Pattern Recognition 41 (2008) 2965 – 2979

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Invited paper: Automatic speech recognition: History, methods and challenges[☆]

Douglas O'Shaughnessy

INRS-EMT, University of Quebec, 800 de la Gauchetiere West, Montreal, Quebec, Canada H5A 1K6

ARTICLE INFO

Article history:
Received 22 February 2008
Received in revised form 2 May 2008
Accepted 7 May 2008

Keywords:
Automatic speech recognition
Hidden Markov models
Adaptation
Compensation
Pattern recognition
Spectral representation

ABSTRACT

The field of automatic speech recognition (ASR) is discussed from the viewpoint of pattern recognition (PR). This tutorial examines the problem area, its methods, successes and failures, focusing on the nature of the speech signal and techniques to accomplish useful data reduction. Comparison is made with other areas of PR. Suggestions are given for areas of future progress.

1. Introduction

Among the tasks for which machines may simulate human behavior, automatic speech recognition (ASR) has been foremost since the advent of computers. The logical partner of ASR, automatic speech synthesis, existed before practical computing machines, although the quality of synthetic speech has only recently become reasonable. In earlier times, devices were built that approximated the acoustics of human vocal tracts (VTs), as the basic mechanisms of speech production were evident to early scientists, using models based upon musical instruments. A device to understand speech, however, needed a calculating machine capable of making complex decisions, and, practically, one that could function as rapidly as humans. As a result, ASR has grown roughly in proportion to other areas of pattern recognition (PR), in large part based on the power of computers to capture a relevant signal and transform it into pertinent information, i.e., recognizing a pattern in the (speech) signal.

As in any PR task, ASR seeks to understand patterns or ``information'' in an input (speech) waveform. For such tasks, an algorithm designer must estimate the nature of what ``patterns'' are sought. The target patterns in image PR, for example, vary widely: people, objects, lighting, etc. When processing audio signals such as speech, target information is perhaps less varied than video, but there is nonetheless a wide range of interesting patterns to distill from speech signals. The most common objective of ASR is a textual translation of the speech signal, i.e., the text corresponding to what one has said. Other useful outputs include: the language of the speech, the speaker's emotional state, and the speaker's identity [1]. A very practical use for ASR is as part (along with natural language understanding and automatic speech synthesis) of a human–machine dialogue, whereby a user can interact efficiently with a database, e.g., telephony [2].

Image and speech PR have both similarities and differences. In a sense, video has much greater variability than audio. Many images that meet the eye (or camera), whether natural or artificial (e.g., art, construction), vary greatly in their production, whereas the vast majority of sounds that meets the ear (or microphone) falls into a smaller set of categories. The latter include speech, music, animal sounds, machine sounds, and environmental sounds. In each of these audio classes, there are many features that help humans identify their sound source rapidly: periodicity, directionality, dynamic nature, spectral balance, etc. Such features can, of course, be exploited by machine PR, and we shall describe how this is done for ASR.

For speech, whether produced naturally by a human or reproduced by a machine, the sound origin (as typically assumed by a listener) is a speaker's VT. Thus, ASR has an input signal that is quite different from images, where input may be any display in the form of a gray-scale (or colored) pattern in two spatial dimensions (or in three dimensions for video, adding time as a variable). Human viewers of an image (or image sequence in time) usually try to impose or assume some physical ``structure,'' in terms of reference patterns, while trying to interpret the image, but the potential range of

[☆] This work was supported by the Canadian government under the NSERC Strategic Partnerships Program.

E-mail address: dougo@emt.inrs.ca.

0031-3203/$30.00 © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2008.05.008

possibilities for images is indeed vast. For audio input, on the other hand, a listener will normally and readily label different parts of what they hear as coming from various elements of a limited set of classes (i.e., speech, music, etc). For speech specifically, the restrictions on possible sounds are significant; listeners will normally reject (as non-speech) any audio signal that could not have originated in a VT, in their experience of speech communication. When listening to speech, they assume a VT source and decipher the audio content in terms of what the speaker likely had in mind.

1.1. Variability in speech

While emphasizing the major difference in diversity between speech and various other signals (e.g., images) that are processed by humans, one must note nonetheless a large range of variability in speech signals [3]. Each person has a different VT, controlled by a unique brain. While speakers of any given language follow the same general linguistic rules, there is great latitude in how this is done, producing a vast range of ``acceptable'' utterances that would normally be properly interpreted by most listeners. It is impossible for humans to reproduce the same exact action twice; even when attempting to repeat a word uniformly, slight variations occur. These changes are readily observed in digital representations of speech signals.

Some ASR systems focus on a very limited number of speakers, e.g., subscribers to a service or purchasers of a specific ASR product. In such ``speaker-dependent'' (SD) cases, speech variations are typically less vast (vs. ``speaker-independent'' (SI) cases, where an ASR system makes no assumption of who is talking). However, even when speech is limited to one cooperative speaker, significant variations are often evident owing to environmental (e.g., different communication channels) and speaking conditions (e.g., words in different contexts). When we generalize the ASR task to be SI, as in most services for the general public, we face the much larger range of variability that arises from different people, with their varied VTs and diverse styles of speaking.

The biggest challenge for ASR is how to handle all this variability. As in any PR, a designer develops models or templates for signals of interest, from observed ``training'' data in an initial development phase, and then verifies the performance of the algorithm on new ``testing'' data. (As in all PR, it is essential to test on data not employed during training, as otherwise the risk is great that models would be ``over-trained'' toward the data they have already seen, and thus under-generalized for future variations.) For ASR, a set of speakers typically reads chosen texts, and models are developed from this speech. ASR accuracy is usually proportional to the empirical similarity between training and testing data. For example, we may get high accuracy if an ASR model is properly developed for a single speaker repeating a word many times in a quiet environment, then testing the system with new versions of that same word from that speaker in the same environment. However, if we then test on a different speaker, with a different microphone, or add some background noise, we usually get reduced (and often much lower) accuracy. This is called the mismatch problem. The challenge for ASR designers is to amass sufficient data and employ a good training algorithm. In recent years, great strides have been made toward obtaining adequate databases for training, but many speech databases are insufficiently labeled as to their content, and few are reliably labeled to a precision of individual phonemes (TIMIT is the most common one in ASR research) (see Table 1) [75]. In addition, many databases employ read speech (to facilitate the labeling process, and to allow clear scientific experiments, for purposes of control), yet practical applications for ASR involve spontaneous speech, for which ASR is much more difficult than for read speech, owing to the greater variability in speech when one has to think as one speaks (reading is simpler cognitively than spontaneous speech). In practice, models for spontaneous-speech ASR often derive from examples of conversations.

A major challenge for ASR is to overcome the ``mismatch'' problem, where very often a system is faced with testing speech that is a poor match for the speech the recognizer was trained on. Intra-speaker variability (i.e., speaker freedom) is usually handled reasonably well via statistical models. Inter-speaker variability seems to be a greater problem: in SD systems, each user trains the system to ``learn'' his voice, and only models for that speaker are examined for recognition. In SI systems, at least dozens of speakers provide multiple training tokens for each unit. The simplest approach merges all speakers into a single model for each phoneme. However, in such cases, the state probability density functions (PDFs) tend to broaden significantly (larger variances), causing reduced discrimination between unit classes. One way to reduce this loss of discrimination is to have models for different groups of speakers, e.g., for classes of speakers (e.g., men vs. women, different dialects). The disadvantage is increased computation, since the input speech passes through all potential models (running a gender or dialect detector as a precursor is rarely done, owing to the high risk of error). This approach of multiple models to handle environmental variability easily extends to background and transmission channels.

ASR systems are often speaker-adaptive: for a given input (assumed to be from a single speaker), one starts with an SI system, and then adapts the system parameters to the new individual user's voice [4–6]. (In audio-conference applications, one could also apply speaker tracking, to estimate when the input voice changes identity, so as to restart the adaptation.) Among the common methods of adaptation are maximum a posteriori (MAP) [7,8] (which often requires several minutes of training data, because only those models corresponding to identified sounds in the adaptation speech are modified), maximum likelihood linear regression (MLLR) [9,10] (which calculates transforms of speaker space using unsupervised adaptation data, grouped automatically into similar data sets), vector-field smoothing (adapting parameters across models incrementally), ``eigenvoices'' [11], and vocal-tract-length normalization (VTLN) (where one estimates a speaker's VT length [12]).

The most difficult variability that ASR must handle is due to background, channel noise, and other external distortions [13]. Basic spectral subtraction techniques can help with additive noise, while some cepstral methods (which convert multiplication in the spectral domain to cepstral addition) suppress convolutional noise. Many methods that are used to enhance noisy speech for human listening can be used as preprocessors for ASR. In noisy cases, one should focus on the high-amplitude parts of the input signal spectrum: strong speech formants are the most relevant for speech perception, and are relatively less corrupted by noise [14]. Two methods are normally used: robust parameterization (seek analysis parameters that are resistant to noise) or model transformation (adapt the ASR models to accommodate the distortion).

Cepstral mean subtraction (CMS), like RASTA processing [15], eliminates very slowly varying signal aspects (presumed to be mostly from channel distortion). The mean value for each parameter over time (typically for periods exceeding 250 ms) is subtracted from each frame's parameter, thus minimizing environmental and intra-speaker effects. Channel noise is often assumed to be constant over an utterance, but portable telephones suffer fading channel effects, which require more frequent estimations [16]. Another example of a model transformation to improve ASR is parallel model combination (PMC) [17].

1.2. Brief history of ASR

Using analog circuitry, Bell Labs demonstrated small-vocabulary recognition for digits spoken over the telephone in 1952. As

Table 1
Details of typical databases used in ASR evaluation

Database	Style	Vocabulary size	Perplexity	WER 1980s (%)	WER 1990s (%)	WER 2000s (%)
TIMIT	Read	100s	16			5
Resource Management	Read	1000	60	15	3	
Air Travel Information System	Oz	2000	15	18		3
N. American Business News	Read	64 000	200		13	7
Wall Street Journal	Read	60 000	80	17	9	
Broadcast News	Read	210 000				15
Switchboard	Spontaneous	45 000	85		50	25
Call Home	Spontaneous	28 000	210		40	
Aurora digits	Read	11	11			0.3

Typical percentages of word error rate (WER) are shown in the last three decades. Oz = ``Wizard-of-Oz'' style, asking a simulated travel agent.

