39. Principles of Spoken Language Recognition

C.-H. Lee

In this introductory chapter to Part G of this Handbook on spoken language recognition, we provide a brief overview of the principles of state-of-the-art language recognition approaches, and a general discriminative training framework to improve the performance and robustness of language recognition systems. It is followed by three chapters. The first of these addresses issues related to spoken language characterization in which knowledge sources can be utilized to distinguish one language from another. The second chapter deals with language identification based on phone recognition followed by language modeling using either spectral or token-based approaches. The third chapter presents vector-space characterization approaches to converting speech utterances into spoken document vectors for modeling and classification. With recent progress in speech processing, machine learning, and text categorization, we expect significant technology advances in spoken language recognition in the years to come.

This chapter is organized as follows. Section 39.2 briefly describes the principle of spoken language recognition. Sections 39.3 and 39.4 formulate the popular parallel phone recognition followed by language modeling (P-PRLM) and vector-space characterization (VSC) approaches to spoken language identification. In Sect. 39.5 we extend these formulations to spoken language verification. Finally a general discriminative training framework for non-support vector machine (non-SVM) classifiers is presented in Sect. 39.6, followed by a brief summary in Sect. 39.7.

39.1 Spoken Language

A spoken language can be identified using information from a number of sources. When human beings are constantly exposed to a language without being given any linguistic knowledge, they learn to discriminate subtle differences by perceiving speech cues in the specific languages. Nonetheless lexical knowledge is usually the main source of information for human listeners, especially when the languages to be distinguished are similar. For automatic spoken language recognition by machines, state-of-the-art systems often consider unknown utterances as a concatenation of signal patterns, and probabilistic modeling and classification techniques are then adopted to characterize these patterns and their corresponding language identities. Although lexical models are not explicitly utilized, models of phones in a subset of languages are often used to decode speech for further processing. Most human listeners are usually constrained by their surroundings from effectively learning a large number of languages. On the other hand machines are equipped to take in virtually an unlimited amount of speech and text. It is interesting to note that advanced machines therefore have the potential to surpass human performance in spoken language recognition.

With a coordinated effort between government funding agencies and technical communities for data collection and benchmark performance evaluation, the current availability of large collections of speech examples from a selected set of languages, easy access to fast and affordable computing equipment, and recent

advances in speech modeling and machine learning, many new algorithms have recently been explored in this area. We are now witnessing rapid progress in the area of spoken language modeling and recognition technologies.

In this introductory chapter to Part G of this Handbook on spoken language recognition, we provide a brief discussion of the principles of state-of-the-art language recognition approaches. It is then followed by three more chapters. The first addresses issues of spoken language characterization, in which knowledge sources that can be utilized to distinguish one language from another are presented. Contrasts between human and machine language recognition are made to highlight that language cues perceived by human listeners can be incorporated into current technologies to improve machine capabilities. It is also noted that there are a few orders of magnitude more languages and dialects in existence than current systems are equipped to handle. Many of these languages are rare and extensive data collection to construct recognizers for these languages can be prohibitive. Some of today's prevailing technologies may have to be modified to alleviate these limitations. New paradigms may also have to be established to examine the language recognition problem from a completely different perspective.

The two other chapters in this part of the Handbook describe prevailing trends in current language recognition system design. The first attempts to divide language recognition approaches into two broad categories. The first method is spectral-based methods, in which each spoken utterance or segment is represented by a sequence of feature vectors that are often short-time spectral representation of speech frames. This is similar to what is typically done in the front-end feature-extraction module of an automatic speech recognition (ASR) [39.1] system. These spectral vectors are assumed to be generated from different source languages, and therefore have different characteristics. A collection of models, one for each language to be considered, can then be built with labeled spectral vector examples collected from all the languages. These models can be probabilistic, such as a Gaussian mixture model (GMM) [39.2], or vector based, such as a support vector machine (SVM) [39.3]. It can be seen that this spectral-based framework is purely acoustic, while no linguistic information, such as phones or words, is used. The second method is called the token-based approach, in which an intermediate set of speech units, also referred to as tokens, is used to represent speech. An utterance is first decoded and segmented into a sequence of such tokens. These token streams are then used to extract features, designing classifiers, and performing spoken language classification.

The last chapter in this part of the Handbook represents a new vector-space characterization (VSC) [39.4] approach to language recognition in which an utterance is considered as a spoken document. A term-document matrix representation of all utterances in a training set can then be established for indexing and retrieval purposes. The terms here can refer to acoustic words (AWs), which are sequences of acoustic letters (alphabets), or fundamental sound units, and the vector elements are often co-occurrence statistics describing the frequencies of AWs in each spoken utterance. Language recognition with an unknown spoken query can then be treated as a document-retrieval problem similar to that which is commonly faced in the information retrieval (IR) [39.5] community. It can also be cast as a text categorization (TC) [39.6, 7] problem in which all training documents are used to build text categorizers, or topic classifiers, one for each language to be considered. In so doing language recognition can be accomplished by comparing the unknown spoken query vector with each of the language classifiers to make a recognition decision. This vector-based modeling framework allows one to represent speech utterances with a very high-dimension document vector, sometime in the tens of thousands, so that many of the feature extraction and classifier learning algorithms currently available in the IR and TC literatures can be easily adopted for spoken language recognition. New language cues can also be exploited and incorporated into the spoken document vectors, along the same lines as a recently proposed automatic speech attribute transcription (ASAT) [39.8] paradigm for ASR, to improve system performance.

