Language Identification with Inaccurate String Matching

Kay M. Berkling (berkling@cse.ogi.edu), Etienne Barnard (barnard@cse.ogi.edu)

Center for Spoken Language Understanding,
Oregon Graduate Institute of Science and Technology,
20000 N.W. Walker Road, P.O. Box 91000, Portland, OR 97291-1000, USA

ABSTRACT

We describe a system designed to recognize the language of an utterance spoken by any native speaker over the telephone. The current approach extends our previous work on language-identification based on sequences of speech units [2]. To improve performance we extend this work to allow for inaccurate matches of such sequences. Results are reported for distinguishing between English and German. The strength of this algorithm lies in the generalizability from training to test set. We have obtained a means of discriminating between languages based on statistical derivations. Matching sequences inaccurately in a controlled manner allows us to account for variabilities within languages without sacrificing cross language discrimination.

1. Introduction

We wish to develop an infrastructure for a large scale language identification system which will be able to understand speech in any language and thereby robustly identify the spoken language. By far the best way to identify a language is to understand it [4]. Language understanding is achieved through a combination of vocabulary, grammar, and cultural background and involves a large degree of complexity at each level of modelling.

In order to build such a system, we believe that it is important to systematically reduce the complexity of a language identification system at all levels. In previous work, we have addressed the complexity at the level of multi-lingual speech representation. This paper builds a language identification system which studies the complexities at the "word" level.

We build a statistical model of the frequency, distribution and variability of linguistic units in context of longer sequences. Descriminant features correspond to selected sequences which tend to represent words, sub-words, and frequent, language specific grammatical inflections. Through quantitative analysis we were able to capture regularities within a language and select a minimal set of discriminating features.

In this paper, Section 2 will review the basic system design including the derivation of the multi-lingual speech units. Section 3 will explain the statistical model used for feature selection and the language identification process. Finally, Section 4 will report our results.

2. The System

In this section we will review our baseline language identification system and the speech representation used to align multi-lingual speech input.

2.1. Speech Recognition

The language-identification system used in this study is based on neural networks. Phonemes are recognized by a neural network, followed by a search which aligns phoneme-like labels.

Neural Network Classifiers A neural network is used to assign scores relating the probability of seeing a given speech unit to an input utterance. The neural network classifiers used here are fully-connected, feed-forward, three layer networks trained using back-propagation with conjugate gradient optimization [1] using a mean squared error criterion. Acoustic input is represented with a seventh order Perceptual Linear Predictive (PLP) model [3], yielding 8 coefficients (including one for energy). Coefficients within a 156 msec window, centered on the frame to be classified, were computed and served as input to the phonetic classifier providing substantial contextual information.

Segmenting the Speech Acoustic features as described above are calculated every 6 ms. Thus, the network assigns phoneme category scores to each 6 ms time frame of the utterance. Output scores are computed for each incoming time frame creating a matrix of probability-like scores over time. Speech is segmented into a time aligned string of phonemes by using the optimal path through the outputs of the neural network. Durations are represented as minimum and maximum duration corresponding to the 2nd and 90th percentile

of a histogram computed over all training files for each of the phonemes. A Viterbi search takes duration constraints and transition probabilities into account when searching for the optimal path.

2.2. Speech Representation

Clustering of phonemes across languages is based on the premise that not all phonemes are of equal importance to the language identification task. In fact, decreasing the number of phonemes to be recognized may improve the phoneme recognition accuracy which in turn may improve alignment and language identification by simultaneously decreasing the complexity of the recognizer and increasing the number of training samples for each class. While clustering phonemes it is important not to lose the ability to discriminate languages by pruning the clustering process.