Table 2
Major advances in ASR methodology

Advance	Date	Impact
Linear predictive coding	1969	Automatic, simple speech compression
Dynamic time warping	1970s	Reduces search while allowing temporal flexibility
Hidden Markov models	1975	Treat both temporal and spectral variation statistically
Mel-frequency cepstrum	1980	Improved auditory-based speech compression
Language models	1980s	Including language redundancy improves ASR accuracy
Neural networks	1980s	Excellent static nonlinear classifier
Kernel-based classifiers	1998	Better discriminative training
Dynamic Bayesian networks	1999	More general statistical networks

In some cases, the dates are approximate, as they reflect a gradual acceptance of new technology, rather than a specific breakthrough event.

computers grew in power during the 1960s, filter banks were combined with dynamic programming to produce the first practical recognizers, mostly for words spoken in isolation (i.e., with pause after each word), so as to simplify the task. The 1970s saw much progress using custom special-purpose hardware in commercial small-vocabulary applications over the telephone. Linear predictive coding (LPC) became a dominant ASR representation, as an automatic and efficient method to represent speech. LPC is still the standard today in cellphone speech transmissions, but was replaced for ASR by the MFCC (Mel-frequency cepstral coefficient) approach (see below) in the 1980s. This decade also saw the creation of large widely available databases in several languages, allowing comparative testing and evaluation (see Table 2). During the 1990s, commercial applications evolved from isolated-word dictation systems to general-purpose continuous-speech systems. Since the mid-1990s, ASR has been largely implemented all in software. Medical reporting and legal dictation have been two driving applications, as well as automation of services to the public over the telephone. Core ASR methodology has evolved from expert-system approaches in the 1970s, using spectral resonance (formant) tracking, to the modern statistical method of Markov models based on MFCC, which has remained the dominant ASR methodology since the late 1980s. In the 1970s ASR focused on simulating the human processes as closely as possible. The difficulty of handling the immense amount of variability in speech production (and transmission channels) led to the failure of simple if-then decision-tree approaches to ASR.

A major issue in ASR over the years has been how to handle both temporal and spectral variability in speech. Around 1980, in ASR it was common to nonlinearly stretch (or warp) templates to be compared, to try to synchronize similar acoustic segments in test and reference patterns. This dynamic time warping (DTW) procedure is still used today in some applications [18]. DTW combines alignment and distance computation in one dynamic programming procedure [19–21], to find an optimal path through a network of possibilities. Linear time alignment is a special case of DTW, where only a single path comparing synchronous frames between templates is considered. In DTW, small deviations from this linear frame-by-frame comparison are allowed if the distance for a frame pair slightly off the main path is smaller than other local frame comparisons. DTW finds a time warping that minimizes the total distance measure, summing the measures of successive frame-to-frame matches.

In the early 1980s, it was thus common to compare sets of specific templates of target units, such as phonemes, to each testing unit, eventually selecting the one with the closest match as the estimated label for the input unit. This led to high levels of computation, as well as difficulty in determining which and how many templates to be used in the test search. Since then, the standard has been hidden Markov models (HMMs), in which statistical models replace templates, as the former have the power to transform large numbers of training exemplars into simpler probabilistic models. Instead of seeking the template closest to a test frame, test data are evaluated against sets of PDFs, selecting the PDF with the highest probability.

In the 1990s, ASR experimented with wavelets, where the variable time-frequency tiling more closely matches human perception, but the nonlinearity of wavelets has been a major obstacle to their use [22]. Artificial neural networks (ANNs) and support vector machines (SVMs) have also found recent application in ASR, but are not as versatile as HMMs [23,24]. SVMs maximize the distance (called the ``margin'') between the observed data samples and the function used to classify the data. They generalize better than ANNs, and tend to be better than most nonlinear classifiers for noisy speech. Unlike HMMs, SVMs are essentially binary classifiers, and do not provide a direct probability estimation. They need to be modified to handle general ASR, where input is usually not just ``yes'' versus ``no.'' HMMs do better on problems such as temporal duration normalization and segmentation of speech [25], as basic SVMs expect a fixed-length input vector.

ANNs have not replaced HMMs for ASR, owing to their relative inflexibility to handle timing variability. Among promising new approaches is the idea that we must focus attention on specific patterns in both time and frequency, and not simplistically force the ASR analysis into a frame-by-frame approach [26]. Recent progress has occurred in the use of finite state networks, statistical learning algorithms, discriminative training, and kernel-based methods [27].

As is often the case in engineering applications, one must make compromises in ASR. We will examine where these compromises are often made and motivate them, as assumptions about practicality that were made in one era may no longer apply later (i.e., necessary compromises owing to limited computer power in the 1980s or 1990s have gradually been lifted in recent years, leading to improved ASR accuracy). Current training methods have relied heavily on simple procedures, e.g., steepest gradient descent, Bayesian modeling, first-order HMMs [28]. The reality of speech communication is that human-to-human transfer of information via voice is highly complex, and involves many disciplines: acoustics, phonetics, linguistics, psychology, transmission media, and VT articulation.

1.3. General ASR methods

As in any PR task, ASR follows a standard procedure: signal capture, data reduction, feature comparison, distance/similarity metric (or likelihood) model, and decision to minimize cost or error. A major challenge for any PR is the choice of how to reduce or compress the acquired data, while minimizing loss of relevant information. Data reduction is essential in all PR, not just for efficiency, but also to focus the recognition process on the relevant aspects of an input signal. Similar challenges occur in the choice of a comparison metric, which relates to how speech is modeled during data reduction.

The objective in PR is to classify an unknown pattern as one from a set of candidate groups. For speech, this usually means labeling each input utterance with its corresponding text. In full generality, an audio input could be any sound waveform and the output could comprise all possible texts. For tutorial purposes here, we will sometimes restrict utterances to be simple words or even just phoneme sounds; however, practical ASR must accept any sound sequence that may enter a microphone as a possible intended speech signal to be interpreted.

In some types of PR, the output classes are artificial, e.g., in manufacturing quality control, a PR system may be used to verify the precision of produced goods matching some ideal template. Speech units have no such ideal templates for comparison, as each person's phonemes and words lie within a range of acceptable sound patterns for which ``targets'' may be hard to define. Speech does, however, share with most PR the difficulty of establishing suitable feature domains and distance metrics. Even if one is given an ideal template for each of N classes (for a given PR application, say), it is rarely self-evident which features to extract in data acquisition, nor which parameter dimensions are most important. A PR designer must delineate differences among pattern classes (e.g., what distinguishes a good product from a bad one, or an /i/ versus an /u/ or other phoneme) (/.../ refers to phonemes, e.g., as in the International Phonetic Alphabet, and employed in many dictionaries).

A common approach to PR design is to seek many examples of acceptable members of each class to be recognized, e.g., ask a speaker (or group of speakers) to utter a set of words (e.g., the digits 0–10, as one very practical application for ASR is the recognition of telephone or credit card numbers). This provides a set of exemplars or ``templates'' for each class. We could then simply compare an unknown pattern (e.g., input utterance) to each template, and assign the unknown to the class of the best-matching template. This approach, while practical, is not efficient, due to wasteful duplication of information among sets of redundant templates, as well as much time spent comparing an unknown against many similar exemplars from each class.

For ASR training, an alternative to using sets of exemplars is to create one or more PDFs for each class to recognize. Then, during the recognition phase, one evaluates the likelihood that each PDF may have produced the pattern under test. The one yielding the highest probability is thus identified as the output class.

Whether template-based or statistical, PR needs to specify relevant parameters or features, to establish a relevant search space. Poor data reduction, leading to inappropriate feature spaces, dooms PR to poor accuracy. It is often assumed that minimizing a distance between a template and an unknown pattern is equivalent to finding the PDF with the highest likelihood, yet this assumes a proper feature space, where each parameter has utility in discriminating among the PR candidate classes. So, one of the major objectives in ASR is to seek such useful feature spaces.

To assist this search, we first examine the nature of speech signals. Before we can discuss how modern ASR methods accomplish the crucial steps of PR, it is appropriate first to examine what speech is, to better decide what aspects may be relevant. The accuracy of many a PR system is often proportional to how well its designer identifies useful features in its input signals.

2. Speech production

Typically, speech is generated by human speakers, who wish to convey information to listeners. Thus, it is useful to review human speech production in the context of seeking acoustic aspects that are pertinent for speech perception. Human speech communication differs greatly from artificial communication systems, e.g., radio or data transmission. In the latter, a system designer has significant control of the nature of transmitted signals, and seeks to minimize cost (spectral bandwidth, signal energy, and signal durations); e.g., to send a bit sequence in a noisy Gaussian channel, one may use orthogonal sinusoidal bursts with matched receiver filters. In artificial channels, designers can adjust many aspects of the transmission to optimize communication. On the other hand, ASR designers have little control over speech production; one might recommend that speakers talk more slowly or clearly, but it is rare to ask users to adapt their style of speech.

During evolution, the human speech system likely developed along lines of economy; humans generally wish to communicate the most information with the least effort as quickly as possible. We are not confined to a specific bandwidth for our speech channel, unlike electronic audio transmission. The latter (AM, FM, etc.) allocates bands for transmission, which affect the quality of the signals. For natural speech communication, the obvious channel is the air between speaker and listener, in which we use the base-band of the lowest audio frequencies (e.g., at most 0–20 kHz)—the band to which our ears are most sensitive and for which our VTs are capable of producing useful sounds.

Given a time-frequency ``space'' in which to send audio information, languages seem to have all developed a communicative procedure in which speech is a sequence of sounds (phonemes), as drawn from an inventory (i.e., the set of phonemes particular to each language). Local sequences of a few phonemes group conceptually into words, which have logical meaning as objects, actions, and descriptions. (The phonemes themselves are components of words, and have no syntactic or semantic meaning; the acoustics of phonemes serve only to convey their identity to listeners.)

2.1. Phoneme units

The relative rate of speech varies greatly across speakers and conditions, but normally is 10–12 phonemes/s. Each rate is a compromise: speaking more rapidly might accomplish faster data conveyance, but only if the information is reliably received by the listener, and too rapid speech can lead to lower perceptual accuracy. In addition, rate relates to speaking effort. So a balance occurs, with variations depending on conditions, e.g., in noisy environments and when talking to strangers, one speaks more slowly. We could also make an analogy here to the Shannon communication theorem, in which bit rate varies with channel conditions; the rate of conveyed speech information is clearly affected by communication conditions.

