Automatic Language Identification of Telephone Speech

Zhirong Yang

Laboratory of Computer and Information Science
Helsinki University of Technology
rozyang@cc.hut.fi

Abstract

The project work involves implementation and comparison of three approaches for automatic language identification of speech utterance: Gaussian mixture model (GMM) classification; single-language phone recognition followed by language-dependent, interpolated n-gram language modeling (PRLM); parallel PRLM, which uses multiple single-language phone recognizers, each trained in different language. Also, the performance by merging multiple language phone recognizers is also investigated.

1. Introduction

The task of determining the language in which a given piece of speech was spoken is important. This is a particular field of speech recognition, which is called language identification or Language-ID. Given a number of languages and their training data, a recognizer is constructed to identify which language is used in a newly input (telephone) speech stream.

The applications of language-ID mainly fall into two categories: pre-processing for matching understanding systems and preprocessing for human listeners, for example, a hotel lobby or international airport in which one might find a multi-lingual voice-controlled travel information retrieval system. Alternatively, language ID might be used to route an incoming telephone call to a human switchboard operator fluent in the corresponding language.

In this paper, I re-implemented and re-tested three approaches (GMM, PRLM and parallel PRLM) of language-ID which were reviewed in [1]. Different from the approaches in that paper, I did the feature extraction and acoustic model training mainly by means of SONIC toolkit.

2. Related work

There are a variety of cues that humans and machines can use to distinguish one language from another. The following characteristics differ from language to language [1]:

- *Phonology*. Phone/phoneme sets are different from one language to another, even though many languages share a common subset of phones/phonemes. Phone/Phoneme frequencies may also differ, i.e. a phone may occur in two languages, but it may be more frequent in one language than the other. Phonotactics, i.e., the rules governing the sequences of allowable phones/phonemes, can be different, as can be prosodics.

- *Morphology*. The word roots and lexicons are usually different. Each language has its own vocabulary, and its own manner of forming words.

- *Syntax*. The sentence patterns are different. Each when two languages share a word, the sets of words that may precede and follow the word will be different.

- Prosody. Duration, pitch, and stress differ from one language to another.

Research in automatic language identification from speech has a history extending back at least twenty years. At present, all automatic language-ID systems of which the author is aware take advantage of one or more of these sets of language traits in discriminating one language from another.

Early automatic language-ID systems are primarily based on static classification, in that the feature vectors are assumed to be independent of each other and no use of feature vector sequences is made. Examples includes using prototypical spectra comparison [2] and vector quantization classification [3].

In recent years, HMM model has been introduced into this field to model the sequential characteristics of speech production. This was first proposed by House and Neuburg [4], and followed by Nakagawa [5] for example.

Language-ID systems trained with multiple languages have been also proposed, for instance, [6]. Multiple recognizers may run in parallel, for example, [7].

3. Data source

The data in my project work came from Oregon Graduate Institute Multi-Language Telephone Speech (OGI-TS) Corpus [8].

The corpus contains speech data from eight languages: English, Hindi, Japanese, Korean, Mandarin, Spanish, Tamil, and Vietnamese. In total there 8,824 speech files, which are all in .wav file format.

However, only a small part of the data is labeled for supervised learning. Thus, the training data includes 522 speech files from five languages: English, Hindi, Japanese, Mandarin, and Spanish. The languages without any labeled data were omitted in this project.

The labeled files are in .ptlola format. A segment of an example .ptlola file is shown as follows:

```
MillisecondsPerFrame: 1.000000
END OF HEADER
3 42 .pau
42 61 D
61 131 i:
131 182 gc
182 210 g
210 291 ^
291 344 v
344 417 3r
417 478 m
478 548 ^
548 616 ^_?
616 725 w
725 784 ^
784 818 l
818 870 bc
870 885 b
885 967 i:
967 1029 pc
1029 1080 ph
1080 1164 3r
1164 1221 v
1221 1371 aI
1371 1415 dc
1415 1434 d
1434 1491 I
1491 1564 dc
1564 1587 D
1587 1639 ^
```

4. Preprocessing

SONIC cannot handle the OGI data directly and some preprocessing is required.

OGI corpus contains .wav files with useless header, which can be removed by the following command:

```
sox infile -w -s outfile
```

Afterwards, 39-dimensional PMVDR feature vectors were extracted from the raw audio content by means of the *fea* program of SONIC toolkit.

