Language identification and accent variation detection in spoken language recordings

Shyamal Buch[†], **Jon Gauthier**[‡], **Arthur Tsang**[†]
{*shyamalb, jgauthie, atsang2*}@*stanford.edu*
[†] Computer Science Dept., [‡] Symbolic Systems Program
Stanford University

***Abstract*—We develop a model for identifying languages and accents in audio recordings. Our Hierarchical-Sequential Nodule Model (HSNM) incorporates both short-distance features (which capture simple linguistic distinctions, e.g. phoneme inventories) and long-distance features (which detect long-distance suprasegmental patterns, e.g. tone and prosody) which help a classifier discriminate intelligently among two or more languages.**

We apply this model to the language identification (LID) and accent detection (AD) tasks, and investigate the potential for simple knowledge transfer between the two tasks. We demonstrate that the HSNM model performs well on binary and multi-class classification tasks in LID and AD.

I. Introduction

A. Motivation

Language identification (LID) systems are utilities for determining the language being spoken in an audio recording. Many modern LID systems operate on audio inputs of variable length, and output a single decision or a probability distribution over the possible languages being spoken in the recording [1]. Applications for LID systems include emergency call routing and multilingual translation systems [1], [2].

Accent detection (AD), a variant of language identification, can be used in speech training and also to improve accuracy of LID systems, which can suffer large accuracy losses on accented speech [3]. Accented corpora can be small relative to traditional speech corpora, due to the higher cost of accurate data collection [3]. Thus, data-light methods to improve accent detection systems can be of value for the task.

B. Prior work

Research on the LID task can be broadly divided into two approaches. The first approach focuses on building generative language-dependent models, passing audio through language-specific tokenizers or speech recognizers in order to approximate the probability of a particular recording being in a particular language [4], [5]. Systems built in this way yield high-accuracy results, but are slow and likely difficult to maintain. Such pipeline systems also have a critical failure point in that errors committed by components early in the pipeline — e.g., tokenizers which convert audio into phone transcriptions — can yield unrecoverable system-wide failures.

The second approach is a lower-level solution, where language-independent audio features (and complicated conjunctions of such features) are the primary source of information for classification [6], [7]. This method is more appealing from an engineering perspective, as it can be highly portable and robust among different linguistic environments.

Corpus	German			Mandarin		
	Train	Dev	Test	Train	Dev	Test
OGI	478	195	188	462	188	182
CSLU	202	56	67	170	55	57

TABLE I: Number of recordings per split in datasets used. OGI refers to the language identification data, and CSLU refers to the accent identification data. CSLU numbers are approximate.

There has also been significant prior work in the field of accent detection, particularly in the specific problems of Shanghai-accented Chinese [3] and North vs. South-accented American English [8]. These approaches typically rely on Gaussian Mixture Models (GMMs), and adopt the second approach described above.

C. Our work

This paper's objectives are as follows:

1) Develop a model that can capture distinguishing language patterns with only language-agnostic audio-based features.
2) Develop an LID system which integrates the model to give accurate classifications.
3) Develop an Accent Detection (AD) system which incorporates both the model and the corresponding LID model to leverage language data for additional accuracy.

We situate ourselves firmly on the second side of the dichotomy mentioned earlier, employing only language-independent features drawn directly from the audio (along with conjunctions and synthetic forms of those features). Section III gives a detailed description of the model we develop centered around such features.

We apply this system first to binary LID classification between German and Mandarin audio recordings. German and Mandarin differ in everything from basic linguistic details like phoneme inventory to suprasegmentals and long-term patterns like phonotactics (syllable structure), tone (Mandarin has lexical tone), and prosody. For this reason we see the language pair as a useful first target in developing an LID system.

II. Dataset

A. Language Identification

The OGI Multi-Language Telephone Speech Corpus [9] is a collection of thousands of phone-call recordings in 11 languages. The recordings in the corpus vary in duration from 2 seconds to 1 minute, and are either responses to prompts or free speech.