39.2 Language Recognition Principles

Automatic spoken language identification (LID) is a process of determining the language spoken in a speech sample. LID technology is needed in many applications such as multilingual conversational systems [39.9], spoken language translation [39.10], multilingual speech recognition [39.11], and spoken document re-

trieval [39.12]. It is also a topic of great importance in the areas of intelligence and security, where the language identities of recorded messages and archived materials need to be established before any information can be extracted from them. In the past few decades, researchers have explored many speech and language knowledge sources, including articulatory parameters [39.13], acoustic features [39.14], prosody [39.15,16], phonotactic [39.17,18], and lexical knowledge [39.19]. Recently, investigators have reported promising results using *shifted-delta-cepstral* acoustic features [39.20]. From the perspective of human language recognition, humans constantly exposed to a language without being given any linguistic knowledge learn to determine the identity of the language by detecting some distinct speech cues in the specific language. For example, an English-speaking listener can often appreciate the tonal nature of Mandarin Chinese. It is also noted in human perceptual experiments that listeners with a multilingual background often perform better than monolingual listeners in identifying unfamiliar languages [39.21]. These reasons motivate us to explore useful speech attributes in languages, along the same lines as a recently proposed ASAT paradigm for automatic speech recognition (ASR) [39.8]. Other useful speech and language features, such as prosodic and syllabic content, can also be incorporated into this feature vector.

In a typical pattern recognition setting, one must evaluate the a posterior probability, $P(\lambda_l|X)$, of the l-th language to be considered with a model λ_l given an unknown utterance X. To make a decision, the language that yields the maximum $P(\lambda_l|X)$ is usually identified as the target language. Many algorithms developed in automatic speech and speaker recognition [39.22, 23] can be adopted and extended to language recognition. Recent advances in acoustic and language modeling [39.24] have also contributed to progress in language recognition.

Taking advantage of recent advances in continuous speech recognition with hidden Markov models HMMs [39.25], probabilistic approaches [39.26–29] have been developed by exploiting techniques in acoustic phone modeling and n-gram language modeling. These characterize a spoken language using probability distributions of spectral features in the form of linguistically defined symbols, such as phones and syllable-like units [39.17, 30], where phone models are used to decode speech utterances into sequences of such fundamental symbols. An interpolated phone-based n-gram language model is then constructed for each language, and to derive phonotactic scores. Such an example is parallel phone recognition followed by language modeling (P-PRLM) [39.18], which uses multiple single-language phone recognizers as a front end and language-dependent language models as the back end. The phonotactic approach has been shown to provide superior performance on NIST language recognition evaluation (LRE) tasks [39.27]. It is generally agreed that the fusion of multiple phonotactic features improves performance. For example, the P-PRLM approach derives multiple sets of phonotactic features. Others have found that multiresolution phonotactic analysis, such as phone unigram, bigram, and trigram approaches, complement each other [39.17, 31, 32].

A key question here is whether a phone definition is needed to represent fundamental speech units. If we can *tokenize* speech with a manageable set of *spoken letters* and develop models to decode spoken utterances into sequences of these acoustic units, it is clear that the statistics of these spoken letters and their co-occurrences can be used to discriminate one spoken language from another. Although common sounds are shared considerably across spoken languages, the statistics of these sounds, such as phone n-grams, can differ considerably from one language to another. An interesting generalization through acoustic units is to represent any spoken utterance (also referred to as a spoken document when presented in the form of *spoken letters*) with a high-dimensional feature vector, where each element carries sound co-occurrence statistics. This is similar to a latent semantic indexing (LSI) vector representation [39.33] of text documents, which is commonly used in information retrieval (IR) systems [39.5]. Such statistics are considered to be salient features for indexing and retrieving documents. We call this paradigm vector-space characterization of speech [39.4].

To develop a universal set of fundamental speech units to cover the acoustic characterization of all spoken languages we relax the notion of language-specific phonetic definitions. To relate this universal unit set to language discrimination, it is well known that the entropy of English can be effectively reduced when high-order statistics of letters are computed [39.34]. For example, of the 26 English letters plus the space character, a few of them, such as n, s, and t, occur more often in text than others, such as x and q. By incorporating this first-order statistics, or unigram, the entropy of English text can easily be reduced from 4.76 to 4.03 bits. When letter bigrams and trigrams are added, the entropy is further reduced. This set of statistics can be used to discriminate among languages already decoded, even

if no extra dictionary information is explicitly utilized. The same notion can be extended to spoken language identification. This is sometimes referred to as a bag-of-sound representation [39.26], in analogous to the bag-of-words representation of text documents used in IR. Just as in P-PRLM multiple bags of sound can also be used [39.35].

A model corresponding to the acoustic letters approach mentioned above is called an acoustic segment model (ASMs) [39.36] and can be used to decode spoken utterances into strings of such units. HMMs [39.25] are often used to model the collection of ASMs, which can be established in a bottom-up unsupervised manner. ASMs have been used to construct an acoustic lexicon for isolated word recognition with high accuracy [39.36]. Acoustic words (AWs) can now be formed by grouping adjacent acoustic letters, and serve as a basis to build feature vectors for spoken documents and build language classifiers. Therefore, automatic language classification can be formulated as a text categorization (TC) [39.6] problem based on LSI-derived feature vectors and discriminative classifier design [39.37]. Many existing feature representation and machine learning techniques widely available in the IR and TC literature can now be adopted.