Each .ptlola file contains two header lines and a list of label triples. The first two elements are frame numbers, which define the interval of a phone or silence (.pau). The symbols in the third column are different from that used in SONIC and it is difficult to know the exact meaning of all of them. Here I applied simple mapping to convert the symbols to a set of 3-digit hexadecimal numbers.

The first step is to collect all the symbols. In total there 1081 symbols for the five selected languages. On average there are more than two hundred symbols per language. This is too many for SONIC. Therefore the next step is to shorten the symbols: the characters after "_" or ":" will be omitted. This way multiple symbols are mapped to a single hex number and the cardinality of resulting phone set is only 138. On average there are 60 phones per language, which is reasonable for SONIC handling.

A major trick in the project is to treat the phonemes as words. To make use of SONIC for generating such "words" from input speech, something more need to be done. For the decision tree rules, I construct the following simple question list: $silence contains only one element SIL; $others includes all other phones; and each phone has an identical entry of its own. Since each "word" is pronounced as itself, a lexicon file contains two identical columns.

The language models for SONIC were trained by the CMU-Cambridge statistical language modeling toolkit. Before using this toolkit, every symbol list in .ptlola needs to be converted to "word" sequence and concatenated into .text file for training. This is done by the mapping mentioned above except that the symbol .pau was treated as "sentence" mark and converted to ¡s¿ and ¡/s¿.

When creating the Master Label File (.mlf) for 3-state hmm_training, each interval was divided into 3 equal sub intervals and all positions were marked as "b". Because SONIC computes a vector every 10 ms while the unit in OGI label data is 1 ms, the number will be divided by 10 and rounded up by floor function. Notice that there exist incorrect labeled data. By checking the number of last frame and the length of feature files, the abnormal speech were deleted.

5. Classification by GMM

A GMM language-ID system served as the simplest algorithm for this study. GMM language ID is motivated by the observation that different languages have different sounds and sound frequencies.

Under the GMM assumption, each feature vector $\mathbf{o_t}$ at frame time t ($t = 1, \ldots, T$) is assumed to be drawn randomly according to probability density that is

a weighted sum of multi-variate Gaussian densities:

$$p(\mathbf{o_t}|\lambda) = \sum_{m=1}^{N} N_m(\mathbf{o_t}; \mu_\mathbf{m}, \Sigma_m)$$

where $\mu_\mathbf{m}$ and Σ_m are the mean vector and covariance matrix of the m-th Gaussian component; λ is the set of model parameters:

$$\lambda = \{w_m, \mu_m, \Sigma_m\}$$

The GMM model was trained by the following steps:

1. **Initialization**: Randomly generate M vectors as the starting means of the model; Initialize the covariance matrix to identity matrix; and the weights of the Gaussian components are all set to $1/M$.

2. **Likelihood Computation**: Let $b_m(\mathbf{o_t})$ represent the probability of $\mathbf{o_t}$ at the m-th Gaussian component, then:

$$p(\mathbf{o_t}|\lambda) = \frac{w_m b_m(\mathbf{o_t})}{\sum_{k=1}^{M} w_k b_k(\mathbf{o_t})}$$

3. **Parameter Update**:

$$\begin{aligned}
\bar{w}_m &= \frac{1}{T}\sum_{t=1}^{T} p(\mathbf{o_t}|\lambda) \\
\bar{\mu}_m &= \frac{\sum_{t=1}^{T} p(\mathbf{o_t}|\lambda)\cdot \mathbf{o_t}}{\sum_{t=1}^{T} p(\mathbf{o_t}|\lambda)} \\
\bar{\Sigma}_m &= \frac{\sum_{t=1}^{T} p(\mathbf{o_t}|\lambda)\cdot (\mathbf{o_t}-\bar{\mu}_m)\cdot(\mathbf{o_t}-\bar{\mu}_m)^T}{\sum_{t=1}^{T} p(\mathbf{o_t}|\lambda)}
\end{aligned}$$

Due to huge amount of data it is difficult to load all vectors in main memory. The above GMM training algorithm was slightly modified. The speeches were process one by one and only the vectors of current speech were read into main memory. Additionally, the computation of numerators and denominators was done separately.