Some of the calls in the corpus have an associated basic phonetic transcription, the result of a process of manual segmentation and

Acknowledgement: Thanks to Sebastian Schuster (Stanford NLP Group) for offering useful advice throughout the course of the project.

Note: Appendix material available upon request.

annotation by linguists. We choose to not use this extra data for the following reasons:

- This data requires expert intervention, and makes generalizing to new languages or different corpora difficult.
- No real-time system could depend on this data, as such annotation would not be available at runtime.
- Annotations are provided for only a subset of all recordings, and it would be impractical to depend on such partial data.

Audio recordings vary in quality. Extremely poor-quality calls are labeled as such in the dataset. We retain low-quality calls in our experiments.

The OGI corpus comes with a predetermined train / development / test split. We use this split in all of our experiments. Table I gives the number of recordings in each split of the dataset for both German and Mandarin.

B. *Accent Detection*

The CSLU Foreign Accented English corpus [10], like the OGI corpus, contains thousands of phone-quality calls. All calls are English-language free speech, by speakers native in each of 22 foreign languages. Some recordings are as short as 4 seconds, but the majority are 20 seconds long.

In addition, for each recording, we have the strength of the accent, as judged by three native English speakers on a 4-point scale ranging from Negligible/No Accent to Very Strong Accent.

We randomly split our dataset into sections of approximately 60%, 20%, and 20%, for train, development, and test respectively.

III. Methodology

A. *Hierarchical-Sequential Nodule Model (HSNM)*

We develop the Hierarchical-Sequential Nodule Model (HSNM) to capture distinguishing linguistic features in audio speech recordings. The model is deliberately designed to capture both short-distance patterns (phoneme distributions, fundamental frequency) and long-distance patterns (phonotactics and syllable structure, tone, prosody, etc.). While we wish to capture such phenomena, we also wish to keep the system language-independent. Language-dependent features are costly to engineer and difficult to maintain. By using only language-independent features drawn from the sound waves of the recording itself, we ensure that extending the system to a different language amounts to simply collecting the requisite audio data.

The HSNM is designed to operate at one abstraction level higher than the low-level speech features typically used in the literature [11]. As such, any standard low-level feature set can be used interchangeably. Typical low-level features include mel-frequency cepstral coefficients (MFCCs), volume, pitch, and others [12].

In order to capture both short- and long-distance phenomena in speech, we aggregate the described low-level audio features in two ways: first *hierarchically*, collecting statistics from feature distributions over multiple slices of audio, and second *sequentially*, by passing forward information from one hierarchical aggregation to the next. Figure 1 shows an overview of the structure of the model.

We first split an audio file into short **segments** of t seconds in duration. The audio recording can then be modeled as a sequence of these segments. We build a sequence of **nodules** above this segment collection. Each nodule N_i has n associated consecutive segments $x_1, \ldots, x_n$ as children. Furthermore, each pair of consecutive nodules N_{i-1} and N_i have s_o number of segments that overlap between the two. We construct a representation of a nodule by combining features from its preceding nodule and its associated child segments:

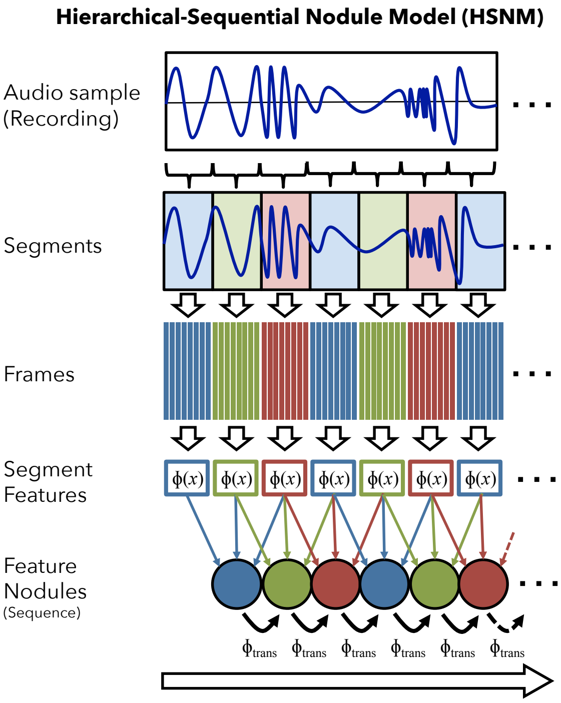


Fig. 1: Hierarchical-sequential nodule model (HSNM) developed for language identification and accent detection tasks. The HSNM synthesizes high-level hierarchical and sequential features from more local audio-based features, themselves drawn from small splits of an audio recording.