The basic content of speech is thus a sequence of phonemes, whose identity (once decoded) can be transformed into a word sequence, via access to a dictionary and other natural language processing (whether by ASR or by a human listener). Each phoneme corresponds to an approximate VT shape. A sound source in the VT (at the glottis and/or a constriction in the upper VT) excites the VT, which acts as a filter to yield the output speech. We characterize phonemes in three feature classes: manner of articulation, place of articulation, and voicing. ``Manner'' refers to a sound's major class of VT configuration: complete closure (for ``stops'', where the pressure behind a closure in the VT is released suddenly, i.e., /p,t,k,b,d,g/), major constriction (for ``fricatives,'' where noise is generated by air passing through a slit, e.g., /f,s,v,z,h/), participation of the nasal cavity (for ``nasals,'' where the mouth is closed but the velum lowered, e.g., /m,n/), relatively open VT (for ``vowels''), and a fifth category (liquids and semivowels, /l,r,w,j/) for some vowel-like ``sonorant'' sounds that are considered to be consonants (owing to their particular constrictions that do not cause noise). The ``place'' of articulation of a phoneme refers to the sound's major point or location of VT constriction: if a major constriction (i.e., for the ``obstruents''—stops and fricatives), the place is a single location along the length of the VT; if the constriction allows air to pass without noise generation, then ``place'' may involve 2–3 parameters (typically, the height and lateral position of the tongue tip). ``Voicing'' refers to the presence of vocal fold vibration, which renders the phoneme quasi-periodic; the rate is called the fundamental frequency (F0) of speech (also known as ``pitch'').

ASR systems typically have the greatest success in classifying manner, at least when using only three broad categories—sonorants, stops and fricatives—as such sounds have relatively robust features of strong periodic energy, silence, and noise, respectively. Detection of periodicity is a major factor in estimating voicing. Further phonemic detail, including place and subtler manner distinctions (e.g., nasal versus liquid), is much more challenging for ASR. In decoding speech for ASR, we seek to estimate its sequence of VT shapes (and, sometimes, vocal cord state), indirectly through the sequence of spectral patterns of the speech signal. The VT shape for each phoneme specifies a series of resonances of the VT, called formants. When the full VT is used (i.e., excited at the glottis), a typical male VT of 17 cm in length has resonances every 1000 Hz on average; e.g., a neutral VT has F1 (the lowest formant resonance) at approximately 500 Hz, F2 at 1500 Hz, etc.

In most sounds (sonorants), energy decreases with increasing frequency, owing to the low-pass nature of their periodic glottal excitation. Therefore the lowest-frequency formants are the most reliably heard, especially in noisy environments. Speakers tend to exercise significant control over the positioning of F1, F2 and F3 (which have the vast majority of energy in sonorants). These resonances are also well preserved in common transmission channels (e.g., telephone), and are readily perceived by listeners (for which the broad region near 1–2 kHz is optimal for perception).

Most languages have developed, in their phonemic vocabulary, a range of vowels well spread in acoustic space, to facilitate reliable discrimination by listeners. For example, all languages seem to have the vowels /i/, /a/ and /u/, which lie at approximate extremes of VT shape: /a/ having the lowest tongue position, /i/ and /u/ the highest, with /i/ the most forward and /u/ the most posterior. These three shapes correspond roughly to extremes in formant positions: F1 correlates well with inverse tongue height (e.g., for a typical man: low at 300 Hz for /i,u/ and high at 750 Hz for /a/), while F2 follows lateral tongue position (low for the back /u/, and high for the front /i/). Higher formants (F3, F4, etc.) assist in vowel perception, but to a much lesser degree. ASR usually encodes all available bandwidth, assuming that high-frequency spectra give some benefit.

The higher frequency range is of use mostly for obstruents, where the excitation occurs much higher in the VT than at the glottis. Such sounds have very little energy below 2.5 kHz (with the possible exception of a ``voice bar'' at extremely low frequencies, which can cue the presence of vocal cord vibration in voiced obstruents). In obstruents, only the forward portion of the VT is excited, and the correspondingly much shorter acoustic tube has higher resonance frequencies than the sonorants. Experiments with synthetic speech suggest that spectral detail in obstruents can be represented much more coarsely than for sonorants. Some overly simple models represent such sounds with just a high-pass spectrum, whose cutoff frequency describes the place of articulation: a low cutoff, at say 2.5 kHz for palatal fricatives (e.g., sh), 3.5 kHz for /s,z/ and much higher for the weak dental fricatives (f, th).

2.2. Intonation

ASR systems tend to focus very little on aspects of excitation, including intensity and pitch. The popular MFCC and LPC analysis methods deal with a broad spectral representation that smoothes out F0 detail, and thus suppresses pitch. While ``pitch'' is a perceptual term, it is often nonetheless used interchangeably with F0 (with the latter being the inverse of the period duration that can be measured physically in the speech signal). Speakers use F0 to help signal many things to listeners: syntactic structure, the semantic importance of individual words, whether a sentence is a yes/no question, emotions, and (in tone languages) phonemic identity. However, as F0 does not integrate well with standard ASR analysis, it has frequently been ignored (even, somewhat surprisingly, for tone languages).

Intensity is a natural cue to distinguish strong versus weak sounds, and ASR systems often exploit this. However, many normalize out intensity as an unreliable factor, as channel and recording variability often lead to wide swings in intensity levels.

Duration is a third feature of speech (other than spectral envelope) that has direct practical use in speech production and perception. Duration, intensity and F0 are often considered to be the suprasegmentals or intonation of speech. These three tend to function at a level ``above'' the ``segmental'' level of the phoneme, in that they vary as a function of a wider context than phonemes. No speech unit is truly independent of its neighbors. Almost everything in speech communication varies with context; e.g., a sound is loud only compared to the intensity of its neighbors, sounds are long or short only versus adjacent sounds, and even the formant positions of vowels are judged in terms of other vowels spoken by a speaker. So most acoustic features in speech are judged in context. However, intonation is inherently contextual; e.g., the duration of a phoneme is only meaningful relative to that of its neighbors, whereas formant values have meaning for phonemes uttered in isolation. Speakers may readily vary baselines for intonation, by speaking lower or faster; such adjustments do not occur for spectral envelope, whose baseline is set by VT length.

Examples of useful intonation phenomena are the following. Vowels tend to be longer when immediately prior to obstruents that are voiced rather than unvoiced (e.g., ``bade'' vs. ``bait''). Vowels considered ``stressed'' have longer durations and more varied F0 than unstressed vowels. F0 tends to rise at the end of questions that request a yes-or-no answer, as well as at the end of phrases not ending a sentence. While a typical phoneme averages about 80 ms in duration, there is wide variability: flapped /t/'s can be as brief as 10 ms, while stressed diphthongs range to 200 ms and more. Consonants have shorter durations in clusters; e.g., nasals in ``limp,'' ``lint'' and ``link'' are very brief. The final syllables of major syntactic phrases lengthen up to 200 ms. Words that are less common have longer durations than frequently used words.

Words more important for the speech message tend to be clearer and more stressed. Some ASR focuses on the identification of such ``keywords'' [29,30]. While humans use intonation greatly, few ASR systems exploit intonation efficiently [31]. This is in part owing to its complex relationships to texts, but mostly because HMMs handle longer-term acoustic effects relatively poorly. Future ASR must exploit intonation better [32].

2.3. Coarticulation

In addition to this ``evaluation by comparison'', we have the phenomenon of ``coarticulation''. Phonemes are uttered in sequence, with pauses typically only occurring every few seconds; it is normal to utter dozens of phonemes without an intervening pause. In such sequences, each phoneme affects its immediate neighbors in many ways. While each phoneme has a nominal ``target'' VT position or shape, speakers compromise in several ways, often undershooting such targets. In anticipatory coarticulation, a speaker often plans ahead and starts moving certain parts (articulators) of the VT toward positions for future phonemes (sometimes well before the actual phoneme is pronounced). By a similar mechanism, if an articulator is not explicitly needed for a given phoneme, it may remain in place long after the phoneme that caused its movement; e.g., in the word ``strewn'', lips may round during the /s/ in anticipation of /u/, despite lips normally not being rounded during the intervening /t/ and /r/.

The immediate effects of coarticulation that occur in adjacent phonemes are typically accommodated in ASR via context-dependent (CD) models. Simpler ASR systems use context-independent (CI) models, which ignore coarticulation (at the cost of lower recognition accuracy). As a language typically has 30–40 phonemes, one only needs this many such CI models to handle that language, which is very efficient. As most segmental coarticulation affects phonemes that are immediately adjacent, many ASR systems use CD models for ``triphones,'' sequences of three phonemes; e.g., assuming N phonemes in a language, each phoneme would have up to N^2 models, one for each possible context of its left and right phoneme neighbors. Many systems prune these to a smaller set, as groups of phonemes sharing certain traits (e.g., with the same place of articulation) have similar coarticulation effects, and thus one can ``cluster'' models to reduce the need for redundant models. Recent ASR models may use articulatory features (using labels related to place and manner) to augment their usual acoustic parameters [33].

3. What should ASR systems look for?

In selecting parameters for ASR, we use the discussion above as a guideline. The peaks of the speech signal's spectral envelope (especially the center frequencies of F1, F3 and F3) seem to be very pertinent features [34], for which various VT shapes used in sonorants cause reliable dispersion of phonemes in F1–F2–F3 space. In addition, the human auditory system seems well tuned to perceive variations in such spectral peak positions [35]. Of less relevance appear to be the formant bandwidths; these are less readily controlled by speakers, and less easily distinguished by listeners. (Similar comments hold for the general fall-off spectral slope: sonorant spectra generally decline at an approximate rate of -6 dB/octave, owing mostly to the low-pass nature of glottal excitation, and variations in such slope generally evoke little perceptual notice.)

Of clear importance to speech perception (and hence to ASR) is the general intensity of speech. Sonorants are much stronger than obstruents, and, within these two classes of sounds, intensity also varies reliably; e.g., /a/ stronger than /i/, /s/ than /f/. Such distinctions can be achieved based on spectral peak position alone, without the cue of intensity, but intensity (although often redundant) is easily measurable and commonly used by listeners. Nonetheless, ASR often uses intensity less than human listeners do, as the level of a speech signal varies greatly with recording conditions.