39.3 Phone Recognition Followed by Language Modeling (PRLM)

Consider a speech utterance X to be represented by a sequence vectors of length τ, $O = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_i, \ldots, \boldsymbol{o}_\tau\}$, in which $\boldsymbol{o}_i$ is a feature vector extracted from X at time i. We can express the a posteriori probability of language l using Bayes theorem, as follows:

$$P(l|O) = P(O|l)\frac{P(l)}{P(O)}\,, \tag{39.1}$$

where $P(l)$ and $P(O)$ are prior probabilities of observing language l and vector sequence O, respectively. Without loss of generality we can assume $P(l)$ to be equal for all languages. The language-independent term, $P(O)$, usually does not affect our decision rules. They will be dropped hereafter in this chapter. Now we can apply the *maximum a posteriori* (MAP) decision rule for LID as:

$$\hat{l} = \arg\max_l P(l|O) = \arg\max_l P(O|\lambda_l)\,, \tag{39.2}$$

where λ_l is an model for the l-th language. Here the model λ_l can be any reasonable characterization of the feature vectors. A straightforward choice is to consider all such vectors to be generated from a language-specific density, and a GMM can then be used to model the source. This is exactly the same formulation as in GMM-based text-independent speaker identification [39.38]. This approach is purely frame based, and no segmental information is used. SVMs [39.39] have been used to design both high-performance GMM-based speaker and language recognition systems [39.2, 3, 40].

If we fold in some fundamental speech units, and associate a given utterance with a sequence of such units then more-detailed models can be incorporated. For example, if we have a set of phone HMMs λ_l^{AM} and a set of phone language models λ_l^{LM} trained with speech data with implied phone sequence labels from language l, then we have

$$\hat{l} = \arg\max_l \sum_{\forall q} P\left(O|q, \lambda_l^{\mathrm{AM}}\right) P\left(q|\lambda_l^{\mathrm{LM}}\right)\,, \tag{39.3}$$

where q is a candidate phone sequence. In many cases the sum in (39.3) is approximated by finding the most dominant phone sequence $\hat{q}_l$ with the l-th phone model λ_l^{AM} using Viterbi decoding,

$$\hat{q}_l = \arg\max_{\forall q} P\left(O|q, \lambda_l^{\mathrm{AM}}\right)\,, \tag{39.4}$$

and solving for the following:

$$\hat{l} \approx \arg\max_l \left[P\left(O|\hat{q}_l, \lambda_l^{\mathrm{AM}}\right) P\left(\hat{q}_l|\lambda_l^{\mathrm{LM}}\right)\right]. \tag{39.5}$$

This is known as the phone recognition followed by language modeling (PRLM) approach to LID. Since obtaining high-accuracy phone and language models for each individual language under consideration is not always possible, one can consider training a single set of phone models to cover all languages and use it to decode all spoken utterances. Although the decoding performance for sounds not properly modeled by this model set is usually not good, we can still get reasonable performance by dropping the dependency on l for the acoustic score term and the phone sequence. Now (39.5) is solved by:

$$\hat{l} \approx \arg\max_l P\left(\hat{q}|\lambda_l^{\mathrm{LM}}\right)\,. \tag{39.6}$$

It is interesting to note that phone models are used only for decoding, and not for scoring when deter-

mining the identified language. We can also expand the phone model collection to include multiple sets of models, each trained by speech examples from a subset of languages. Now the set of F phone models, $\{\lambda_1^{\mathrm{AM}}, \ldots, \lambda_f^{\mathrm{AM}}, \ldots, \lambda_F^{\mathrm{AM}}\}$, are all used to decode a given spoken utterance, resulting in a set of F phone sequences, $Q = \{q_1, \ldots, q_f, \ldots, q_F\}$. Furthermore for the l-th language, we can train F sets of phone language models, $\{\lambda_{l,1}^{\mathrm{LM}}, \ldots, \lambda_{l,f}^{\mathrm{LM}}, \ldots, \lambda_{l,F}^{\mathrm{LM}}\}$, to compute F language-specific scores, $\{P(q_f|\lambda_{l,f}^{\mathrm{LM}}), f = 1, \ldots, F\}$. These scores can then be combined to make language classification decisions. This the basic idea behind parallel PRLM (P-PRLM) [39.18], which is by far the most successful approach to LID.

39.4 Vector-Space Characterization (VSC)

Vector-space modeling has become a standard tool in IR systems since its introduction several decades ago [39.5]. It uses a vector to represent a text document or a query. It has also been applied to text categorization [39.41] in which training vectors are used to design a collection of topic classifiers. A vector-based TC approach to LID has been proposed recently [39.37].

Suppose that an utterance X, represented by a sequence of speech feature vectors O, is decoded or tokenized, into a *spoken document*, $d(X)$, consisting of a series of I acoustic units, $d(X) = \{t_1 \ldots, t_i \ldots, t_I\}$, where each unit is drawn from a universal inventory, $U = \{u_1, \ldots, u_j, \ldots u_J\}$, of J acoustic letters shared by all the spoken languages to be considered, such that $t_i \in U$. We are then able to establish a collection of acoustic words by grouping units occurring consecutively to obtain a vocabulary of M distinct words, $W = \{w_1, \ldots, w_m, \ldots w_M\}$, such that each w_m can be a single-letter word like (u_j), a double-letter word like $(u_j u_k)$, a triple-letter word like $(u_j u_k u_l)$, and so on. Usually the vocabulary size, M, is equal to the total number of n-gram patterns needed to form words, e.g., $M = J + J \times J + J \times J \times J$ if acoustic words up to three tokens in length are considered valid. Next we can use some form of function $f(w_m)$, such as LSI [39.33], to evaluate the significance of having the word w_m in the document, $d(X)$. We are now ready to establish an M-dimension feature vector, $\boldsymbol{v} = [f(w_1), \ldots, f(w_m), \ldots f(w_M)]^{\mathrm{T}}$ in which $\boldsymbol{x}^{\mathrm{T}}$ denotes the transpose of the vector $\boldsymbol{x}$ for each spoken document.