$$N_i = \langle n, t, s_o, \phi_{trans}(N_{i-1}), \phi(x_1, \ldots, x_n)\rangle. \qquad (1)$$

Here ϕ_{trans} is a feature extractor function which draws features from a nodule structure.[1] $\phi(\cdot)$ extracts functionals for each feature of the provided segments. Following [11], the two high-level sets of feature aggregations we use for $\phi(\cdot)$ in our experiments are as follows. These functionals are calculated independently for each sequence of values for a feature drawn from the child segments of a nodule.

Basic: Mean, delta (change from first to last segment)
Complex: Basic + standard deviation, mean

Formally, we can describe the HSNM as a tuple $\mathcal{H} = \langle n, t, s_o, \phi, \phi_{trans}, c\rangle$, where $c : N \rightarrow L$ is a classifier described below.

1) Classification: We classify a recording as being in a language $\ell^* \in L$ using a classifier at the nodule level, which maps a single nodule N_i to a predicted language ℓ. The language of the entire recording is the language with the most nodule-level votes. If a recording has k nodules:

$$\ell^* = \arg\max_{\ell \in L} \sum_{j=1}^{k} 1\{c(N_j) = \ell\}. \qquad (2)$$

2) Training: From every recording we construct k nodules from fixed-size audio segments (where k varies based on the length of the recording). Since each recording is labeled with the language being spoken in the audio, we can construct for a given recording with k nodules k different training examples $\{\langle N_i, \ell\rangle\}_{j=1}^{k}$, where ℓ is the language of the overall recording.

[1] For the first nodule, N_1, we assume $\phi_{trans}(N_0) = \vec{0}$.

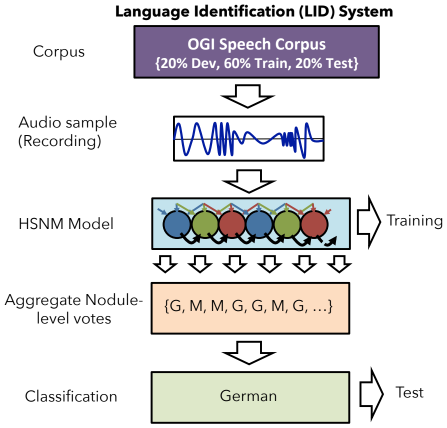

Language Identification (LID) System
Corpus
OGI Speech Corpus
{20% Dev, 60% Train, 20% Test}
Audio sample
(Recording)
HSNM Model
Training
Aggregate Nodule-level votes
{G, M, M, G, G, M, G, ...}
Classification
German
Test

Fig. 2: The LID system based on the HSNM. Speech samples drawn from the corpus are processed into HSNM nodules, which are used at training time. At test time, nodule-level classifications are aggregated to yield an overall classification.

We train the classifier $c : N \to L$ on the concatenation of all training examples drawn from recordings. Note that we train each nodule vote independently of previous nodule vote information. By avoiding constructing a full sequence model over a recording, we keep training highly efficient and simple to implement. Nodules still do carry sequential features (defined by ϕ_{trans}, but the classification at the nodule level is trained without conditioning on previous classifications — a key distinction from a typical sequence model.

B. Language Identification (LID) System

We first perform minor preprocessing on the audio data. We normalize each recording to have a peak amplitude of -3 dB, and drop segments below the required length t (specified as an HSNM hyperparameter). As we discuss in Section IV, we added this preprocessing step after performing some basic error analysis on early output from the model.