For each experiment, 1,000 test samples were randomly chosen from all speeches of the five selected languages, either labeled or unlabeled. The feature vectors $O = \{\mathbf{o_t}\}$, $t = 1, \ldots, T$ extracted from a test speech were treated independent of each other. Thus log likelihood of the vectors given the s-th language model is

$$\log P(O)|\lambda_s = \sum_{t=1}^{T} \log P(\mathbf{o_t}|\lambda_s)$$

And the classification result is the most likely model,

$$k = \arg\max_s \log P(O|\lambda_s)$$

The classification accuracies are shown in Table 1.

GMM components	0s-10s	10s-30s	30s-60s
16	61.0%	65.6%	69.2%
40	66.4%	67.4%	70.1%

Table 1: Language ID accuracies by using GMM classifier. The columns are experiments with different lengths of speech (unit: second). The results in the first row are experiments using 16 Gaussian components and the second row for 40 components.

6. Classification by PRLM

After preprocessing described in Section 4, I configured SONIC with non-standard settings to make it a program for English phone recognizer. The output from *sonic_batch* is the phone sequence of a given speech. The back-end of the system is the bigram language models of different languages, which are defined as follows:

$$\tilde{P}(w_t|w_t - 1) = \alpha_2 P(w_t|w_{t-1}) + \alpha_1 P(w_t) + \alpha_0 P_0$$

where w_{t-1} and w_t are consecutive phones observed in the phone stream. The P's are ratios of counts observed in the training data, e.g.:

$$P(w_t|w_{t-1}) = \frac{C(w_{t-1}, w_t)}{C(w_{t-1})}$$

where $C(w_{t-1}, w_t)$ is the number of times phone w_{t-1} is followed by w_t, and $C(w_{t-1})$ is the number of occurrences of phone w_{t-1}. P_0 is the reciprocal of the number of phones. In this project work, the α's are set to the same values as [1], i.e. $\alpha_2 = 0.399$, $\alpha_1 = 0.6$ and $\alpha_0 = 0.001$.

The block diagram of the PRLM by using SOINC is shown in Figure 1.

The sounds in the languages to be identified do not always occur in English front-end phone recognizer. To improve performance, an approach is to run multiple PRLM system in parallel. The test speech is processed by all the PRLM systems and their respective outputs are summed up and the language with maximum score is then returned.

The block diagram of the Parallel PRLM is shown in Figure 2.

The classification accuracies of PRLM and Parallel PRLM are shown in Table 2.

7. Discussion

The cues for language-ID come from both acoustic and lingual information of the input speech. The GMM classifier makes use of only the former source, but the classification accuracies are already much better than those by random guessing (which should be 20%). The PRLM method also takes the language information into account and it outperforms the GMM when the speech is longer than 10 seconds. The Parallel PRLM performs best

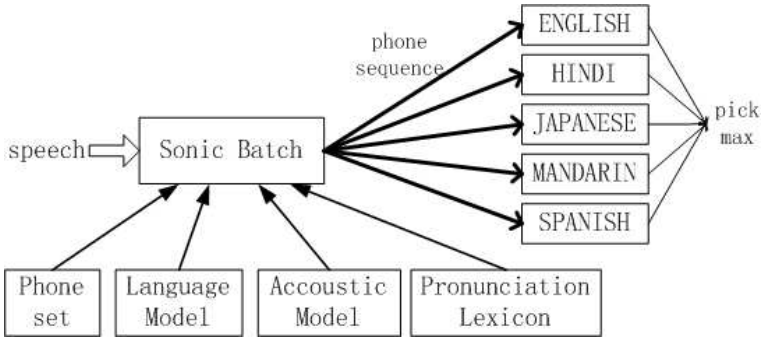

Figure 1: *PRLM block diagram. A single-language recognition front end is used to tokenize the input speech. The phone sequences output by the front end are analyzed, and a language is hypothesized.*