It is clear that we need a number of fundamental units sufficient to cover the acoustic variation in the sound space. However a large J will result in a feature vector with a very high dimension if we would like to cover as many unit combinations when forming acoustic words. For example, with a moderate value of $J = 256$, we have $M = 65\,792$ even when we only consider words less than or equal to two letters. This is already a very large dimensionality not commonly utilized in speech and language processing algorithms. Finally, a vector-based classifier evaluates a goodness of fit, or score function $S_l(\boldsymbol{v}) = S(\boldsymbol{v}|\lambda_l)$ between a given vector $\boldsymbol{v}$ and a model of the l-th spoken language λ_l to make a decision. Any vector-based classifier can be used to design spoken language identification and verification systems. A goodness of fit, such as an inner product using a linear discriminant function (LDF) [39.42], can be used to evaluate vector-based scores:

$$S(\boldsymbol{v}|\lambda_l) \propto \gamma_l^{\mathrm{T}} \cdot \boldsymbol{v}\,, \tag{39.7}$$

where γ_l is a language-dependent weight vector of equal dimensionality to $\boldsymbol{v}$, with each attribute representing the contribution of its individual n-gram probability to the overall language score. The spoken document vector, $\boldsymbol{v}$ in (39.7), is high dimensional in nature when patterns up to triplets are included. When multiple inventories of models, like in P-PRLM, are used to produce a collection of F document vectors, $\{\boldsymbol{v}_f, f = 1, \ldots, F\}$, a large composite document vector, or supervector, $\boldsymbol{v} = [\boldsymbol{v}_1^{\mathrm{T}}, \ldots, \boldsymbol{v}_f^{\mathrm{T}}, \ldots, \boldsymbol{v}_F^{\mathrm{T}}]^{\mathrm{T}}$ can be established by stacking these F vectors. It has been shown that such supervectors have more discrimination power than single document vectors for language recognition [39.35].

Term weighting [39.43] is widely used to render the value of the attributes in a document vector by taking into account the frequency of occurrence of each attribute. It is interesting to note that attribute patterns that occur often in a few documents but not as often in others give high indexing power to these documents. On the other hand, patterns that occur very often in all documents possess little indexing power. This desirable property leads to a number of term weighting schemes, such as *tf-idf* (term-frequency-inverse document frequency) [39.44] and LSI [39.33], which are common for information retrieval [39.5], natural language call routing [39.45], and text categorization [39.41]. VSC is motivated by the same ideas, which aim to derive weights that dis-

criminate between languages using high-dimensional salient-feature vectors.

After representing a spoken document as a vector of statistics of AWs, LID becomes a vector-based classification problem. Many classifier designs in the machine learning literature for high-dimensional vector classification are readily available. For example conventional techniques, such as SVM [39.39], classification and regression tree (CART) [39.46] and artificial neural networks (ANN) [39.47], can be used to train vector-based classifiers, such that LID can be solved in the form:

$$\hat{l} = \arg\max_{l} S(\boldsymbol{v}|\lambda_l) \,. \tag{39.8}$$

We can also consider an indirect vectorization mechanism based on the scores computed for all the classes of interest, obtained from an ensemble of classifiers. Since these scores characterize the distribution of the outputs of a variety of classifiers, they do provider significant discriminatory power among classes, and can be grouped together to form *score supervectors* describing the overall behavior of a score distribution over different classifiers for all competing classes. One example of such a supervector is to concatenate HMM state scores from all the competing models to form an overall score vector. This approach has been shown to have good discriminatory power in isolated letter recognition [39.48, 49]. Since these score supervectors are usually obtained from a finite set of ensemble classifiers, their dimensionality is often much lower than that of the spoken document vectors discussed above. These supervectors are more amendable to treatment using conventional probabilistic modeling frameworks.

39.5 Spoken Language Verification

So far we have only discussed spoken language identification. Another recognition problem of interest is spoken language verification, which can be cast as a statistical hypothesis testing problem, i. e., testing a null hypothesis H_0 that an utterance O is from a claimed language against an alternative hypothesis H_1 that O is not from that language. Many issues we will discuss in this section have similarities to those commonly addressed in speaker and utterance verification [39.50]. According to the Neyman–Pearson lemma [39.51], an optimal test can be formulated as: given a test speech O, accept H_0 if

$$\frac{P(O|H_0)}{P(O|H_1)} > r_{\text{th}} \,, \tag{39.9}$$

where $P(O|H_0)$ and $P(O|H_1)$ are the probability distributions for the null and alternative hypotheses, respectively. The constant r_{th} is a verification threshold. The test in (39.9) is known as a probability ratio test. The ratio $P(O|H_0)/P(O|H_1)$ is called a *probability ratio statistic*. The threshold r_{th} is referred to as the *critical value*; the region $A = \{O : [P(O|H_0)/P(O|H_1)] > r_{\text{th}}\}$ is called the acceptance region, and the region $\overline{A}$, containing points not in A, is called the critical region of the test.

There are two verification decisions to be made, namely rejection and acceptance of the null hypothesis. Correspondingly there are two types of errors, namely false rejection (type I) and false acceptance (type II). The power of a test is defined as the probability of correct rejection, i. e., rejecting H_0 when it should be rejected, which is one minus the maximum of the probability of a type II error. The level of significance of a test is defined as the maximum of the probability of type I errors. The level of significance is computed based on the distribution of the likelihood test statistic and the choice of the threshold.

To test some simple hypotheses under a few regularity conditions, a generalization of the Neyman–Pearson lemma shows that a likelihood-ratio test (LRT) is the *most powerful test* (smallest type II error test) for a given level of significance (type I error) if $P(O|H_0)$ and $P(O|H_1)$ are known exactly. For testing more-complex composite hypotheses, an optimal test is usually hard to come by. Furthermore in most practical verification applications we do not have access to complete knowledge of H_0 and H_1, and therefore optimal tests cannot be designed. In these cases we can attempt to evaluate $P(O|H_0)$ and $P(O|H_1)$ based on some assumed forms for their distributions, such as those discussed in Sect. 39.2, in which case a probability ratio test can still be used.