We proceed to extract our base low-level audio features. Here, we use a standard feature set that has been demonstrated to work well for speech recognition tasks [12]. We use openSMILE [13] to extract these low-level features, which include MFCCs, pitch, loudness, jitter, and line spectral pairs, among others.

We then apply and fine-tune the HSNM described in the previous section to the task of language identification, asking a system to discriminate between Mandarin and German voice recordings. In this task our classifiers are thus functions $c : N \to L$, where $L = \{\text{GERMAN}, \text{MANDARIN}\}$. We also experiment to find optimal values of the other HSNM hyperparameters $t, n, \phi_{trans}(N_{i-1}), \phi(x_1, \ldots, x_n)$ for our LID system.

We also develop a "smart" baseline in order to *isolate* the effects of the hierarchical and sequential features within the HSNM. The baseline uses the same low-level features as described above to yield language votes at the *segment* level — that is, there is no aggregation of segments into nodules and no synthesis of hierarchical or sequential features.

C. Accent Detection (AD) System

We perform the same preprocessing and feature extraction on the accent audio data as we do on the language ID data (described in the previous section).

We then apply the language ID model constructed in the previous two sections to the task of accent detection. Our classifiers now discriminate among a set of two foreign English accents: $L = \{\text{GERMANENGLISH}, \text{MANDARINENGLISH}\}$.

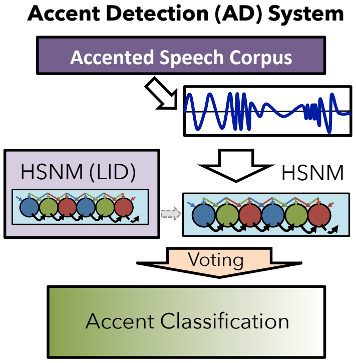


Fig. 3: Condensed schematic of the accent detection system. We apply the same LID sequence-voting model (see Figure 2) to accent data, and leverage LID models to improve performance further.

Model	Dev F1	Test F1
Baseline ($t = 2$)	79.9%	73.9%
Best model ($t = 2$)	82.6%	74.6%
Baseline ($t = 1$)	84.9%	78.9%
Best model ($t = 1$)	**89.9%**	81.7%

TABLE II: Performance of *selected* models on language identification (LID) development and test sets, varying segment duration t. Metric is macro-averaged F1 (over two classes).

We also attempt a simple form of knowledge transfer in order to leverage parameters learned on language identification data for the accent detection task. This development is driven by an intuition that patterns which distinguish a particular language from its peers also appear in the speech of its native speakers in other languages. For example, we expect trends in German prosody to also distinguish German-accented English speech.

With this motivation, we pose a knowledge-transfer experiment where a model trained on accent data is *ensembled* with a corresponding language identification model. We redefine the HSNM as a tuple $\langle n, t, s_o, \phi, \phi_{trans}, C, W \rangle$, where C is now a set of distinct classifiers trained on the same feature sets extracted by ϕ, ϕ_{trans}. W is a collection of per-classifier vote weights (where higher weights indicate greater confidence in the classifier). We determine the most likely accent for a recording by collecting votes from each classifier at each nodule (extending Equation (2)):

