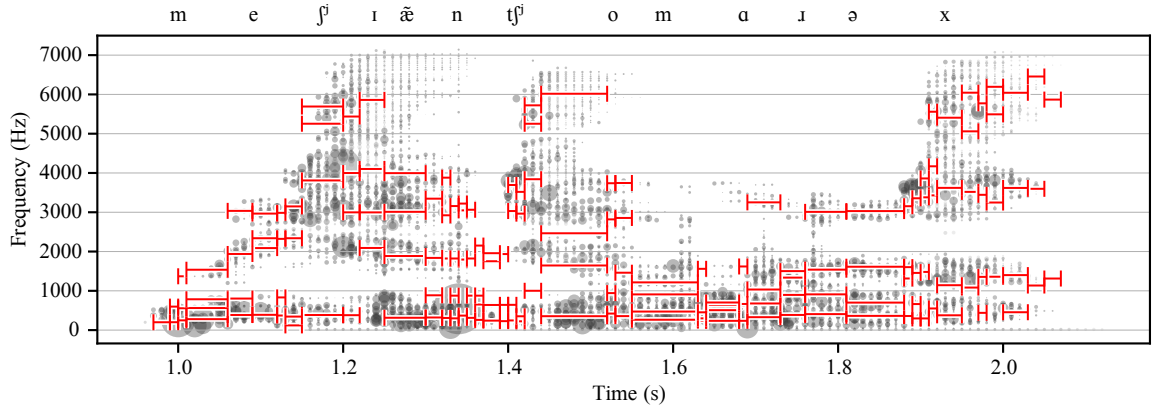

Figure 7: Pyknogram of Irish utterance “An mbeidh sé anseo amárach?” (English: “Will he be here tomorrow?”) in grayscale. $\theta_1 = 0.99$, $\theta_g = 10^{-8}$, $d_{\text{EM}} = 200$ μs. Formant selections at segmentation intervals in red. The speaker completely elides the particle “an.” On a clip like this, a G2P converter would insert extra phones, and text-dependent aligners would likely derail. Multiple blip artifacts contributed to Phonotomizer’s oversegmentation.

7 Limitations

7.1 Gold Label Accuracy

Gold labels were transcribed solely by the primary author. Manual transcription is inherently subjective, and one would ideally employ multiple transcribers including native speakers to normalize inter-transcriber disagreement. However, the authors did not have funding at the time of writing to support a fair wage for transcription.

The gold label datasets are tracked for source control purposes via `git`. Abbreviated commit hashes of the revisions used in experiments are listed below:

- Irish (`goldgrids-gle`): `fef9c60`
- Twi (`goldgrids-twi`): `773e89e`
- Votic (`goldgrids-vot`): `36d9449`

7.2 Experimental Parameters

Testing was executed solely on Linux (Ubuntu 24.04). Phonotomizer introduced limited support for execution on MacOS and Windows via Docker in v3.0.10, but the lack of support for micro-benchmark timing on AMD or `aarch64` processors precluded examination of the deadline-bounded EM algorithm on these platforms and architectures.

Originally, we had run our experiments against ARTIC/Phonotomizer v3.0.12 but discovered a micro-benchmark timing bug that inflated the crystal clock frequency by two orders of magnitude due to an apparent misreading of the source for the Linux kernel project's `turbostat` utility (Torvalds et al., 2025), conflating the crystal clock frequency derived from `cpuid 015h` with the base CPU frequency derived from `cpuid 016h`. The system we tested does not provide crystal clock frequency programmatically, so the value must be inferred from the Intel Software Developer's manual (Intel Corporation, 2022).

A second bug masked this discrepancy by coercing the value into a 32-bit unsigned integer, thereby truncating it to the expected gigahertz range. Consequently, when we re-ran the experiments with ARTIC/Phonotomizer v3.0.13, which fixed the bug, the results were virtually indistinguishable in the aggregate, validating our prior analysis, albeit with slightly improved stage timing for the v3.0.13 runs since the TSC frequency value rose from 2.864 GHz (truncated from an erroneous 569.8 GHz) to the correct 3.696 GHz for the system tested.