A common form of signal normalization in ASR is to await the end of an utterance, and then subtract the average value from each parameter in the time sequence. This ``differential'' analysis focuses ASR's attention on frame-to-frame changes in speech, rather than on absolute values. Such differentiating can still allow ASR to notice relatively loud sounds, in terms of a series of frame-to-frame increases in intensity. However, most ASR systems use very localized feature measures, owing to the first-order Markov models employed.

3.1. Challenges for ASR

One difference between human and artificial communication systems is the lack of symmetry between transmitter and receiver for speech signals. In many communication systems, one designs a receiver to invert all processing (e.g., data compression) done at the encoder. In humans, on the other hand, hearing developed long before speech did; indeed, the hearing systems for most animals are similar, whereas only humans speak. The auditory system likely evolved to avoid dangers. Humans discovered much later the possibilities of complex sounds from their VTs for useful communication. The vast difference in anatomy and physiology between the speech production and perception systems in humans renders its analysis difficult. Humans learned that manipulations of their tongue, lips, jaw, and vocal cords, combined with exhaling, could reliably convey complex ideas. As the VT shares its speech function with breathing and eating, it is far from ideally designed to produce arbitrary efficient sounds for the ears.

Successful ASR usually yields a textual estimate of what was said. Thus one supposes that the initial step in most speech production starts with an intended text in the speaker's head (such a text may not always be well formed prior to being spoken). In the human communication process, there are several transformations (many nonlinear) of information: (1) thought-to-articulation, (2) VT movements-to-acoustical signal, (3) propagation of the speech signal to a microphone, (4) electronic transmission/storage, (5) loudspeaker to the listener's ears, (6) acoustic-to-electrical in the inner ear, (7) interpretation by the listener's brain.

Steps 1 and 7 (those closest to our brains) remain little understood in any functional or practical sense. For step 1, we may examine indirect relationships between the intended text and observed VT positions (or their speech consequences). Step 2 deals with articulatory phonetics, and is of prime interest to ASR designers, as it describes the nature of speech. Steps 3–5 may merge into one step (propagation through the air between speaker and listener) if communication is without artificial means. In all cases, we simply assume that speech arriving at an ASR microphone may be distorted in such transmission, either by a noisy channel and/or the presence of other sound sources. Step 6 is of interest in suggesting suitable ways to do data compression in ASR. For step 7, researchers generally posit speech perception theories, treating the listener as a ``black box'', to whom questions can be posed about what was heard, given various auditory stimuli. Knowledge about steps 6 and 7 is useful for ASR design in noting the degree to which human listeners can discriminate relevant aspects in speech. For example, ASR likely need not quantify speech features beyond a resolution level that can be perceived (i.e., difference limens, or JNDs (just-noticeable-differences)), as speakers (who are listeners themselves) likely do not control their articulations beyond a precision that may be perceived.

3.2. Approaches to ASR

When designing a system to solve a problem, as in PR, one often aims for generality, i.e., the system should work for as wide a range of tasks as possible. Such an approach, however, may lead to overly general methods. A key element for success in any PR is to properly exploit all relevant features for the form (e.g., speech) one is trying to recognize. Thus, it is very useful to understand the nature of the form to be recognized, and to not simply apply ``off-the-shelf'' PR methods unless appropriate.

One approach to ASR design is to simulate the human auditory system (i.e., emulate a listener). In artificial intelligence (AI), one often uses a model of how humans accomplish a task as a guideline for a simulation. Typical AI algorithms show evidence of this, but also show large deviations; e.g., one often exploits the great serial power of computers to do calculations that are much different from how the (much more parallel) human neural system operates. In the ASR field, recent advances in performance have largely come from exploiting powerful computers and the availability of increased speech and language data.

While ASR may emulate the way a listener receives an incoming speech signal, it is perhaps more important to understand how the human produces speech, as it is that signal that ASR must process. Speech is generated in one's VT by expelling air from the lungs through the VT, which acts as a filter. This results in variations in air pressure in the region of the speaker's mouth (some sound may exit the nostrils, while weaker sound radiates from the throat and cheeks). The speech sound waves radiate outward, to be picked up by a listener's ears or a microphone. A speech signal, as usually displayed, shows amplitude as a function of time. This continuous-time signal is typically converted to a digital signal via an analog-to-digital (A/D) converter. At each further step of signal processing, one must be concerned about maintaining those aspects of speech that are necessary for accurate recognition, while doing appropriate data reduction to facilitate the process.

Choices for ASR in its initial processing stages that have practical ramifications are the sampling rate and the number of bits per sample in the A/D conversion. ASR systems rarely need to compromise here; they normally retain all available bandwidth, and use 16 bits/sample. Perceptual tests have shown that 8-bit logarithmic quantization (as used in telephony) is adequate to avoid deterioration in quality. As computers are organized in bytes of 8 bits, a de facto standard for speech sampling is 16 bits with linear quantization, which avoids the extra complexity of log conversion.

Speech has significant energy over a wide range of frequencies. However, the utility of energy above 10 kHz for ASR is very small. Thus, few ASR systems function above 20 000 samples/s, but rates vary among systems (while ASR is rarely done on speech from CDs, one would normally down-sample (decimate) from their excessively high 44 100/s sample rate—useful perhaps for music appreciation, but much higher than needed for speech). The most common reason for use of a relatively low sampling rate is to deal with speech that has passed through traditional telephone lines, whether landline or cellular, as such speech has very limited energy above 3.3 kHz. For such speech, the standard is 8000/s (hence the basic 64 kbps for landline telephone speech). It is clear that such a bandwidth limitation lowers both speech quality and ASR accuracy, but the widespread telephone network imposes this constraint. In many other cases, a compromise rate of either 10 000/s or 16 000/s is chosen, allowing access to the 0–5 or 0–8 kHz audio range, respectively (always obeying the Nyquist theorem, to avoid aliasing). Such choices are somewhat arbitrary, although there is clear improvement in going from 8000/s to 10 000/s, as the 3.3–5 kHz range, lost in telephone applications, contains useful information about obstruent phonemes. Ever-diminishing returns occur if we retain spectra above 5 kHz, as such high audio frequencies, while perceivable, have mostly redundant information. In cases of competing noise signals, where speech redundancy is often reduced and higher spectra might be therefore useful, signal-to-noise ratio is often low at high frequencies, again lessening the utility of the high range.

4. ASR choices

A main objective of ASR design is to minimize its word error rate (WER). Ideally, the parameters of ASR models could be chosen to reduce WER, but WER varies in a highly complex fashion as a function of all the model parameters. Simple steepest gradient optimization of parameters is common, but only locally optimal models result. Thus, there is a wide range of training methodologies. The basic maximum likelihood (ML) method chooses model parameters M to be arg max $P(O|W,M)$ over all observed speech data O for a given unit model (e.g., a word W). However, this approach does not take account of how likely O may be for other words, and thus it does not minimize WER. As a result, there has been great recent interest in alternative, discriminative training, in which the emphasis is placed on better separating the likely classes of words [36]. These include maximum mutual information estimation (MMI), minimum classification error (MCE), minimum phone error (MPE) [37], maximum entropy [38] and boosting [39]. MMI maximizes the mutual information between training data and their respective models, often via an optimization method called the extended Baum–Welch algorithm. MCE approximates WER by a smooth, differentiable objective function, using a generalized probabilistic descent (gradient) method to minimize the function. A weakness of these methods is that good performance on training data does not always extend to future unseen test data, i.e., the mismatch problem remains.

Recent research in machine learning has focussed on the margin between classes in acoustic spaces. Rather than simply adjust speech models so that WER is minimal on the training data, the space is remapped to push the classes as far apart as possible. Typically, a feature space (e.g., from an MFCC vector of a frame of speech) is mapped via a kernel to a space of higher dimension, which allows a linear classifier to readily separate speech classes with wide margins between them [27]. Earlier methods minimizing WER were satisfied with classes that minimized overlap (overlap causes word errors). However, later test data often fell into the overlap regions. Maximizing the margins, rather than only avoiding overlap, raises the likelihood of future good ASR performance.

A major problem for most ASR systems is robustness [40], which refers to the fact that they may be insufficiently general or overtrained if using small training sets. A truly robust ASR system should be able to properly decode speech from any speaker, in reasonable environments, and with reasonable microphones. In practice, non-native speakers, and indeed often simply speakers other than those in the training set, cause significant reductions in recognition accuracy. Environmental noise, from natural sources (e.g., weather, other talkers, etc.) or machines (e.g., cars), as well as communication link distortions (e.g., fading on cellular phones), all tend to degrade ASR performance, often severely. Even simple microphone substitution reduces accuracy. Human listeners, by contrast, often can adapt rapidly to these difficulties, which suggests that there remain significant flaws in current ASR. It is not, of course, necessary to directly mimic human perceptual processes when designing ASR systems (just as airplanes do not flap their wings to fly). However, much of what we know about human speech production and perception has yet to be integrated into practical ASR [41].

Most ASR employs one microphone to capture the desired speech signal. If the speaker is in a noise-filled environment, that microphone receives the sum of the desired speech and various other sounds (i.e., background noise). Some methods to improve ASR in such conditions rely on noise suppression or require other nearby microphones to pick up alternative versions of the noise. In this latter case, in suitable conditions of decoupling between the microphones, an estimate of the noise can be subtracted from the primary microphone signal, yielding a cleaner speech signal to process. However, the extra cost and inconvenience of needing other inputs has prohibited widespread use of this approach (e.g., it is totally infeasible for telephone applications, unless phones are redesigned for multiple-microphone input).

5. Spectral processing

It is theoretically possible to recognize speech directly from a digitized waveform, but virtually all ASR performs some spectral transformation on the speech time signal. Numerous experiments on human audition show that the inner ear acts as a spectral analyzer, phase has been rarely found to be useful for ASR, and analysis of human speech production shows that speakers tend to control the spectral content of their output much more than details of their speech waveforms.