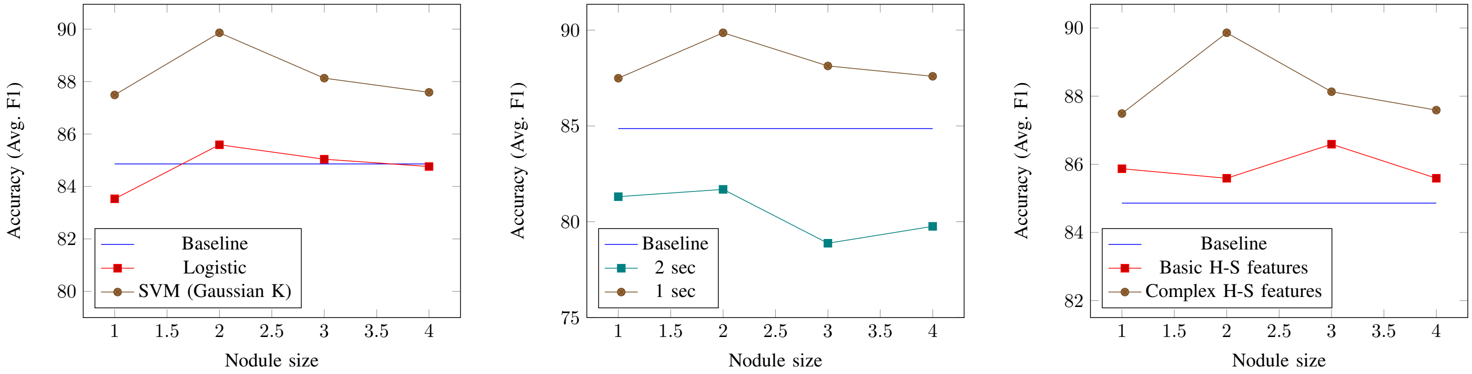
$$\ell^* = \arg\max_{\ell \in L} \sum_{j=1}^{k} \sum_{c \in C} 1\{c(N_j) = \ell\} W_c. \tag{3}$$

In practice, we find including a language ID classifier increases accent detection performance, but only when it is downweighted relative to the main accent detection classifier. See Section IV-B for more information.

IV. Results: Discussion and Analysis

A. Language Identification (LID) Results

Table II shows a brief overview of the improvement our model achieves over the baseline described in Section III, varying the hyperparameter of segment duration t. The following section describes in detail results from several other hyperparameter experiments.



(a) Classifier experiments: SVM outperforms logistic regression by a large margin (otherwise identical feature sets / HSNM configuration)

(b) Segment size experiments: 1-second segments help the HSNM best capture short- and long-distance features useful for language ID

(c) Feature set experiments: complex features yield the greatest performance margin over the baseline (at $n = 2$).

Fig. 4: Experimental results for LID-HSNM system on German vs. Mandarin classification.

Property	% of failed examples w/ property German	Mandarin
Silence	15.4%	10.0%
Speech disfluencies	23.1%	30.0%

TABLE III: Error Analysis — Common properties of recordings on which LID classification failed.

1) Hyperparameter grid search: The HSNM developed in Section III-A has a large number of degrees of freedom. The most significant are:

- c, the classifier used to collect language votes from nodule data;
- t, the duration of each segment drawn from a recording;
- n, the number of segments inherited by each nodule; and
- ϕ and ϕ_{trans}, the feature extractors used to generate hierarchical and sequential features from low-level segment data.

We perform a grid search over the possible combinations of hyperparameters on a held-out development set and report results below, with accompanying visualizations. We vary nodule size and a single hyperparameter in these graphs; unless otherwise mentioned, by default we configure with an SVM, complex features, and $t = 1$ sec.

Figure 4a shows an experiment evaluating different classifiers $c : N \to L$ for discriminating among languages at the nodule level. (The models in this graph all use the ”complex” feature set, with segment duration $t = 1$.) We find that an SVM with a Gaussian kernel offers a significant boost over a baseline.

Figure 4b demonstrates the effect of different segment durations t on development set performance. It is clear that in using 2-second segments, we forfeit the benefits of the HSNM; in fact, it seems we lose information by aggregating over segments of too large a duration. With $t = 1$, however, we see the HSNM features add orthogonal information which increases classification performance.

Figure 4c compares the performance of the two feature sets tested on different nodule sizes. The complex feature set trials outperform by a large margin both the basic feature set trials and the baseline. Note that complex features have a clear peak at $n = 2$, while basic features peak at $n = 3$.