ARTIC/Phonotomizer v3.0.13 also added provisional support for AMD processors via empirical TSC frequency sampling per Downs (2024) since reading the requisite machine specific registers (MSRs) would have required root permissions (McCalpin and Downs, 2023). Support for `aarch64` processors was added via the `cntfrq_el0` (Arm Limited, 2025a) and `cntvct_el0` (Arm Limited, 2025b) instructions.

Initial validations on a laptop with an AMD Ryzen 7 chip and on a Raspberry Pi 500 desktop kit yielded reasonable timings, but we did not run the entire set for the sake of brevity. We plan to extend our testing to these platforms in future work.

7.3 Competitive Baselines

The evaluations of MFA only included training alignments, and Phonotomizer was only run in zero-shot mode. In future work, we would like to extend our evaluation suite to more comprehensive partition cross-validation and cross-lingual tests. In order to sufficiently explain the novel algorithms introduced in this paper, we constrained our focus during evaluation accordingly.

7.4 Speaker Diversity and Dataset Composition

The Irish Common Voice dataset had a balanced speaker count between the "dev" and "train" datasets. The lower resource Twi and Votic Common Voice datasets only had a handful of speakers. The "train" partitions of Twi and Votic each had samples from one speaker who had made the majority of contributions to their respective data sets.

Since Mozilla Common Voice datasets are comprised purely of volunteer contributions, it is difficult to control for speaker diversity, especially for languages as critically endangered as Votic.

Although Mozilla has recently taken steps to ensure a better balance between partitions, there is still room for improvement on partitioning, validation, and reporting of errors or offensive language. In one case with the Twi dataset, we noted a clip whose reason for getting flagged was the Twi variant to which it belonged. Two of the phrases marked offensive in the Irish dataset seem to be false positives unless there is a cultural context of which the authors are unaware. The phrases themselves appear to be stock sayings that appear routinely in dictionaries and grammars. Nevertheless, we did not reiterate any speech that could potentially be considered offensive in the paper.

8 Ethical Considerations

Anonymous contributors to Mozilla Common Voice datasets (The Mozilla Foundation, 2025) may withdraw their data, but no such requests were received for Irish, Twi, or Votic. In accord with the license, no diarization was attempted.

To further protect the anonymity of the data, statistics were given as broad aggregates, and points in scatter plots of the clustering metrics were presented without identifying labels. Salient spectrogram examples only listed the speaker's language, phonetic transcription, orthographic transcription, and English translation without identifying the Mozilla Common Voice speaker ID or the dataset partition to which the clip belonged, and the sentences were all from scripted sentences determined by the project, as opposed to the recently introduced spontaneous speech collections available with some datasets.

No large language models (LLMs) were employed to write or edit this paper nor to develop the formant estimation algorithm or any other original code used in experiments.

The authors followed the ethical considerations established by Yantosca and Cheng (2025) with respect to usage of third-party research products.

Acknowledgments

The authors wish to thank Harald Hammarström and Viktor Martinović for their assistance on collecting language documentation for the critically endangered Votic language, as well as Suzanne Kemmer and Caroline Crouch for their input on linguistic and phonetic considerations.

References

Paavo Alku, Jouni Pohjalainen, Martti Vainio, Anne-Maria Laukkanen, and Brad H. Story. 2013. Formant frequency estimation of high-pitched vowels using weighted linear prediction. *The Journal of the Acoustical Society of America*, 134(2):1295–1313.

Anonymous. 2022a. Software to translate audio to phonemic transcription. Reddit discussion.

Anonymous. 2022b. Transcribe to IPA (international phonetic alphabet). OpenAI Whisper forum discussion.

Anonymous. 2023. Speech to phonetic transcription: Does it exist? Reddit discussion.

Arm Limited. 2025a. *CNTFRQ_EL0, Counter-timer Frequency Register*.

Arm Limited. 2025b. *CNTVCT_EL0, Counter-timer Virtual Count Register*.

Bishnu S. Atal. 1975. Linear prediction of speech—recent advances with applications to speech analysis. *Speech Recognition*, pages 221–230.

Bishnu S. Atal and M.R. Schroeder. 1978. Linear prediction analysis of speech based on a pole-zero representation. *The Journal of the Acoustical Society of America*, 64(5):1310–1318.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. Wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in Neural Information Processing Systems*, 33:12449–12460.