Even in cases where a probabilistic characterization is not easy to determine, we can still compute scores such as $S(O|H_0)$ and $S(O|H_1)$, so that the VSC models presented in Sect. 39.3 are equally applicable to the design of spoken language verification systems. Even with probabilistic modeling we can also evaluate scores of the form $S(O|H_i) = \log P(O|H_i)$ so we can use any score to compute a generalized log likelihood ratio (GLLR) test

statistic, which can be considered as a distance measure:

$$d(O, \Lambda) = -S(O|\lambda_{H_0}) + S(O|\lambda_{H_1}) , \qquad (39.10)$$

where $\Lambda = (\lambda_{H_0}, \lambda_{H_1})$ is the collection of all models. We then accept H_0 if $d(O, \Lambda) < \tau_{\text{th}}$. Here τ_{th} is another form of the verification threshold. This formulation can be applied directly to two-class verification problems. However the alternative hypothesis H_1 is often difficult to model. For example if H_0 is defined for a claimed language l, H_1 is often a composite hypothesis representing all other competing languages other than l. There are often more negative and diverse samples to train the *impostor* model λ_{H_1} than positive and consistent samples to train the target model λ_{H_0}.

One way to alleviate this difficulty is to train one model for each of the languages of interest to obtain a collection of models $\Lambda = (\lambda_1, \ldots, \lambda_L)$, one for each language. Now, to verify whether an unknown utterance O is generated from a claimed language l, the target score can be computed as $S_l(O|H_0) = S(O|\lambda_l)$. Furthermore we can assume that $S_l(O|H_1)$ is a function of all the competing language scores other than $S(O|\lambda_l)$. One approach is to assume that $S_l(O|H_1)$ is evaluated as an antidiscriminant function commonly used in discriminative training (DT) in ASR [39.52,53] to compute a geometric average of all competing scores as

$$S_l(O|H_1) = \log \left\{ \frac{1}{L-1} \sum_{i \neq l} \exp[\eta S(O|\lambda_i)] \right\}^{\frac{1}{\eta}} , \qquad (39.11)$$

where η is a positive constant. It is noted that the right-hand side of (39.11) is the L_η norm in real analysis, and that it converges to $\max_i S(O|\lambda_i)$ as $\eta \to \infty$. This score has been used in multicategory (MC) text categorization [39.37], and it was shown that MC TC outperforms binary TC, especially in cases when there are very few positive examples, sometimes only one sample, to train a topic classifier. The same idea was also applied to the design of language verification systems in the 2005 NIST language recognition evaluation [39.42]. Part of the successes in [39.37] and [39.42] can be attributed to discriminative classifier learning, which will be discussed in Sect. 39.6.

By now it is clear that the modeling and classifier design techniques discussed in Sects. 39.3 and 39.4 for language identification can be extended to language verification as well. One remaining key issue is the selection of verification thresholds, which is critical in designing high-performance spoken language verification systems for real-world applications. They are often determined empirically according to the specific application requirements. Since the distance measure in (39.10) also characterizes a separation between the target and competing models [39.54], we can plot histograms of $d_l(O, \Lambda)$ over positive and negative training samples, respectively. The size of the overlap region of the two curves is often a good indicator for predicting type I and type II errors. The verification thresholds can then be determined by selecting a value in the overlap region to balance the false-rejection and false-acceptance errors. One good illustration was demonstrated in the recent 2005 NIST LRE [39.42].

39.6 Discriminative Classifier Design

We have now briefly addressed most of the research issues in designing language recognition systems. More detail can be found in the last two chapters of this Part G. Next we will discuss an important methodology for discriminative training that has created a lot of enthusiasm in the fields of speech recognition [39.52,53], utterance verification [39.55], and text categorization [39.41], but not yet been fully utilized in the language recognition community. So far SVM is still the dominating discriminative classifier design approach to LID. However the techniques used in SVM training are not easily extended to other popular classifiers, such as GMM, HMM, ANN, LDF, and CART. In the following we describe a general framework to formulate a broad family of DT algorithms for any classifier based on any performance metric.

Consider a set of training patterns $\boldsymbol{X} = \{X_i, 1 \leq i \leq N\}$, with N being the total number of training tokens, and O_i, an observation vector sequence associated with X_i. Each token belongs to one of L classes, C_l, $1 \leq l \leq L$. The goal of pattern classification is to use the labeled set X to design a decision rule based on a set of parameters $\Lambda = \{\lambda_l, l = 1, ..., L\}$ such that the classification error is minimized. The optimal classification approach is the Bayes decision rule if the a priori probabilities $P(C_l)$ and the class-conditional probability $P(O|C_l)$ are known. Unfortunately, in pattern recognition applications we rarely have complete knowledge of the probabilistic structure of the problem. We only have some vague knowledge about the distribution and are given a finite number of training samples to design the

classifiers. A conventional approach to this problem is to assume a parametric form for the conditional densities and obtain estimates of the parameters from the given finite set of labeled training patterns. However, any assumed parametric distributions will create some mismatch between the true and estimated conditional distributions. The limited availability of training data is another problem for effective characterization of the conditional densities.

An alternative approach that alleviates these problems is to use a set of class discriminant functions (similar to the score functions discussed above) $\{g_l(O, \Lambda) : l = 1, \ldots, L\}$ to perform classification and hypothesis testing, and to use discriminative training to estimate the parameters Λ. The form of the probability distributions is not needed in this case. Each $g_l(O; \Lambda)$ evaluates the similarity between a given test utterance O and the class C_l. To combine current speech pattern classification techniques and the discriminative training approach we can use the conditional class distributions as the class discriminant functions and obtain the classifier parameters accordingly based on minimum error classification (MCE) [39.52,53] and minimum verification error (MVE) [39.55] training. Recently maximal figure-of-merit (MFoM) [39.41] learning for vector-based classifiers has also been shown to give superior and robust performance compared with SVM-trained classifiers in text categorization.