2) Error analysis: Table III presents the results of an analysis of all examples on which one of the best-performing LID systems fails. By far the properties most often appearing in these failed examples are:

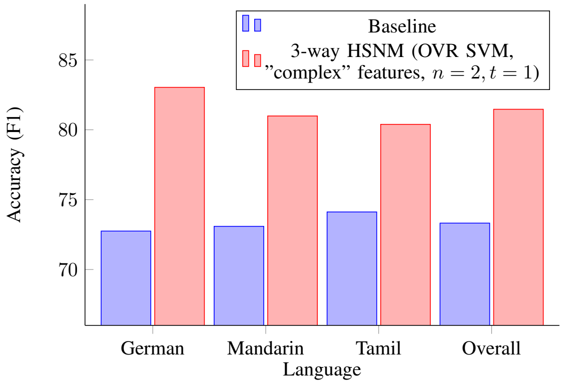


Fig. 5: Results from a proof-of-concept trial suggest that the HSNM-based LID system generalizes well to multi-class classification. Metric is average F1 score.

- **Silence:** sustained pauses in speech, with no to little background noise.
- **Speech disfluencies:** filler words like *uh*, *eh*, etc., which often precede / succeed long silences.

There were 76 total recordings on which the LID system failed for this analysis. We randomly sampled 76 recordings from those on which the LID system succeeded and looked for the same properties as listed above; we confirmed that the failed examples have these properties far more often than the succeeded examples (based on the analysis of the random sample).

Another error not listed above is volume variance. We recognized a significant variation in volume among recordings early on during development, and added a preprocessing step to remove this variation.

3) Multi-class classification: We also validate the performance of the model with a preliminary experiment on a multi-way classification task, drawing on the Tamil portion of the OGI corpus. There is a similar number of recordings in each split of the corpus for Tamil as we’ve seen for German and Mandarin.

Figure 5 shows the performance of the HSNM models trained with a one-versus-rest SVM classifier. We train one-versus-rest logistic regression models which show similar improvements over

Model	Dev F1	Test F1
Baseline ($t = 2$)	70.5%	68.9%
Best model ($t = 2$)	79.4%	68.7%
Baseline ($t = 1$)	68.6%	72.4%
Best model ($t = 1$)	78.4%	73.2%
Ensembling ($t = 2$)	79.5%	76.0%

TABLE IV: Performance of selected models on accent detection (AD). Metric is macro-averaged F1 (over two classes). Ensembling (AD + LID) shows improved performance for both dev and test.

Property	% of failed examples w/ property	
	German Accent	Mandarin Accent
Background sounds	25.0%	10.0%
Southern-influenced accent	12.5%	20.0%

TABLE V: Error Analysis — Common properties of recordings on which AD failed

the baseline (which is the same baseline as used in other LID experiments, re-applied to the three-class problem).

B. Accent Detection (AD) Results

1) General Results: Table IV shows a brief overview of the improvement our model achieves over the baseline described in Section III, varying the hyperparameter of segment duration t. Like with LID, we also performed experiments to fine-tune HSNM hyperparameters, and observed similar trends as before. Full results for all the experiments are in appendix material.

2) Error Analysis: Our system failed on 26 out of 111 recordings on the AD development set. We listened to all 111 recordings, looking for characteristics common in the errors.

A significant portion can be attributed to problems in recording quality, especially background sound. We can approach this issue by more sophisticated preprocessing. In addition, some of the failed examples, which were typically rated by the judges as a Strong accent, sound heavily influenced by Southern US English. Since there is not much Southern-influenced speech in the training data, it makes sense that the system missed these. However, since we use an SVM with a Gaussian kernel, our model should theoretically be able to handle these kinds of examples if given more Southern-influenced training data. See Table V for percentages.

We also notice that the F1 score among recordings rated as Negligible/No Accent by at least one judge is significantly lower than average. For a German Accent, this rate is 69.6% compared to 75.5%, and for a Mandarin accent, this rate is 36.4%, compared to 77.6%.

3) Ensembling: We see that ensembling boosts performance for AD, as shown in Table IV and Figure 6. By admitting weighted votes from the LID model with a $W_{LID} = 0.625$, we see performance improvement in both the dev and test set. This shows promise in the ensembling method.

V. Conclusion: Challenges and Future Work

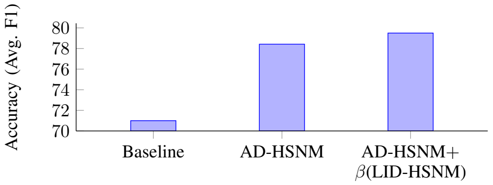


Fig. 6: Results of accent ensembling experiment. We find that $W_{LID} = 0.625$ gives perf. boost on both dev/test.