Paul Boersma and David Weenink. 2023. Praat: doing phonetics by computer. Software program.

John Parker Burg. 1975. *Maximum Entropy Spectral Analysis*. Stanford University.

Alexis Conneau, Alexei Baevski, Ronan Collobert, Abdelrahman Mohamed, and Michael Auli. 2021. Unsupervised cross-lingual representation learning for speech recognition. *Interspeech 2021*.

Travis Downs. 2024. avx-turbo. GitHub repository.

Mario A. T. Figueiredo and Anil K. Jain. 2002. Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):381–396.

Daniel Frisch and Uwe D. Hanebeck. 2021. Gaussian mixture estimation from weighted samples. In *2021 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 1–5. IEEE.

Sean A. Fulop. 2011. *Speech Spectrum Analysis*. Springer Science & Business Media.

Reshad Hosseini and Suvrit Sra. 2020. An alternative to EM for gaussian mixture models: batch and stochastic Riemannian optimization. *Mathematical Programming*, 181(1):187–223.

Intel Corporation. 2022. *Intel® 64 and IA-32 Architectures Software Developer's Manual*.

Peter Ladefoged. 1993. *A Course in Phonetics*. Harcourt Brace Jovanovich.

Xinjian Li, Siddharth Dalmia, Juncheng Li, Matthew Lee, Patrick Littell, Jiali Yao, Antonios Anastasopoulos, David R. Mortensen, Graham Neubig, Alan W. Black, and Metze Florian. 2020. Universal phone recognition with a multilingual allophone system. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8249–8253. IEEE.

Xinjian Li, Siddharth Dalmia, Juncheng Li, Matthew Lee, Patrick Littell, Jiali Yao, Antonios Anastasopoulos, David R. Mortensen, Graham Neubig, Alan W. Black, and Metze Florian. 2021. Allosaurus. GitHub repository.

Richard F. Lyon. 2017. *Human and machine hearing*. Cambridge University Press.

Changxue Ma, Yves Kamp, and Lei F. Willems. 1993. Robust signal selection for linear prediction analysis of voiced speech. *Speech Communication*, 12(1):69–81.

Elena Markus and Fedor Rozhanskiy. 2022. Votic. In Marianne Bakró-Nagy, Johanna Laakso, and Elena Skribnik, editors, *The Oxford Guide to the Uralic Languages*, chapter 19, pages 330–349. Oxford University Press.

Michael McAuliffe, Michaela Socolof, Sarah Mihuc, Michael Wagner, and Morgan Sonderegger. 2017. Montreal forced aligner: Trainable text-speech alignment using Kaldi. In *Proc. Interspeech 2017*, pages 498–502.

John D. McCalpin and Travis Downs. 2023. Find alternative to get_TSC_frequency for AMD processors. GitHub issue (low-overhead-timers).

Volodymyr Melnykov and Igor Melnykov. 2012. Initializing the EM algorithm in Gaussian mixture models with an unknown number of components. *Computational Statistics & Data Analysis*, 56(6):1381–1395.

Isaac Newton. 2009 [orig. 1686]. Philosophiae naturalis principia mathematica. Project Gutenberg.

Omnilingual ASR team, Gil Keren, Artyom Kozhevnikov, Yen Meng, Christophe Ropers, Matthew Setzler, Skyler Wang, Ife Adebara, Michael Auli, Can Balioglu, Kevin Chan, Chierh Cheng, Joe Chuang, Caley Droof, Mark Duppenthaler, Paul-Ambroise Duquenne, Alexander Erben, Cynthia Gao, Gabriel Mejia Gonzalez, Kehan Lyu, Sagar Miglani, Vineel Pratap, Kaushik Ram Sadagopan, Safiyyah Saleem, Arina Turkatenko, Albert Ventayol-Boada, Zheng-Xin Yong, Yu-An Chung, Jean Maillard, Rashel Moritz, Alexandre Mourachko, Mary Williamson, and Shireen Yates. 2025. Omnilingual asr: Open-source multilingual speech recognition for 1600+ languages. *Preprint*, arXiv:2511.09690.