Three additional functions are required to formulate discriminative training, namely:

1. the class antidiscriminant function, $G_l(O; \Lambda)$ [similar to the quantity in (39.11)]
2. the class misclassification measure, $d_l(O; \Lambda)$, which is usually defined as the difference between $G_l(O; \Lambda)$ and $g_l(O; \Lambda)$ [similar to the measure in (39.10)]
3. the class loss function, $l_l(O; \Lambda) = l[d_l(O; \Lambda)]$, which measures the loss or cost incurred for a given value of $d_k(O; \Lambda)$

The loss is often a smooth 0–1 function such as a sigmoid to approximate error counts based on its distance from the decision boundary, $d_l(O; \Lambda) = \beta_l$:

$$l_l(O; \Lambda) = \frac{1}{1 + \exp[-\alpha_l(d_l - \beta_l)]} , \tag{39.12}$$

where α_l and β_l are parameters for the sigmoid function characterizing the slope near, and the location of, the decision boundary, respectively.

Given the four sets of functions the optimization objective functions can be defined differently, depending on the performance metrics to be optimized in each application. For language identification, we are interested in minimizing the overall approximate empirical classification error rate for all training data, $\boldsymbol{O} = \{O_i, 1 \le i \le N\}$:

$$L(\boldsymbol{O}, \Lambda) = \frac{1}{N} \sum_{i=1}^{N} \sum_{l=1}^{L} l(d_l) \mathbf{1}(O_i \in C_l) , \tag{39.13}$$

where the function $\mathbf{1}(S)$ is the indicator function for a logical variable S. For language verification we can approximate the empirical false-rejection (type I) and false-acceptance (type II) error rates for the l-th language as

$$L_{l1}(\boldsymbol{O}, \Lambda) = \frac{1}{N_l} \sum_{i=1}^{N_l} l(d_l) \mathbf{1}(O_i \in C_l) \tag{39.14}$$

and

$$L_{l2}(\boldsymbol{O}, \Lambda) = \frac{1}{\overline{N}_l} \sum_{i=1}^{\overline{N}_l} l(d_l) \mathbf{1}(O_i \in \overline{C}_l) , \tag{39.15}$$

where N_l and $\overline{N}_l$ are the numbers of tokens in and not in class C_l in the training set, respectively, and $\overline{C}_l$ denotes the set of all tokens not belonging to the class C_l. According to the application requirements we can specify ω_{l1} and ω_{l2}, for the l-th class, as the costs of making type I and II errors. Now the overall empirical average cost can be defined as:

$$L(\boldsymbol{O}, \Lambda) = \frac{1}{L} \sum_{l=1}^{L} \omega_{l1} L_{ll}(\boldsymbol{O}, \Lambda) + \omega_{l2} L_{l2}(\boldsymbol{O}, \Lambda) . \tag{39.16}$$

In IR and TC applications precision, *recall*, and $F1$ measures are commonly adopted as performance metrics. They are defined for the l-th class as:

$$\mathrm{Pr}_l(\boldsymbol{O}, \Lambda) = \frac{(1 - L_{l1}) \times N_l}{(1 - L_{l1}) \times N_l + L_{l2} \times \overline{N}_l} , \tag{39.17}$$

$$\mathrm{Re}_l(\boldsymbol{O}, \Lambda) = 1 - L_{l1} , \tag{39.18}$$

and

$$\mathrm{F1}_l(\boldsymbol{O}, \Lambda) = \frac{2 \times \mathrm{Pr}_l \times \mathrm{Re}_l}{\mathrm{Pr}_l + \mathrm{Re}_l} . \tag{39.19}$$

Now we can define the overall loss objective as the negative of the average F1 value over all L classes:

$$L(\boldsymbol{O}, \Lambda) = -\frac{1}{L} \sum_{l=1}^{L} \mathrm{F1}_l(\boldsymbol{O}, \Lambda) . \tag{39.20}$$

The three empirical loss functions in (39.13), (39.16), and (39.20) are the objectives to be minimized in MCE, MVE, and MFoM discriminative learning, respectively. Generalized probabilistic descent (GPD) algorithms [39.53] are often used to solve these optimization problems such that Λ is adjusted from one iteration to the next according to

$$\Lambda_{q+1} = \Lambda_q + \Delta\Lambda_q \tag{39.21}$$

with Λ_q being the parameter set at the q-th iteration. The correction term $\Delta\Lambda_q$, in a batch mode, is:

$$\Delta\Lambda_q = -\epsilon_q V_q \nabla L(\Lambda_q)\,, \tag{39.22}$$

where V_q is a positive-definite learning matrix and ϵ_q is a small positive real number representing the learning step size. We usually set V_q to be a diagonal matrix and define $\epsilon_q = (1 - q/Q_c)$, with Q_c denoting a prescribed maximum number of iterations to meet the convergence conditions, as required by the stochastic gradient search algorithm. GPD algorithms usually converge slowly to local minima. To speed up their convergence, QuickProp algorithms can also be applied to adjust the learning rate dynamically [39.56]. Globally optimal algorithms are also being actively pursued by the machine learning community.