Despite this, it is useful to examine whether ASR might be feasible directly on speech waveforms, as (potentially) forgoing the need for computation of spectra could simplify processing. In theory, one could record many speech exemplars, each labeled with its corresponding text, and then simply search this database for the closest match when trying to identify the text for each new unknown input utterance. However, the search space is enormous, with no evident way to make practical compromises. While it is feasible to create a very large database of millions of labeled utterances, such would always fall well short of that needed for successful waveform-based ASR. Note how many possible waveforms exist just for a brief half-second word such as ``yes'' (one simple yet practical application is to identify ``yes'' versus ``no''). It is possible to reasonably represent the major spectral content of speech with as few as 2 kbps. So, to represent a brief word would require 1000 bits, thus many billions (2^{1000}) of potential waveforms. The vast majority of such signals do not correspond to speech, but there is no evident way to prune this possible set. Thus, assuming we operate with a large (but far from complete) database of labeled speech signals, it would be necessary to devise a suitable distance metric so as to find the closest match to an input utterance. However, speakers appear to have little direct control over waveform details in the time domain; when one repeats words as closely as possible (making them identical perceptually), one still observes large differences in time-sample sequences, owing to small variations in phase below the level of perception but above the level of typical signal representations. In addition, there has been little success in finding a suitable distance metric in the time domain. As a result, we shall assume that ASR must operate in another domain, and the obvious choice is spectral.

5.1. Fourier transform

The simplest spectral mapping is the Fourier transform (FT), realized in the digital domain as the (discrete) DFT (or, in practice, the (fast) FFT). Of immediate interest is the size N of the DFT, i.e., the number of points in the time frame, which is also the number of spectral samples. Speech is a non-stationary signal, as the VT and its excitation change with time to cue the sequence of phonemes. To approximate stationarity, one normally limits a speech sequence $s(n)$ in the time domain by multiplying by a window $w(n)$, assuming that the signal within the window ``frame'' does not change characteristics. A common choice for windows is 20–30 ms, often choosing a specific value in terms of the number of samples that is a power of 2 (e.g., 256 samples, using a sampling rate of 10 000/s), to facilitate the use of a radix-2 FFT [22]. This duration is a compromise: most of speech shows relatively slow changes within 20–30 ms, but rapid changes (e.g., 5–10 ms) occur when closures happen in the VT, as for stop and nasal consonants. In such cases, the spectral representation from a DFT is a version averaged or smeared in time (e.g., at a stop release, one may have an initial few milliseconds of silence, then a 10-ms burst, then some aspiration; its DFT would yield a smeared spectrum of the three components). An alternative to accepting a few smeared frames would be to use shorter frames, but that increases computation, requiring analysis at a higher rate (e.g., a 20–30 ms window allows an analysis frame rate of 50–100/s, and shorter frames would raise this rate). Since ASR functions reasonably at 50–100 frames/s, this rate is a common compromise.

DFT-length N imposes a single choice for both frame length and spectral resolution, but often these two do not suggest the same value. In the example above, the DFT has a resolution of about 40 Hz. If a higher resolution is desired, one may use a 512-point FFT, but still impose a window that is shorter than 512 points, by ``padding with zeros'', i.e., augmenting the short window with zero-valued samples. In all these cases, the windowing operation imposes a distortion on the resulting signal: the product $s(n)w(n)$ corresponds to convolving $S(k)$ and $W(k)$, their respective DFTs. Two common choices for $w(n)$ are the rectangular and Hamming (raised cosine) windows; both have relatively smooth time shapes, thus weighting the $s(n)$ samples mostly uniformly and with a dominance of low-frequency energy. The convolution smears the ``ideal'' $S(k)$ spectrum, leading to a distorted result. The spectral distortion is usually less with the preferred Hamming window, but that window significantly down-weights many of the samples within its window range. The choice of frame rate varies accordingly: with a rectangular window, one may set the rate exactly to the inverse of the frame duration, while treating all speech samples the same; with a Hamming window, one often uses overlapping frames, requiring a higher frame rate, as otherwise the samples at the edges of the window would be underutilized.

The DFT imposes a fixed frequency resolution, whereas it is known that the importance of frequencies is non-uniform. Human signal perception (and, perhaps as a consequence, human production of signals as well) tends to be more logarithmic than linear. Weber's law notes a general trend toward perceptual resolution that varies with the amplitude of a signal dimension, be it intensity or frequency. For example, in amplitude, we use the logarithmic decibel scale, to describe the vast range of amplitudes that humans can produce and perceive. Neural hair cells in the cochlea respond to audio selectively as a function of frequency, which has led to a model of the ear as a set of 24 band-pass filters, whose bandwidth is about 100 Hz at low frequency and increases with frequency (i.e., roughly constant Q of about 5) above 1 kHz [22]. This nonlinear frequency scale called the Bark or mel scale, and most modern ASR systems warp spectra so, as a partial auditory model, which seems to raise ASR accuracy [35]. Both production and perception experiments on speech have demonstrated the greater importance of the lower portion of the audio spectrum, where JNDs are smaller. Experiments with band-passed filtered speech show that basic comprehension is feasible with as little as the range of 300–3300 Hz (e.g., the telephone bandwidth), which preserves the first three resonances of the VT.

Thus, N in the DFT imposes a uniform choice of spectral resolution despite the non-uniform importance for speech. Some applications may vary N to fit the situation, e.g., a low N for cases such as obstruents, which tolerate lower frequency resolution, and a high value for vowels, where increased spectral resolution can be more fruitful. As, a priori, an ASR system does not know which sound is being uttered, few practical systems make such adjustments, preferring instead to simply choose a fixed analysis.

While very useful in transforming speech into a representation more readily exploited for ASR, the DFT accomplishes little data reduction. The data rate (bits/s) for speech spectra may even exceed that for time signals. Thus, ASR needs to compress the speech information further. To better understand which transformation may be most beneficial, we now relate the nature of speech to common data compression methods. Perhaps the simplest way is via bandpass filtering, or ``sub-bands.''

For ASR, it is useful to divide the spectrum into individual resonances, e.g., one sub-band for each formant, as then we could analyze more closely such resonance details as center frequency, bandwidth, harmonics, and phase. Few, if any, ASR systems do so, however, as reliable automatic estimation of formants has been a formidable problem. Instead, ASR has typically employed simpler, fixed bandwidths in sub-band analysis. Along the lines of sub-band and channel vocoders (used in speech coding), about 8–16 fixed, but not necessarily uniformly spaced, bands have been used as an approximation to the DFT. Following the mel scale, it is common to use wider bandwidths as frequency increases, corresponding to the ear's decreasing sensitivity. The fidelity of such a representation is not great, however, as it changes significantly when formants cross band borders. ASR representations should emulate human perception more closely.

A major difference between coding and recognition is that the former requires speech re-synthesis (for human listening) from the reduced representation (after analysis), while the latter only needs to estimate the corresponding VT shape. So, ASR allows much more speech distortion in the data reduction process, as long as parameters relevant for phoneme discrimination are preserved. LPC assumes a specific model for speech, that of an all-pole spectrum with N poles. N is typically 10–16, but typically varies in proportion to the speech bandwidth: two poles for each kHz (to model each formant) plus another 2–4 for general spectral shaping (corresponding to poles and zeros from the VT excitation and the mouth radiation effect). The objective is to approximate the spectral envelope of the DFT, while greatly reducing the number of parameters. The LPC N-pole model compresses by about two orders of magnitude, effectively smoothing the DFT, which often has harmonic detail [22]. Efficient algorithms locate the N poles near values corresponding to the natural frequencies of the VT, via a minimization of a mean-square error criterion applied to the speech spectrum. This criterion focuses attention on the spectral envelope, rather than on less relevant detail. LPC does not explicitly reveal formant frequency estimates, but instead parameters that can specify multiplier coefficients for re-synthesis filters and can produce a smooth spectral envelope estimate.

5.2. Mel-frequency cepstral coefficients (MFCCs)

The most common analysis method for ASR is the MFCC approach [42]. An FFT or LPC spectrum is obtained for each speech frame, for which the logarithm is then taken of the spectral amplitude (converting to decibels, and discarding the spectral phase), a set of triangular filters spaced according to the perceptual mel scale weights this result, and finally an inverse FFT is done. The low-order coefficients (e.g., 10–16 in number) of this last step provide the spectral vector for evaluation. This approach needs no difficult decisions to determine features (e.g., formant or F0 estimation, which risks error). ASR results appear to be better than with other methods (e.g., simple LPC, or filter bank), and one may interpret the MFCCs as roughly uncorrelated (since the inverse FFT uses orthogonal sinusoidal basis functions).

Despite their widespread use, MFCCs are suboptimal. Their major flaw lies in their final calculation step, the inverse FFT; taking low-order cosine weightings of the log spectrum is motivated entirely on mathematical grounds unrelated to speech communication. The first MFCC (C0) is simply a version of energy (i.e., weighting with a zero-frequency cosine), and C1 has a reasonable interpretation as indicating the global energy balance between low and high frequencies (the low range positively weighted by the first half of the single cosine period, and vice versa for the second half). However, other MFCCs are difficult to relate to any clear aspects of speech production or perception. They contain finer spectral detail, which altogether allow discrimination between similar sounds, but their lack of interpretation leave them highly vulnerable to non-ideal conditions such as noise or accents. In particular, the MFCCs give equal weight to high and low amplitudes in the log spectrum, despite the well known fact that high energy dominates perception. Thus, when speech is corrupted by noise, which often fills the spectral valleys between harmonics and between formants, the MFCCs deteriorate.

The spectral precision of MFCCs is directly related to their number, e.g., for a speech bandwidth of 4 kHz (e.g., for telephone applications) and 10 coefficients, the last MFCC uses a cosine weighting function with a period of 400 Hz, thus discriminating no better than 200 Hz (using more than 10 coefficients raises precision, but at increased cost). (This analysis ignores the mel scale, which deforms the frequency axis.) JND experiments on formants have suggested better human perceptual precision. The MFCCs are purported to be uncorrelated, due to the orthogonal functions of the inverse FFT, but they clearly contain overlapping information, which makes the covariance matrices of their joint probability densities far from diagonal.

When different speakers present varied spectral patterns for the same phoneme, the lack of interpretability of the MFCCs forces one to use simple merging of distributions to handle different speakers. Such merging leads to larger variances and hence lowered ability to discriminate against other phoneme models. A related approach, called perceptual linear prediction [43], employs a nonlinearly compressed power spectrum, and is found in some ASR systems.