Pekka Paalanen, Joni-Kristian Kamarainen, Jarmo Ilonen, and Heikki Kälviäinen. 2006. Feature representation and discrimination based on Gaussian mixture model probability densities—practices and algorithms. *Pattern Recognition*, 39(7):1346–1358.

Alexandros Potamianos and Petros Maragos. 1995. Speech formant frequency and bandwidth tracking using multiband energy demodulation. In *Proc. 1995 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 1, pages 784–787.

Vineel Pratap, Andros Tjandra, Bowen Shi, Paden Tomasello, Arun Babu, Sayani Kundu, Ali Elkahky, Zhaoheng Ni, Apoorv Vyas, Maryam Fazel-Zarandi, et al. 2024. Scaling speech technology to 1,000+ languages. *Journal of Machine Learning Research*, 25(97):1–52.

Fedor Rozhanskiy and Elena Markus. 2017. On the mid and high non-labial vowels in luuditsa votic. *Linguistica Uralica*, 53(4):241–255.

Christine H. Shadle, Hosung Nam, and D.H. Whalen. 2016. Comparing measurement errors for formants in synthetic and natural vowels. *The Journal of the Acoustical Society of America*, 139(2):713–727.

Navid Shokouhi and John H. L. Hansen. 2017. Teager-Kaiser energy operators for overlapped speech detection. 25(5):1035–1047.

The Mozilla Foundation. 2025. Mozilla Common Voice. Orthographically annotated datasets.

Linus Torvalds, Patryk Wlazlyn, and Len Brown. 2025. Linux kernel source tree: linux/power/tools/turbostat. GitHub repository.

Yihong Wu and Pengkun Yang. 2020. Optimal estimation of Gaussian mixtures via denoised method of moments. *The Annals of Statistics*, 48(4):1981–2007.

Qiantong Xu, Alexei Baevski, and Michael Auli. 2021. Simple and effective zero-shot cross-lingual phoneme recognition. *arXiv preprint arXiv:2109.11680*.

Michael S. Yantosca and Albert M.K. Cheng. 2025. Phonotomizer: A compact, unsupervised, online training approach to real-time, multilingual phonetic segmentation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12135–12147.

Tatu Ylonen. 2022. Wiktextract: Wiktionary as machine-readable structured data. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1317–1325.

Tatu Ylonen. 2025. Votic word forms. Online machine-readable dictionary.

Baibo Zhang, Changshui Zhang, and Xing Yi. 2004. Competitive EM algorithm for finite mixture models. *Pattern recognition*, 37(1):131–144.

A Appendix

To quantify the impact of one-hot similarity and formant estimation changes, we ran Phonotomizer v3.0.6 as well against the dataset from Table 2 with the parameters from Yantosca and Cheng (2025) ($\nu = 0.05$, $\theta_1 = 0.5$) and this paper ($\nu = 0.01$, $\theta_1 \in \{0.75, 0.85, 0.99\}$) as shown in Table 5.

Version	θ_1	g	ν	n	c	h	NMI
v3.0.13	0.75	10^{-6}	0.01	1320	0.665±0.097	0.586±0.112	0.618±0.086
		10^{-7}		1320	0.669±0.096	0.565±0.116	0.607±0.090
		10^{-8}		†1317	0.672±0.099	0.554±0.104	0.603±0.083
	0.85	10^{-6}		1320	‡0.629±0.070	0.818±0.074	0.710±0.051
		10^{-7}		1320	0.640±0.069	0.800±0.072	0.710±0.052
		10^{-8}		1320	0.650±0.070	0.779±0.072	0.708±0.052
	0.99	10^{-6}		1320	0.591±0.057	0.886±0.056	0.709±0.045
		10^{-7}		1320	0.603±0.056	**0.872**±0.054	0.714±0.044
		10^{-8}		1320	0.614±0.057	0.854±0.053	0.715±0.044
v3.0.6	0.50	N/A	0.05	440	**0.676**±0.060	0.805±0.074	**0.734**±0.046
	0.50		0.01	440	0.630±0.067	0.837±0.040	0.720±0.047
	0.75			440	0.554±0.054	0.913±0.040	0.691±0.044
	0.85			440	0.522±0.049	0.936±0.042	0.671±0.043
	0.99			440	0.516±0.049	0.940±0.043	0.668±0.042

† One quiet clip did not generate segmentations and was not scored. All versions performed poorly on this clip.