These mentioned discriminative training algorithms are flexible enough to deal with any performance metric for any given classifier as long as these metrics can be expressed as functions of the classifier parameters. They can also be considered as a decision feedback mechanism in that every training token will go through a classification process first and its contributions to parameter adjustment depend on how well the decision is made. Therefore they usually work very well on the training data. On the other hand, SVM-based learning algorithms are designed to provide a *margin* to serve as a tolerance region around the decision boundaries implied by the classifiers. In so doing they usually provide better generalization capabilities than other learning algorithms. Since the test risk is often expressed as a function of the empirical risk and a regularization penalty term as a function of the VC dimension [39.39], we can design a family of margin-based discriminative training algorithm to enhance both the accuracy and robustness of the classifiers. This is a promising new direction, and some attempts in this direction have recently been proposed for estimating HMM parameters (e.g., [39.57]).

39.7 Summary

In this chapter, we have provided an overview of the three chapters to be presented later in this Part G. We have also highlighted two currently popular approaches to spoken language identification, namely phone recognition followed by language modeling and vector-space characterization. We demonstrated that the techniques developed for LID can be directly extended to spoken language verification if we can properly evaluate scores for the competing null and alternative hypotheses. Finally, we briefly presented a general framework for discriminative classifier design of non-SVM classifiers. The field of spoken language classification has witnessed rapid technological progress in recent years. We expect this trend to continue as novel paradigms are explored and high-performance systems are developed.

References

39.1 L.R. Rabiner, B.-H. Juang: *Fundamentals of Speech Recognition* (Prentice Hall, Englewood Cliffs 1993)

39.2 W.M. Campbell, D.E. Sturim, D.A. Reynolds: Support vector machines using GMM supervectors for speaker recognition, IEEE Signal Process. Lett. **13**(5), 308–311 (2006)

39.3 W.M. Campbell, J.P. Campbell, D.A. Reynolds, E. Singer, P.A. Torres-Carrasquillo: Support vector machines for speaker and language recognition, Comput. Speech Lang. **20**(2-3), 210–229 (2005)

39.4 H. Li, B. Ma, C.-H. Lee: A vector space modeling approach to spoken language identification, IEEE Trans. Audio Speech Lang. Process. **15**(1), 271–284 (2007)

39.5 G. Salton: *The SMART Retrieval System* (Prentice-Hall, Englewood Cliffs 1971)

39.6 F. Sebastiani: Machine learning in automated text categorization, ACM Comput. Surv. **34**(1), 1–47 (2002)

39.7 T. Joachims: *Learning to Classify Text Using Support Vector Machines* (Kluwer Academic, Dordrecht 2002)

39.8 C.-H. Lee: From knowledge-ignorant to knowledge-rich modeling: a new speech research paradigm for next generation automatic speech recognition, Proc. ICSLP (2004) pp. 109–112

39.9 V.W. Zue, J.R. Glass: Conversational interfaces: advances and challenges, Proc. IEEE **88**(8), 1166–1180 (2000)

39.10 A. Waibel, P. Geutner, L.M. Tomokiyo, T. Schultz, M. Woszczyna: Multilinguality in speech and spoken language systems, Proc. IEEE **88**(8), 1181–1190 (2000)

39.11 B. Ma, C. Guan, H. Li, C.-H. Lee: Multilingual speech recognition with language identification, Proc. ICSLP (2002) pp. 505–508

39.12 P. Dai, U. Iurgel, G. Rigoll: A novel feature combination approach for spoken document classification with support vector machines, Proc. Multimedia Information Retrieval Workshop (2003) pp. 1–5

39.13 K. Kirchhoff, S. Parandekar, J. Bilmes: Mixed memory Markov models for automatic language identification, Proc. ICASSP (2002) pp. 761–764

39.14 M. Sugiyama: Automatic language recognition using acoustic features, Proc. ICASSP (1991) pp. 813–816

39.15 A.G. Adami, H. Hermansky: Segmentation of speech for speaker and language recognition, Proc. Eurospeech (2003) pp. 841–844

39.16 M. Adda-Decker, F. Antoine, P.B. de Mareuil, I. Vasilescu, L. Lamel, J. Vaissiere, E. Geoffrois, J.-S. Liénard: Phonetic knowledge, phonotactics and perceptual validation for automatic language identification, Proc. ICPhS (2003) pp. 747–750

39.17 T.J. Hazen: *Automatic Language Identification Using a Segment-Based Approach*, M.Sc. Thesis (MIT, New York 1993)

39.18 M.A. Zissman: Comparison of four approaches to automatic language identification of telephone speech, IEEE Trans. Speech Audio Process. **4**(1), 31–44 (1996)

39.19 D. Matrouf, M. Adda-Decker, L.F. Lamel, J.-L. Gauvain: Language identification incorporating lexical information, Proc. ICSLP (1998) pp. 181–184

39.20 P.A. Torres-Carassquilo, E. Singer, M.A. Kohler, R.J. Greene, D.A. Reynolds, J.R. Deller Jr.: Approaches to language identification using Gaussian mixture models and shifted delta cepstral features, Proc. ICSLP (2002) pp. 89–92

39.21 Y.K. Muthusamy, N. Jain, R.A. Cole: Perceptual benchmarks for automatic language identification, Proc. ICASSP (1994) pp. 333–336

39.22 C.-H. Lee, F.K. Soong, K.K. Paliwal (Eds.): *Automatic Speech and Speaker Recognition: Advanced Topics* (Kluwer Academic, Dordrecht 1996)

39.23 C.-H. Lee, Q. Huo: On adaptive decision rules and decision parameter adaptation for automatic speech recognition, Proc. IEEE **88**(8), 1241–1269 (2000)

39.24 J.L. Gauvain, L. Lamel: Large-vocabulary continuous speech recognition: advances and applications, Proc. IEEE **88**(8), 1181–1200 (2000)