5.3. Methods of comparison for ASR

At the heart of ASR lies the measurement of similarity between two localized (windowed) speech patterns, i.e., the representation of a frame of the input speech and one from a set of reference patterns or models (obtained during training). A memory of reference models, each characterized by an N-dimensional feature vector, is established during training in which speakers usually utter a controlled vocabulary, and acoustic segments are parametrized and automatically labeled with phonetic codes corresponding to the training texts. For units of short speech frames, the codes involve phonetic segments (e.g., phonemes). If we generalize to longer speech units (syllables, words, or phrases), we can expand N to include multiple frames and thus time variation in the parameters. For word-based ASR, templates could be M-dimensional vectors or $L \times N$ matrices ($M = LN$), where the L vectors of dimension N are extracted at uniformly spaced intervals for each word in the system vocabulary.

The similarity between two patterns can be expressed via a distance or distortion. (A correlation between patterns may replace this distance measure.) To handle multi-frame utterances (i.e., all practical cases), local (frame) distance measures typically sum to yield a global (utterance) distance, or probabilities are multiplied to yield a joint likelihood (assuming limited conditional independence between frames).

An immediate issue that arises is how to represent all the training data in the models to be used for comparison. In some PR applications, each evaluation standard may be quite precise, e.g., in quality control for manufactured goods, a designer might specify a pattern that is considered to be ideal, and all products would be compared against this ``standard''. In ASR, on the other hand, the models are determined from large sets of exemplars, i.e., speech from many human talkers, as ASR must handle many sources of variability in language (speakers, contexts, channels, etc.) as well as a lack of standards (i.e., a diversity of speech representations for linguistic speech units).

Suppose that we reduce the general ASR problem (e.g., transcribing a complete utterance into a sentence) to a simpler task that nonetheless retains many elements of general ASR: selecting which phoneme has been uttered by a given, known speaker. (We treat the important and complicating issues of coarticulation and different speakers elsewhere.) Let us further assume that the phoneme is in steady state (e.g., a simple vowel or fricative) that can thus be well modeled by one frame of data, which is transformed to a set of N parameters via data reduction (e.g., N is approximately 10–16 for MFCC). In such circumstances, one can assume that: (1) the uttered test phone is well modeled by a point in N-dimensional space and (2) one can obtain a set of suitable exemplars for all phonemes from this speaker, who would utter all phonemes of his language many times. (Such training, involving only minutes, is eminently feasible for SD ASR.)

There are accepted procedures to select a reasonably small portion of the exemplars for each phoneme for use as templates in the test ASR phase. It is usually better to have the speaker utter many examples and then prune away ones that are redundant, rather than work from a smaller set. Pruning is needed, as it is wasteful to repeatedly compare a test pattern against many similar training patterns for the same phoneme. On the other hand, if the speaker only initially furnishes a few examples of each phoneme, the likelihood increases that he may later utter that phoneme differently during the test phase. Thus, major problems for ASR are to determine how many templates are needed, and how to prune large training sets. This issue of variability is compounded when one includes other sources of variability in speech.

Continuing with our simple situation of one speaker and one phoneme, we have for each phoneme a set of typical points in N-dimensional space. Using the same data representation procedure (as in training) on the test frame of speech data, we get a single point in this space. We can then use a distance measure to locate the training point closest to the test point, and thus label the test phone with the identity of the closest training point. However, the reliability of the training templates is an issue. If the feature space (i.e., data reduction) is well chosen, each phoneme's exemplars might group tightly in the space and be sufficiently distant from all other phonemes' templates. In actual ASR, however, there are virtually always significant amounts of overlap. Thus basing one's ASR estimation only on the closest training point to a test point is sub-optimal. The k-NN (k-nearest neighbor) approach recognizes this risk, and suggests labeling a test frame with the phoneme that has $k > 1$ exemplars close to the test point; typically, k is a small number, thus accommodating cases where each phoneme might have a few ``outlier'' exemplars (uttered examples which are rare in one's speech). If such an outlier point is surrounded by more numerous exemplars of another phoneme, ASR using k-NN would not choose the outlier label even if the outlier happened to be the closest match [22].

5.4. Statistical measures

This issue of how to handle sets of exemplars is often settled by using a PDF, using all training data as a histogram. As more training data are available (and a common refrain in the ASR field is that one never has enough data), the PDF becomes more reliable. One need not prune at all, retaining all exemplars as equal contributors to the PDF. If all training frames have equal weight, one could retain full PDFs, and choose the output label based on which PDF yielded the highest likelihood for a training frame. In practice, to save computation and memory, we tend to model the training PDFs with simpler parametric PDFs, e.g., Gaussians.

Employing full PDFs has two drawbacks: (1) increased memory to store all their data and (2) inaccurate modeling. Often, we characterize each dimension with a resolution corresponding to the degree with which patterns can vary (and/or that humans can discriminate). This can mean hundreds of values along each dimension, and thus PDFs with millions of parameters. ASR memory is thus not efficiently utilized, although search can be quite fast. A more serious problem is that full PDFs rarely model natural variability in patterns well, unless one has access to huge amounts of training data. It is reasonable to assume that ultimate (or ``true'' target) PDFs would be relatively smooth, if one trained on an infinite supply of data. In other words, if A and B are both good exemplars of a class, normally all points intervening between A and B would be as well. If one has chosen the feature space well, class PDFs should be both smooth and with only one mode, e.g., Gaussian. Indeed, the Gaussian density is a common assumption for many PR classes. Practical PR cases are not Gaussian, as each dimension has a finite range, but their smooth curve and simple parameter models (mean and variance) have led to their widespread adoption.

Actual PDFs in most ASR are not smooth, which has led to the widespread use of ``mixtures'' of PDFs to model speech units. Such Gaussian mixture models (GMMs) are typically described by a set of simple Gaussian curves (each characterized by a mean vector—N values for an N-dimensional PDF—and an N-by-N covariance matrix), as well as a set of weights (summing to one) for the contributions of each Gaussian to the overall PDF.

There are many reasons why a class's PDF might not have a single mode, e.g., if a phoneme were to encompass different distinct acoustic realizations. In some languages, e.g., Spanish and French, the phoneme /r/ can be acceptably produced in diverse ways, e.g., trills or uvular fricatives. Similarly, each language typically has several dialects, and what is labeled as a specific phoneme may occur acoustically with several variants. Some ASR systems thus extend the basic phoneme inventory (25–40 phonemes, depending on the language) to include different common ``allophone'' variants. (It is often easier to extend the set of classes to be recognized, as one can later collapse such extended sets readily to the standard phoneme set when accessing the language's dictionary.)

A reduction of the feature set to a minimal set of orthogonal parameters would greatly simplify ASR procedures, but this is usually always rejected owing to expense. To account for all the interaction (inter-dependence) of feature dimensions, one must normally include full covariance matrices in PDFs, showing explicitly the correlation between each pair of parameters in the N-dimensional space. However, evaluating each such Gaussian PDF takes about N^2 mathematical operations (versus N operations for simpler diagonal matrices). Hence, ASR often makes an assumption that parameters are uncorrelated, which then allows use of the simpler matrices, despite clear evidence that the dimensions are virtually always correlated. The net effect of employing such assumptions (usually based on cost, and lacking scientific basis) on ASR accuracy is difficult to estimate; however, they usually lead to poorer, but faster, recognition. As diagonal covariance matrices greatly oversimplify reality, there are recent compromises that constrain the inverse covariance matrices (as used in evaluating Gaussian PDFs) in various ways: so-called semitied and subspace-constrained GMMs [36]. While true orthogonalization (e.g., KLT, or principal components analysis) is rarely applied in ASR, various approximations have been tried, e.g., heteroscedastic linear discriminant analysis (HLDA).

5.5. Comparison measures

Owing to complex correlations among LPC parameters (unless they are partly de-correlated, say, into formant parameters), ASR using LPC has been done with distance measures. The simplest such measure is the Euclidean distance, but that assumes that each dimension has equal variance and importance, and is independent of each other. Unless speech parameters are pre-normalized, e.g., with orthogonalization, this assumption does not hold. Even parameters that are simple to interpret (although often difficult to reliably extract), such as the formants, rarely satisfy these conditions. (As one example, the range of values for F2 is about three times that of F1; hence treating the F1 and F2 dimensions the same would unjustifiably distort their perceptual relevance.) The variance issue is readily handled by properly weighting each dimension inversely as its standard deviation, but this does not deal with issues of parameter interdependence.

While a weighted Euclidean distance is still sometimes used in ASR, a more appropriate distance measure was found for LPC. The basic LPC parameters specify the impulse response of an ``inverse'' digital filter that transforms an analyzed speech frame into its ``residual'' excitation, i.e., the smallest energy excitation that, when exciting the synthesis filter, produces the original speech. Thus, LPC ASR effectively passed each unknown speech frame through a set of candidate inverse filters, choosing the one with the lowest energy output. When the candidate inverse filter corresponded to a sound similar to that of the input speech, the resulting match would lead to a minimum in residual energy. Several other distance measures have been used in fields related to ASR, such as Battacharya in speaker verification [44].

We often represent a word (or other appropriate speech unit) as a point in a parameter (or feature) space of N dimensions. The many possible pronunciations of that word describe a multivariate PDF in this space. If we assume ASR among equally likely words and use ML as the decision criterion, Bayes' rule specifies that we choose the word whose PDF is most likely to match the test utterance. The Gaussian PDF of an N-dimensional parameter vector X for, say, word i is

$$P_i(X) = (2\pi)^{-N/2}|\mathbf{W}_i|^{-1/2} \times \exp\left[-\frac{(X-\mu_i)^{\mathrm{T}}\mathbf{W}_i^{-1}(X-\mu_i)}{2}\right],$$

where $|\mathbf{W}_i|$ is the determinant of $\mathbf{W}_i$ and μ_i is the mean vector for word i. Most ASR systems use a fixed $\mathbf{W}$ matrix (vs. a different $\mathbf{W}_i$ for each word) because (a) it is difficult to obtain accurate estimates for many $\mathbf{W}_i$ from limited training data, (b) using one $\mathbf{W}$ matrix saves memory, and (c) $\mathbf{W}_i$ matrices are often similar for different words. Given a test vector X for recognition, word j is selected if

$$P_j(X) \geqslant P_i(X) \quad \text{for all words } i \text{ in the vocabulary.}$$

Applying a (monotonic) logarithmic transformation and eliminating terms that are constant across words (e.g., a common $|\mathbf{W}|$), these equations can reduce to minimizing the Mahalanobis distance.