‡ Rows highlighted in yellow indicate where parameters yielded generally coherent segmentations with best scores in bold.

Table 5: Average Phonotomizer Clustering Metrics by Version and Parameter Selection

Phonotomizer v3.0.6 with Yantosca and Cheng's (2025) original parameters achieves the best Normalized Mutual Information (NMI) and completeness with high homogeneity. This paper's optimal parameter choice has the best NMI for v3.0.13.

Homogeneity decays in v3.0.13 at $\theta_1 = 0.75$ whereas completeness in v3.0.6 decays for $\theta_1 \geq 0.75$. Qualitatively sampling the v3.0.6 segmentations with $\theta_1 \geq 0.75$ found oversegmentation at nearly every frame. Despite the high homogeneity, the lower NMI contraindicates these parameter values. The one-hot similarity check dominated the segmentation decision with the formant estimation preserving continuity against transient band noise and providing the classification basis.

The v3.0.13 formant estimates surpass the naive, one-pass approach of v3.0.6, especially for high-frequency fricatives like sibilants. The band estimator stage's noise and harmonic suppression may incur discontinuity across quiescent bands and induce early termination of the v3.0.6 linear formant search. The v3.0.13 spectral gravity method offers a more holistic view of the energy distribution.

Fig. 8 depicts the average formants with standard deviation error bars for maximally distinct pairs of fricatives and vowels. The discriminability by formant is apparent in the v3.0.13 estimates. Some v3.0.6 formant averages sink below their predecessors (e.g., F4 < F3), most notably in Votic.

In v3.0.13, the high formants of /s/ and lower formants of /h/ reflect the respective vocal tract openness. The low F1 of /i/ and the high F1 of /ɑ/ demonstrate the inverse proportionality of F1 to vowel height. The high F2 of /i/ and relatively lower F2 of /ɑ/ also match expectations of direct

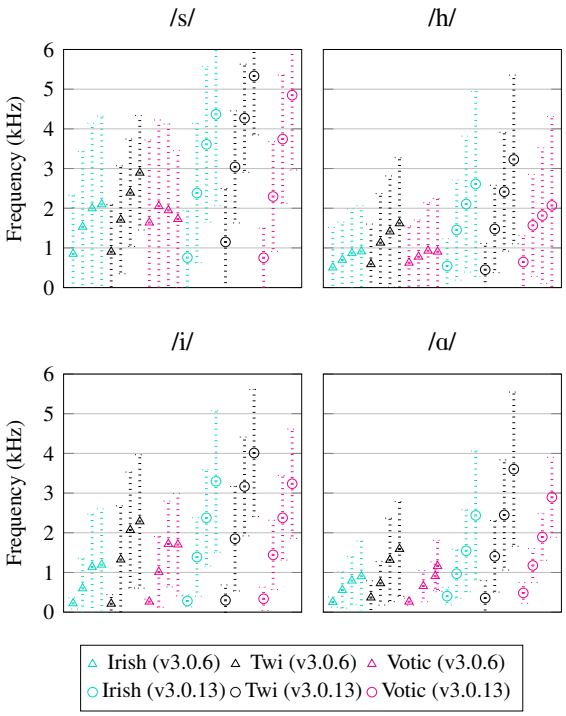

Figure 8: Average Formant Estimates (F1–F4) by Gold Phone, Phonotomizer Version, and Language. Averages were derived from deinterleaving classification labels and do not represent per-frame estimates. F1–F4 appear from left to right within each version/language pair. v3.0.13 surpasses v3.0.6 in discriminability and accuracy, but variance is still high, possibly from spurious formant selection or classification bleed across gold phone boundaries. The relative distinctions between fricative and vowel pairs follow expectations.

proportionality to the degree of fronting (Ladefoged, 1993) (Rozhanskiy and Markus, 2017).

Although there is room for improvement, v3.0.13 presents a more accurate and linguistically coherent picture of the formant soundscape than v3.0.6. The groundwork laid here should facilitate better results in future experiments.