Clustering If we look at the phoneme recognizer as a channel between the acoustics and the Viterbi search, then we want this channel to carry a maximum amount of information about the incoming signal as reflected by the mutual information measure. In order to guarantee an increase in performance of phoneme recognition, we chose at each step to merge the pair of phonemes which maximizes the mutual information. Let $p(x|y)$ be the conditional probability of recognizing y as x after alignment. With $q(y)$ denoting the prior probability of y, $P(x) = \sum_y q(y)p(x|y)$ is the estimated occurrence frequency of x after alignment. The mutual information measure is then given by:

$$MI = \sum_{x,y} q(y)p(x|y)log(\frac{q(y)p(x|y)}{P(x)q(y)}) \quad (1)$$

In practice, we derive $p(x|y)$ by using the confusion matrix which is derived by aligning utterances before clustering and comparing frame-based labels of the aligned files to the hand labeled files. Deriving the prior probability $q(y)$ from the labeled files, the mutual information between the observed and actual phonemes can now be calculated.

Pruning While increasing alignment accuracy by clustering phonemes it is important to retain discrimability between languages. We can estimate the discrimination error at each level of clustering. Because sequences in both languages can now be expressed in terms of these derived speech units, we have gained the ability to automatically select discriminating features consisting of the occurrence frequencies of sequences of speech units. The resulting algorithm, based on an optimally chosen and weighted set of such features, allows us to theoretically predict the language classification error. By estimating the error at each level of merging we can disallow the mergers of phonemes which decrease the discrimination ability between a given set of languages.

2.3. Feature Selection

Features for language identification consist of sequences of these derived speech units. For each sequence we derive an estimate of the language discrimination error. We assume normal distribution of the occurrence frequencies for each sequence i in language l:

$$N(\bar{u}[i]_l, s[i]_l^2) = N(\bar{u}[i]_l, s1[i]_l^2 + s2[i]_l^2) \quad (2)$$

where (n is the number of speakers). The two components of the variance $s[i]_l^2$ represent the variability due to the length of the utterance and the speaker dependent variability and are expressed as follows:

$$s1[i]_l = \frac{1}{n-1} \sum_{p=1}^{speakers} (u[i]_{p,l} - \bar{u}[i]_l)^2 \quad (3)$$

and, with t ranging over segments,

$$s2[i]_l = \sqrt{\sum_{x=0}^{t} \binom{t}{x} \bar{u}[i]_l^x (1-\bar{u}[i]_l)^{t-x} (\frac{x}{t} - \bar{u}[i]_l)^2} \quad (4)$$

We can thus estimate the discrimination error between two languages, 1 and 2, using this approximation of the Bhattacharyya distance measure:

$$\frac{1}{2} e^{-\frac{1}{4}[\frac{(a[i]_2 - a[i]_1)^2}{s[i]_1^2 + s[i]_2^2}]} \quad (5)$$

Sequences are ordered based on the estimated discrimination error. The top N sequences make up the list of vocabulary used for discriminating two chosen languages.

2.4. Phoneme Recognition Accuracy

Clustering using the above algorithm results in 78 speech units for the six language system, including Hindi, Japanese, Mandarin, Spanish, German, and English. Only those speech units which do not result in decreased performance for any of the language pairs in the six language set are allowed to cluster. Discriminating between German and English results in 59 speech units. In Table 1 it can be seen that clustering improves frame based recognition accuracy. Using a grammar to align the files is more meaningful after clustering.

Classes	Grammar	Performance (%)	Alignment (%)
95	SIX	16	15
	ENGE	16	16
78	SIX	19	21
59	ENGE	19	25

Table 1: Summary of Results from Alignment

3. Language Identification

Language identification is based on deriving a "vocabulary" of representative sequences for each language in the system as described in the previous section. Occurrence frequencies of these sequences are computed for an utterance of unknown language and is classified as the language which best matches the vocabulary. We describe how each sequence is represented, selected, and applied towards language identification.

3.1. Sequence Representation

We want to allow for inaccurate matches of sequences in an incoming utterance in order to account for unseen data, inaccurate alignment, and pronunciation/dialect variability within a language. In order to represent a sequence we define a measure of allowed inaccuracy. If we imagine the space of all sequences partitioned as shown in Fig. 1, we use the following terms:

Center The **Center** sequence of a set of associated sequences - their representative sequence.