The use of logarithms of probabilities minimizes computation, as the likelihood to be maximized is a joint probability of all the speech frames in an utterance. With the standard HMM method, the joint probability is a product: $P(A)P(B|A)P(C|A,B)P(D|A,B,C)\ldots P(Z|A,B,\ldots Y)$, where $A\ldots Z$ represent successive frames in order. The first-order assumption means we assume that we can ignore conditions beyond one frame in the past: $P(A)P(B|A)P(C|B)P(D|C)\ldots P(Z|Y)$. This still leaves a large amount of multiplications to evaluate each model. Taking logarithms converts the product into a sum, and since log is monotonic, maximization yields the same result, with less computation.

Ideally, repetitions of the same speech segment would yield consistent parameter values and therefore small ``clusters'' or regions in the feature space. Different speech segments should provide widely separated points in the space. If chosen well, the speech parameters would show little intra-segment variance and large inter-segment variances. Ideally, they also would be independent of each other and with equal importance. In practice, however, ASR parameters always share speech information, with some being much more relevant than others. Not accounting for such interdependence and unequal importance lowers ASR accuracy. The model above is often extended to include a term for pronunciation modeling, to account for speaker dialect, accent, and other speaker freedoms, as words are often uttered with a varied choice of phonemes [45].

6. Timing issues in ASR

The complexity of the search task in ASR increases with utterance length. As a simplification, older systems required speakers to modify their speech, e.g., pausing after each word. This illustrates one of the major difficulties of ASR—segmentation. It is very hard to segment speech reliably into useful smaller units, e.g., phonemes or words. Sudden large changes in speech spectrum or amplitude may help to estimate unit boundaries, but these cues are unreliable. Syllable units can often be located approximately via intensity changes, but exact boundary positions are elusive in languages that allow successive vowels or consonants (e.g., English). In languages having polysyllabic words, word boundaries are even harder to locate than phone or syllable boundaries. Segmenting utterances into smaller units for ASR simplifies computation and often aids accuracy by reducing the search space, but only if the partitioning is correct. Pauses play an important role in segmenting speech, but silences may correspond to stop closures, and speakers normally pause only after several words (and sometimes pause, or hesitate, within words).

6.1. Linear and dynamic time interpolation

When comparing individual sounds, a single representation (even just one frame) may suffice. In virtually all ASR, however, vocabulary entries involve sequences of acoustic events, even in simple cases (e.g., vocabularies of the digits or the letters A–Z). ASR typically compares parametric templates (or evaluates probabilities) frame-by-frame, which leads to alignment problems. Utterances are rarely spoken at a single, uniform speaking rate; thus, test and reference utterances generally have different durations (i.e., unequal numbers of frames). A suboptimal way to do frame-by-frame comparison is to normalize frame lengths so that all templates share a common number of frames, i.e., linear time normalization. Accurate time alignment of templates is crucial for ASR accuracy. Linear warping is rarely sufficient to align all speech events properly because the effects of speaking-rate change are nonlinear; e.g., vowels and stressed syllables tend to expand and contract more than consonants and unstressed syllables.

ASR must also address what may seem to be a trivial issue, that of determining exactly when speech starts and stops: speech versus silence (or background noise). Such voice activity detection itself may be viewed as ASR with a binary output: speech or not [22]. The background is rarely completely silent. Noise may come from speakers (lip smacks, breathing, mouth clicks), the environment (stationary: machines, traffic, weather; non-stationary: music, moving objects), and/or transmission (channel noise, crosstalk). Most VAD relies on signal energy to separate non-speech from speech, often augmented by some simple spectral measures; e.g., zero-crossing rate (ZCR) can provide a basic estimate of the frequency of major energy concentration. Background noise often has a broad low-pass spectrum, with a resultant ZCR of medium value. For speech obstruents, which are the most confused with silence, the ZCR is either high (corresponding to the high-frequency concentration of energy in fricatives and stop bursts) or very low (if a voice bar dominates).

6.2. Hidden Markov models (HMMs)

Since the mid-1980s, ASR has been dominated by a network approach for handling the step of comparing two speech patterns [46]. Knowledge about syntactic and semantic constraints on allowable sequences of words, as well as stochastic modeling of spectral and timing patterns of speech, has all been efficiently coded in terms of a network whose states may correspond to the words of a given vocabulary or to VT spectra. Transitions among states of an ASR network are allowed only if the resulting string of words yields a legal sentence following the system's grammar. ASR networks employ a statistical, rather than rule-based, representation of acoustic and linguistic information. For word-based ASR, one may have a network to model each word in a given vocabulary (e.g., the digits) with a succession of phonetic states i to j (each corresponding roughly to a part of a phoneme), linked by transitions specified by likelihoods a_{ij}.

The most common network for ASR is the first-order Markov process or chain, where the likelihood of being in a given state depends only on the immediately prior state [47–49]. As phonemes have a clear temporal order in words, only left-to-right transitions are allowed. Thus the model states are ordered, with initial, middle and final states, respectively, corresponding roughly to the beginning, middle and end of an utterance unit being modeled. The networks are often called hidden Markov models (HMMs) [50] because the models must be inferred from observations of speech outputs, rather than any direct observation of the VT. States correspond roughly to acoustic events, but there is no explicit alignment in most applications. One normally chooses the number of states for an HMM to allow approximately one state for each distinct acoustic segment in the speech unit (e.g., word or phoneme) being modeled. A phoneme HMM might use three states to represent, in order, an initial transition spectrum from a prior phoneme, a spectrum from the phone's presumed steady-state, and a final transition spectrum to an ensuing phoneme. To account for variability (from coarticulation, speaking rate, different speakers, etc.), each state is represented by a PDF of spectra rather than one spectrum. A widely used system for developing HMMs is HTK [2].

(1) *Training Markov models*: The parameters in each HMM are trained via an initial estimation and then an iterative re-estimation. A standard method is the expectation–maximization (E–M) algorithm [51], a procedure similar to that of designing vector quantization codebooks. This uses a gradient or hill-climbing method, which yields a locally optimum model. The initial prototype acoustic vector for each state (however obtained) evolves iteratively to a full PDF, through an averaging or clustering procedure involving the rest of the training data. Typically the initial vector provides the first estimate of the mean of the PDF. The most common averaging procedures are variations of the Viterbi algorithm [52–54] (e.g., the gradient method), the *Baum–Welch algorithm* [55], and the forward–backward (F–B) algorithm [56].

Classification of HMMs via the simpler Viterbi search is much faster than the F–B method because it locates only the best path in the HMM, rather than calculating the sum of probabilities of all paths. When log probabilities are used, no multiplications are needed. When used in training, the Viterbi method also provides a segmentation of the utterance (via backtracking to know the states along the optimal path), specifying which frames correspond to each of the states of the HMM. There is no explicit segmentation in the F–B method since all paths are examined and contribute equally to the outcome.

ASR often prunes its search space for efficiency, at the risk of possibly missing the right answer. Beam searches (constraint maintaining multiple hypotheses in parallel), stack decoding, and rescoring of N-best word lattices are among many ways to make the expensive ASR search more efficient. A depth-first search can replace the basic breadth-first Viterbi approach in ``stack decoders.'' With recent weighted finite-state transducers, all model information (acoustic and language) can be integrated into a large, optimized network [57]. An N-best (a few hundred) set of hypotheses is often used with multiple recognition passes. A compact and efficient structure for this is the word lattice—a set of nodes for points in time with spanning arcs (having acoustic and language model (LM) scores) for word hypotheses.

(2) *Sub-word models*: There is a crucial question of which speech unit size is best to use for ASR. For applications with small vocabularies (e.g., ``yes''–``no,'' 0–9, A–Z), words provide the units, as memory and search are very feasible. In less restricted cases, where extremely large numbers of different long utterances are permissible, use of long units of the size of words and larger becomes much less feasible. This leads to units smaller than the word, and yet bigger than the phoneme, so as to handle coarticulation effects well. There is clearly more vocabulary flexibility and smaller memory requirements with smaller stored speech units.

Use of sub-word units for models reduces ASR accuracy [58]. The most common choices are phone, biphone and triphone models, that each pose models for individual phonemes, but in the context of 0, 1 or 2 immediate phoneme neighbors. The difficulties lie in that coarticulation and stress effects extend well beyond immediately adjacent phonemes, yet such CD models can easily number in the many thousands, and thus be less well trained if training data is limited.

One important way to reduce such under-training is to share or ``tie'' parameters across models, using the same values in all models pertinent to a given context. Having separate triphone CD models for all sequences of three phones is inefficient, since many phoneme contexts have similar coarticulatory effects; e.g., the effects of labial /f,v,p,b,m/ on an adjacent vowel are very similar. Tied models share the same parameter values, so as to reduce the total number of parameters to train. Tying can be automatic (e.g., with unsupervised, data-driven decision trees [59]), or guided by linguistic properties (e.g., grouping contexts with labels such as labial, velar, fricative, etc.).

(3) *Comparing HMMs and DTW*: As in DTW, HMMs use a form of dynamic programming to find the best match between a test utterance and different recognition models. Viterbi HMM determines the sequence of states that maximizes probability, while DTW finds a warp path to minimize an accumulated distance. Unlike most DTW methods, no multiplications are required for Viterbi HMM. A big advantage of HMM over DTW is reduced calculation in the test phase. DTW represents temporal and speaker variability simply by storing many templates. HMMs, on the other hand, incorporate more structure than DTW and capture much speaking variability in their models during the off-line training phase. The test phase is relatively rapid, needing no expensive distance calculations and only summing path probabilities for a network whose size is usually less than the number of frames in templates. HMMs are more flexible than DTW in modeling temporal and spectral variability.

(4) *Modeling durations in HMMs*: Modeling successive frames of speech, which are often acoustically similar, in a single HMM state decreases memory and computation, but yields an inaccurate model of temporal information. Durational information for a first-order HMM lies in the probability of a loop transition a_{ii} for state i. The likelihood of remaining in state i for n frames is $a_{ii}^{n-1}(1-a_{ii})$. This geometric (exponential) distribution is a poor model for speech durations, which follow Poisson distributions more closely. Extending the basic HMM to allow non-geometric models for duration [60] can raise ASR accuracy, but only at the cost of added complexity [61]. Instead, most systems concentrate on modeling the state probabilities, which, in practice, dominate ASR decisions [62].