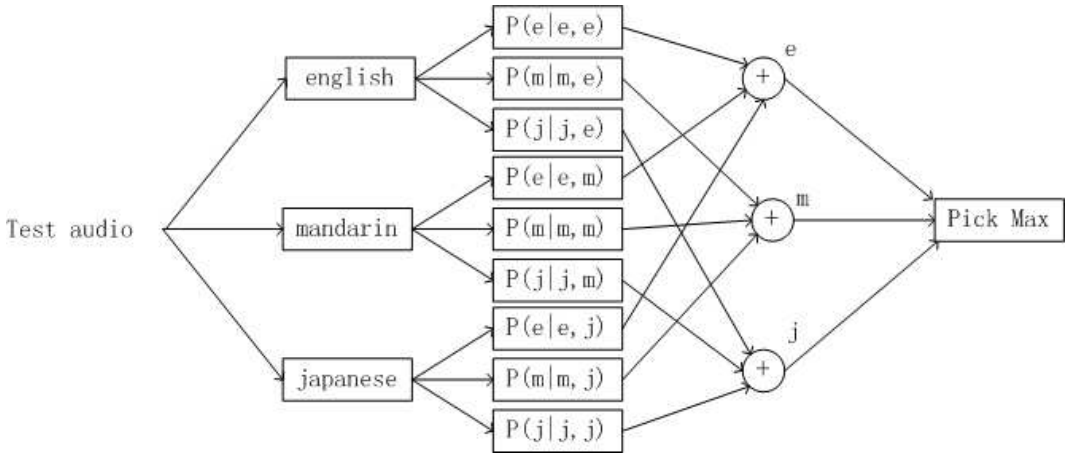

Figure 2: *Parallel PRLM block diagram. For illustrative purpose only three languages are displayed. The letters in the back-end PRLM bigram models represent the respective languages. For example,* $P(j|j,e)$ *means the probability of a sequence is in Japanese given the input is Japanese and with English frontend.*

Type	0s-10s	10s-30s	30s-60s
PRLM	55%	75%	75%
Parallel PRLM	71%	88%	89%

Table 2: Language ID accuracies by using PRLM and Parallel PRLM classifiers. The columns are experiments with different lengths of speech (unit: second). The results in the first row are experiments using PRLM with the English frontend and the second row are results by using Parallel PRLM with all frontends.

among these three methods. The improvement by using Parallel PRLM can be up to 29 percent compared with GMM.

The length of speech also affects the accuracy of recognition. The experiments results reveal that for a speech shorter than 10 seconds it is not easy to determine its language accurately. On the other hand, the accuracies for the speeches longer than 30 seconds are almost the same with those between 10 to 30 seconds. This indicates that a proper length of speech for language-ID might be 10 to 30 seconds.

In this project all the knowledge about the selected languages come from the .ptlola files. However, since it is difficult to understand the exact meaning of all symbols in the .ptlola files, the mapping is done in a simple dummy manner. This would cause some labeled information lost and consequently it is hard to read the output sequence from the frontends by human being.

I configured the SONIC toolkit in a non-standard manner in the project but it turns out that it worked quite well as a frontend program in language-ID application.

8. References

[1] Marc A. Zissman. “Comparison of Four Approaches to Automatic Language Identification of Telephone Speech”, IEEE Transaction on Speech and Audio Processing, Vol. 4, No.1, January, 1996.

[2] R. G. Leonard. “Language recognition test and evaluation”, RADC/Texas Instruments, Inc., Dallas, TX, Tech, Rep. RADC-TR-80-83, Mar. 1980.

[3] M. Sugiyama. “Automatic Language Recognition Using Acoustic Features”, Proc. ICASSP ’91, vol. 2, May 1991, pp. 813-816.

[4] A. S. House and E. P. Neuburg. “Toward Automatic Identification of the Language of An Utterance. I. Preliminary Methodological Considerations”, J. Acoust. Soc. Amer., vol. 62, no. 3, pp. 708-713, Sept. 1977.

[5] S.Nakagawa, T. Seino and Y. Ueda. “Spoken Language Identification by Ergodic HMMs and Its State Sequences”, Electron. Commun. Japan, Pt. 3, vol. 77, no. 6, pp. 70-79, Feb. 1994.

[6] L. F. Lamel and J. -L. Gauvain. “Cross-lingual Experiments with Phone Recognition”, in Proc. ICASSP ’93, vol. 2, Apr. 1993, pp. 507-510.

[7] S. Mendoza. “Private Communication”.

[8] Y. K. Muthusamy, R.A. Cole, and B. T. Oshika. “The OGI multi-language telephone speech corpus”, in Proc. ICSLP ’92, vol. 2, Oct. 1992, pp. 895-898.