Radius Defines the degree of allowed inaccuracy between sequence and **Center**.

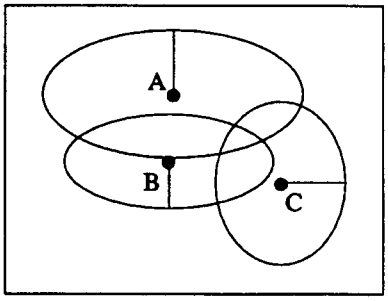

Figure 1: Space of All Sequences. A,B, and C represent the centers of the three sets. Each set is associated with a **Radius** shown by the line. Sets may overlap.

We next derive the degree of inaccuracy with which this **Center** can be matched without deteriorating language discriminability.

3.2. Associating Sequences

A "Word" is created by associating a set of sequences with each other. The aim is to associate sequences which cover the variability of a word within one language without sacrificing any discriminability across languages. For example, suppose the given list of pronunciations represents the word "und" in German and "and" in English:

"und"	/uh/n/tcl/t/	/eh/n/tcl/t/	/eh/n/dcl/d/
"and"	/ah/n/dcl/d/	/ah/m/dcl/d/	/eh/n/dcl/d/

Looking at these pronunciations, we note that the first two pronunciations belong only to one of the languages while the third pronunciation is no longer distinctive. We want to develop some measure by which the addition of the third pronunciation is restricted, while still allowing the flexibility of associating the first two with the respective language dependent words.

Distance scores between two sequences are calculated using dynamic time warping with bigram based confusions derived from the labeled files (represented by x,y) and the aligned files (represented by a,b) as follows:

$$\begin{array}{ll} \textbf{P(sub1)} & P(a|x) \\ \textbf{P(sub2)} & \lambda P(ab|xy)P(xy) + (1-\lambda)P(a|x)P(b|y) \\ \textbf{P(del)} & P(a|xy) \\ \textbf{P(ins)} & P(ab|x) \end{array} \quad (6)$$

Thus the cost β of a substitution of label **a** for label **x** during alignment is equal to P(sub1), and the probability that the sequence **a** in the aligned file actually represents **xy** is equal to P(del). Due to the lack of data P(sub2) interpolates bigram probability with unigram probability. λ was chosen to be .9. The main reason for this extra term is uniqueness for each substitution for the purpose of defining a unique **Radius** for each added sequence. Probabilities corresponding to longer lengths sequences are always derived from the above bigrams. The reason for this approach is two-fold. First, bigrams are a reasonable basic unit since the alignment is based on a bigram-grammar only. Second, we want to be able to create a score for any occurring sequence in the test set that we have not seen before. Using bigrams we can create a consistent scoring algorithm. In general for sequences of length i and j, greater than two, the distance $\beta[A_i, X_j]$ between sequence A and X is computed below:

$$\beta[A_1, X_1] = P(sub2), \beta[A_1, X_0] = P(ins), \beta[A_0, X_1] = P(del)$$

$$\beta[A_i, X_j] = max \begin{cases} \beta[A_{i-1}, X_{j-1}] & *P(sub2) \\ \beta[A_{i-1}, X_j] & *P(del) \\ \beta[A_i, X_{j-1}] & *P(ins) \end{cases}$$

The score β relates to the probability that a given sequence A is a variation of the **Center** sequence X. β will be used to weigh the occurrence frequency of A.

3.3. Feature selection

We now have an estimate of the language discrimination error due to a set of associated sequences. The size of the set is delimited by its **Radius**. The goal is to restrict the **Radius** such that the allowed variability does not reduce the ability to discriminate languages. Sequences are sorted with respect to their distance to the **Center** estimated by β. The new mean μ and variance σ is recomputed based on

the assumption that all sequences in the corresponding set are treated as one in the training files. The discrimination error is estimated at each **Radius** using the approximated Bhattacharyya distance as in Equation 5 by substituting μ for u and σ for s. The optimal **Radius** is chosen to represent the **Center**. Figure 2 depicts this process.