We find that the HSNM structure manages to successfully encapsulate useful representations of both long- and short-distance language-independent features. We see boosts in performance after adding more nuanced functionals which capture more structural (hierarchical) and time-based (sequential) patterns. Additional experiments with HSNM hyperparameters (segment durations, nodule size, other feature sets) show that performance is indeed dependent on the correct choices; we grid-search over hyperparameters to produce the reported results. Preliminary experiments suggest that the HSNM model generalizes well to multi-class classification; this is likely due to its flexible, language-agnostic structure. We also successfully ensemble an AD model with an LID classifier for the corresponding language, effectively transferring learned knowledge from the LID data to the accent task.

Future work includes further developing our LID and AD system to be even more robust by addressing specific issues found in error analysis, such as silence and disfluencies, and examining potential overfitting problems on the accent corpus. We also would like to quantitatively compare the HSNM with standard models used in speech processing (GMMs, HMMs) and other models such as RNNs to further support the value in the model developed here. We are also interested in investigating other knowledge transfer methods to further enhance the AD system.

References

[1] Y. Muthusamy, E. Barnard, and R. Cole, “Reviewing automatic language identification,” *IEEE SPM*, vol. 11, no. 4, pp. 33–41, Oct. 1994.

[2] E. Ambikairajah, H. Li, L. Wang, B. Yin, and V. Sethu, “Language identification: A tutorial,” *IEEE Circuits and Systems Magazine*, vol. 11, no. 2, pp. 82–108, 2011.

[3] Y. Zheng, R. Sproat, L. Gu, I. Shafran, H. Zhou, Y. Su, D. Jurafsky, R. Starr, and S.-Y. Yoon, “Accent detection and speech recognition for shanghai-accented mandarin,” in *EUROSPEECH 9*, 2005.

[4] M. A. Zissman, “Comparison of four approaches to automatic language identification of telephone speech,” *IEEE Trans. on Speech and Audio Processing*, vol. 4, no. 1, p. 31, 1996.

[5] M. A. Zissman and E. Singer, “Automatic language identification of telephone speech messages using phoneme recognition and n-gram modeling,” in *ICASSP-94.*, vol. 1. IEEE, 1994, pp. I–305.

[6] P. Torres-Carrasquillo, E. Singer, T. Gleason, A. McCree, D. Reynolds, F. Richardson, and D. Sturim, “The MITLL NIST LRE 2009 language recognition system,” in *ICASSP-2010*, Mar. 2010, pp. 4994–4997.

[7] I. Lopez-Moreno, J. Gonzalez-Dominguez, and O. Plchot, “Automatic language identification using deep neural networks,” in *Proc. ICASSP*, 2014.

[8] J. T. Purnell and M. Magdon-Ismail, “Learning american english accents using ensemble learning with GMMs,” in *ICMLA 4*. IEEE, 2009, pp. 47–52.

[9] Y. K. Muthusamy, R. A. Cole, B. T. Oshika, and others, “The OGI multi-language telephone speech corpus,” 1992.

[10] T. Lander, “CSLU: Foreign accented english release 1.2,” *LDC*, 2007.

[11] D. Bone, M. Black, M. Li, A. Metallinou, S. Lee, and S. S. Narayanan, “Intoxicated speech detection by fusion of speaker normalized hierarchical features and GMM supervectors.” in *INTERSPEECH*, 2011, pp. 3217–3220.

[12] B. Schuller, S. Steidl, A. Batliner, F. Burkhardt, L. Devillers, C. A. Mller, and S. S. Narayanan, “The INTERSPEECH 2010 paralinguistic challenge.” in *INTERSPEECH*, 2010, pp. 2794–2797.

[13] F. Eyben, F. Weninger, F. Gross, and B. Schuller, “Recent developments in openSMILE, the Munich open-source multimedia feature extractor,” in *Proc. ACMM 21*. ACM, 2013, pp. 835–838.