39.25 L.R. Rabiner: A tutorial on hidden Markov models and selected applications in speech recognition, Proc. IEEE **77**(2), 257–286 (1989)

39.26 H. Li, B. Ma: A phonotactic language model for spoken language identification, Proc. ACL (2005) pp. 515–522

39.27 E. Singer, P.A. Torres-Carrasquillo, T.P. Gleason, W.M. Campbell, D.A. Reynolds: Acoustic, phonetic and discriminative approaches to automatic language recognition, Proc. Eurospeech (2003) pp. 1345–1348

39.28 Y. Yan, E. Barnard: An approach to automatic language identification based on language dependent phone recognition, Proc. ICASSP (1995) pp. 3511–3514

39.29 K.M. Berkling, E. Barnard: Language identification of six languages based on a common set of broad phonemes, Proc. ICSLP (1994) pp. 1891–1894

39.30 T. Nagarajan, H.A. Murthy: Language identification using parallel syllable-like unit recognition, Proc. ICASSP (2004) pp. 401–404

39.31 K.M. Berkling, E. Barnard: Analysis of phoneme-based features for language identification, Proc. ICASSP (1994) pp. 289–292

39.32 P.A. Torres-Carrasquillo, D.A. Reynolds, R.J. Deller Jr.: Language identification using Gaussian mixture model tokenization, Proc. ICASSP (2002) pp. 757–760

39.33 J.R. Bellegarda: Exploiting latent semantic information in statistical language modeling, Proc. IEEE **88**(8), 1279–1296 (2000)

39.34 C.E. Shannon: Prediction the Entropy of Printed English, Bell Syst. Tech. J. **30**, 50–64 (1951)

39.35 H. Li, B. Ma, R. Tong: Vector-based spoken language recognition using output coding, Proc. Interspeech (2006)

39.36 C.-H. Lee, F.K. Soong, B.-H. Juang: A segment model based approach to speech recognition, Proc. ICASSP (1988) pp. 501–504

39.37 S. Gao, B. Ma, H. Li, C.-H. Lee: A text-categorization approach to spoken language identification, Proc. Interspeech (2005) pp. 2837–2840

39.38 D.A. Reynolds, R.C. Rose: Robust text-independent speaker identification using Gaussian mixture speaker models, IEEE Trans. Speech Audio Process. **3**(1), 72–83 (1995)

39.39 V. Vapnik: *The Nature of Statistical Learning Theory* (Springer, Berlin, Heidelberg 1995)

39.40 W.M. Campbell, T. Gleason, J. Navratil, D. Reynolds, W. Shen, E. Singer, P.A. Torres-Carrasquillo: Advanced language recognition using cepstra and phonotactics: MITLL system performance on the NIST 2005 language recongnition evaluation, Proc. IEEE Odyssey Speaker and Language Recognition Workshop (2006)

39.41 S. Gao, W. Wu, C.-H. Lee, T.-S. Chua: A MFoM learning approach to robust multiclass multi-label text categorization, Proc. ICML (2004) pp. 42–49

39.42 J. Li, S. Yaman, C.-H. Lee, B. Ma, R. Tong, D. Zhu, H. Li: Language recognition based on score distribution feature vectors and discriminative classfier fusion, Proc. IEEE Odyssey Speaker and Language Reognition Workshop (2006)

39.43 K.S. Jones: A statistical interpretation of term specificity and its application in retrieval, J. Docum. **28**, 11–20 (1972)

39.44 J. Chu-Carroll, B. Carpenter: Vector-based natural languagecall routing, Computat. Linguist. **25**(3), 361–388 (1999)

39.45 H.K.J. Kuo, C.-H. Lee: Discriminative training of natural language call routers, IEEE Trans. Speech Audio Process. **11**(1), 24–35 (2003)

39.46 L. Breiman, J.H. Friedman, R.A. Olshen, C.J. Stone: *Classification and Regression Trees* (Chapman Hall, New York 1984)

39.47 S. Haykin: *Neural Networks: A Comprehensive Foundation* (McMillan, Englewood 1994)

39.48 S. Katagiri, C.-H. Lee: A new hybrid algorithm for speech recognition based on HMM segmentation and discriminative classification, IEEE Trans. Speech Audio Process. **1**(4), 421–430 (1993)

39.49 K.-Y. Su, C.-H. Lee: Speech recognition using weighted HMM and subspace projection approaches, IEEE Trans. Speech Audio Process. **2**(1), 69–79 (1994)

39.50 C.-H. Lee: A unified statistical hypothesis testing approach to speaker verification and verbal information verification, Proc. COST Workshop on Speech Technology in the Public Telephone Network: Where are we today? (1997) pp. 62–73

39.51 E.L. Lehmann: *Testing Statistical Hypotheses* (Wiley, New York 1959)

39.52 B.-H. Juang, W. Chou, C.-H. Lee: Discriminative methods for speech recognition, IEEE Trans. Speech Audio Process. **5**(3), 257–265 (1997)

39.53 S. Katagiri, B.-H. Juang, C.-H. Lee: Pattern recognition using a generalized probabilistic descent method, Proc. IEEE **86**(11), 2345–2373 (1998)

39.54 Y. Tsao, J. Li, C.-H. Lee: A study on separation between acoustic models and its applications, Proc. InterSpeech (2005)

39.55 M. Rahim, C.-H. Lee: String-based minimum verification error (SB-MVE) training for speech recognition, Comput. Speech Lang. **11**(2), 147–160 (1997)

39.56 S.E. Fahlman: An empirical study of learning speed in back-propagation networks, CMU CS Tech. Rep. **CMU-CS-88-162** (1998)

39.57 J. Li, M. Yuan, C.-H. Lee: Soft margin estimation of hidden Markov model parameters, Proc. InterSpeech (2006)