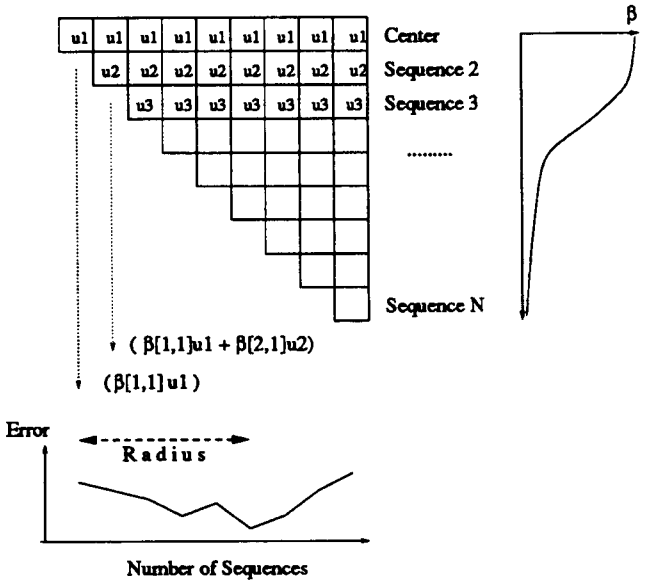

Figure 2: Grouping sequences together. ui denotes mean occurrence of sequence i. The **Radius** indicates how many sequences may be added to a set before the discrimination error increases.

3.4. Language Identification

The incoming sequences are matched with all **Centers** which are represented by a sequence and a **Radius**. If the returned score of the match is within the given **Radius**, then the corresponding word count is incremented by the corresponding weight. It is possible that a given sequence matches more than one **Center** but the weight with which they are associated differ. All resulting occurrence frequencies of these **Centers** are normalized by the length of the utterance. Since the normal assumption that was used during error-estimation for clustering may not be appropriate, we have used a non-linear neural network as classifier which is also able to take co-occurrence of input features into account.

4. Results

Table 2 compares language discrimination for sequences of length 1. It can be seen that results are best when combining inexact sequence matching with phoneme clustering. Thus, the error rate is reduced substantially by inexact matching when the phonemic units are clustered into 59 classes. When 95 classes are retained, however, inexact matching degrades performance, because there is too much variability in the resulting groupings of phonemes.

Classes	type	accuracy (%)
95	exact	87
	inexact	80
59	exact	88
	inexact	93

Table 2: Summary of Results from Language Identification

5. Conclusions

From the results it can be seen that through inexact sequence matching we can improve generalization from training to test set for our language identification system. It has been reported in the literature for similar approaches that sequences of longer length carry more information than shorter ones. We believe that the key to success lies in applying inexact sequence matches to shorter sequences by allowing intersection between sets of sequences that are clustered together.

6. Acknowledgements

Support for this research comes from NSF and the member companies of CSLU. We would also like to thank Todd Leen and Mark Zissman for helpful advice.

7. REFERENCES

1. Etienne Barnard and Ronald A. Cole. A neural-net training program based on conjugate-gradient optimization. Technical Report CSE 89-014, Oregon Graduate Institute, 1989.
2. Kay Berkling and Etienne Barnard. Theoretical error prediction for a language identification system using optimal phoneme clustering. In *Proceedings Eurospeech*, volume 1, pages 351–354, Madrid, Spain, September 1995.
3. Hynek Hermansky. Perceptual linear predictive (PLP) analysis of speech. *Journal of the Acoustic Society of America*, 4:1738–1752, April 1990.
4. Y. K. Muthusamy, Neena Jain, and Ronald A. Cole. Perceptual benchmarks for automatic language identification. In *International Conference on Speech and Signal Processing*, volume 1, pages 333–336, Adelaide, Australia, April 1994.