(5) *Improving HMMs*: A major difficulty with HMMs is the frame-independence assumption, from the use of first-order Markov models. ASR usually processes speech in frames of 10 ms, and assumes independence of successive HMM states. Use of ``Delta'' and ``Delta–delta'' coefficients to include timing information over several frames (e.g., 50 ms) is helpful but inefficient. Typically, one uses a 39-dimensional MFCC vector: 12 static MFFCs plus energy, and

1st and 2nd derivatives to model, in limited fashion, VT velocity and acceleration, in an attempt to look beyond one frame. However, this is inefficient, owing to the need to process many parameters.

Future ASR must exploit timing better. Among the attempts to model speech better include segment models, stochastic trajectory models [63], trended HMMs [64] and linear trajectory HMMs [65]. Generalizing the basic HMM to allow Markov models of order higher than one raises ASR accuracy (by exploiting restrictions on how speech frames occur in sequence); however, the computational complexity of such models has greatly hindered their application in ASR. Minimization of computation and memory, while not always critical for all ASR, is important for portable applications [66].

The basic HMM is a generative model, which has been generalized recently into an alternative graphical representation called dynamic Bayesian networks (DBNs), which emphasize conditional dependencies in the model.

7. Language models (LMs)

Prior to the 1980s, ASR only used acoustic information to evaluate text hypotheses. It was then noted that incorporating knowledge about the text being spoken (exploiting textual redundancies) would significantly raise ASR accuracy. Speech usually follows linguistic rules (e.g., syntax and semantics). Sometimes speech is merely a random sequence of words drawn from a very limited vocabulary (e.g., digits in a telephone number); such cases have no textual redundancies. Normally, given a history of prior words in an utterance, the number P of words that one must consider as possibly coming next is much smaller than the vocabulary size V. P is called the perplexity of a LM. LMs are stochastic descriptions of text-likelihoods of local sequences of N consecutive words in training texts (typically $N = 1, 2, 3$). Integrating a LM with the normal acoustic HMMs is now common in ASR.

ASR may output a sequence of symbols representing phonemes, with associated likelihoods, e.g., a weighted list of boundary locations and phoneme candidates, often in the form of a lattice [29]. A postprocessor or ``rescoring'' algorithm may apply a LM to this lattice. A LM may take the form of a traditional grammar, i.e., syntactic rules through which sentences are parsed into component words and phrases [67]. However, natural language (e.g., unrestricted English) can be very complex.

Typically, N-gram models estimate the likelihood of each word, given the context of the preceding $N-1$ words, e.g., bigram models use statistics of word pairs and trigrams model word triplets. Unigrams are simply prior likelihoods for each word, independent of context. These probabilities are obtained through analysis of much text, and capture both syntactic and semantic redundancies in text.

As vocabulary (V words) increases for practical ASR, the size of a LM (V^N) grows exponentially with V. Large lexicons lead to seriously under-trained LMs, inadequate appropriate texts for training, increased memory needs, and lack of enough computation power to search all textual possibilities. As a result, most ASR has employed only unigram, bigram and trigram statistics. Back-off methods fall back on lower-order statistics when higher-order N-grams do not occur in the training texts [68]. Grammar constraints are often imposed on LMs, and LMs may be refined in terms of parts-of-speech classes. LMs can be designed for specific tasks.

An important subfield of ASR concerns determining whether speakers have said something beyond an acceptable vocabulary, i.e., out-of-vocabulary detection [69]. ASR generally searches a dictionary to estimate, for each section of an audio signal, which word forms the best match. One does not wish to output (incorrect) words from an official dictionary when a speaker has coughed or said something beyond the accepted range [70]. Yet it is practically important to detect such OOV conditions.

8. ASR evaluation

Several databases have become de facto standards for the development and testing of ASR algorithms (Table 1), and are widely available through the Linguistics Data Consortium and the European Language Resources Association. Many have been used in various competitions organized by NIST [71]. Typically, data are made available to all research groups willing to commit to competing and presenting their results at appropriate meetings. Algorithms are developed on a large set of training data, then refined on a separate set of development data, and finally tested on another set. As always, it is critical in ASR to employ diverse sets, involving different speakers and channel conditions, so as to avoid tuning one's models too closely to one set of conditions. In most cases, success is defined in terms of WER, although some recent applications have goals more difficult to define, e.g., making summaries (gists) of conversations [72]. Other related applications include language identification and speaker verification [73]. In verification cases, two types of error exist: false acceptance and false rejection. By adjusting thresholds in their algorithms, one can offset the two errors, as a tight threshold leads to a high rejection rate for subscribers and a loose one yields accepting too many impostors. While a balanced, ``equal error rate'' (setting the threshold so that the two errors are equal) simplifies evaluation, one usually examines a wide range of balanced outcomes, as costs vary greatly across applications (e.g., low rates of false acceptance when security must be high; low rates of false rejection for public applications where cost of error is small).

9. Conclusion

One may well ask whether adequate ASR will ever truly be accomplished. We believe that it can. In general, one may assume that almost all artificial intelligence (AI) tasks are potentially feasible; certainly great progress in chess-playing machines and robotics supports this view. Compare ASR to the task of automatically driving a car; the latter requires intelligent interpretation of the field of vision for cameras mounted on a vehicle. While algorithms needed for cars would be very different than for ASR, there are similarities in signal processing and both challenges seem daunting (i.e., replacing a human driver with a similar-performing algorithm might seem as far-fetched as having a fully understanding ASR device). It would seem that ASR is much closer to potential solution, however. Unlike the vast diversity (and significance) of what a car camera may see in a rapidly evolving 3-dimensional signal (images at 100 km/h), speech is completely captured (at least for purposes of intelligibility—as needed for ASR) at rates of 10 kbits/s (e.g., as in cell phones). Crude representations such as wide-band spectrograms have been shown to allow highly accurate speech-to-text translation by human experts, which strongly suggests that one should be able to formulate intelligent ways to process speech at the relatively slow data rates of spectrograms [74]. The diversity of environments for speech (i.e., channel conditions, speakers, conversation contexts, etc.), while admittedly large, pales in comparison to that possible in driving vision. Adequate ASR may not be so evident with today's relatively simple approaches, but a more focused approach, integrating knowledge about human speech production and perception should allow much higher accuracy in the near-to-medium future.

The gap between human and machine recognition of speech remains large for many practical tasks [31]. Current ASR accuracy is adequate for small vocabularies; humans, on the other hand, are highly capable in difficult conditions, e.g., understanding unknown speakers in noisy environments saying arbitrary utterances. ASR functions well in ``matched conditions,'' where the system has been previously trained on all: (1) speakers who would test the system, (2) words that may be used, and (3) possible recording conditions. The challenge remains for ASR to increase accuracy for mismatched conditions: the WER for spontaneous speech remains as high as 25% in many cases, while commercially acceptable WERs are often under 2%. The market for ASR is growing, but its rate of growth will increase greatly only when performance approaches that of humans.

The simplicity of models used in current ASR, when compared to the complex, nonlinear processing done by humans, suggests that there remain many ways to improve ASR. At each level (i.e., feature estimation, temporal and acoustic modeling, language modeling, search, decision making), compromises have been made in ASR to have simple and fast processing, at the expense of lower accuracy. We believe that ASR can be done at high accuracy within the constraints of computation available in current mobile devices.

Recently, ASR has been almost entirely data-driven, with statistical models purported to accommodate all variability. For example, the HMM structure, where spectral variability (e.g., owing to different VT sizes across different speakers) is handled via Gaussian mixtures and temporal variability (aspects of different speaking rates) is handled via transition likelihoods, may appear to have the flexibility to accommodate much variation observed in speech. Yet it has serious weaknesses, most notably the assumption of a first-order model, where temporal dependence is limited to extremely short window lengths (10-ms frames). What is needed is a hybrid approach to ASR, one that combines both structure and statistics. Researchers must strive to impose more structure in the algorithms, based on knowledge of human speech production and perception.

For example, the most common acoustic models are triphone context-dependent HMMs. As many applications attempt to accommodate any user, we often have SI models trained on hundreds of speakers. The result is often PDFs that are too broad to discriminate among similar phonemes, despite the use of many hundreds of models to accommodate various phonetic contexts. A common failure of this approach is that it does not exploit the fact that, in a given utterance, a single speaker is talking; SD models have more narrow (and hence more discriminating) PDFs. In addition, the basic HMM has an inherent geometric PDF for state durations, yet we know that typical phonemes last 80 ms (8 frames). Further, coarticulation in human speech production causes spectral carryover effects over many frames; yet the first-order HMM assumes independence beyond the immediately prior frame.

Thus, the HMM has been both a boon and a bane for ASR: it has allowed practical progress in use of ASR, yet has tended to hinder further advances; its model simplicity has resisted many modifications to allow lowering ASR error rates, yet no clear alternative has appeared. Similar comments can be made of the MFCC approach to data compression. We have long realized that the critical information (for ASR) resides in the spectral envelope (formant structure) of the input speech signal. Attempts to reliably estimate the center frequencies and bandwidths of the resonances of the VT (i.e., formants) are still seen today, but no good formant estimator (nor pitch period estimator) has been perfected. PR that relies on a succession of error-prone modular steps is often doomed to poor accuracy (indeed, a strength of the HMM is its consideration of all data before making any decisions). Thus, the weakness of attempting formant estimation at any early stage of ASR, without any clear method for feedback correction of errors, has made ASR designers look to simple spectral estimation.

What we need are ways to better integrate structure and focus into the ASR process, in addition to ways to handle ever increasing amounts of training data. The ASR field has followed the adage of ``there's no data like more data'' in incorporating many hundreds of hours of speech data and millions of words of text in ASR training. Yet the ability to properly integrate all this data into useful models is in question. Indeed, we have found that, when employing more amounts of training texts (e.g., from the vast WorldWide Web), while improving the accuracy of the models (e.g., alleviating the sparseness issue, seen when training on texts of inadequate size), the improved LM (language model) is often too general for most applications. In human-to-human speech, people effectively employ much smaller LMs in their minds, models that are localized to specific topics of conversation. They retain many versions of LMs in various degrees of generality to efficiently handle the great degree of diversity of communication environments; further, they shift seamlessly among these models as the situation warrants.

ASR models have often been over-trained on limited data, such that the model parameters fit too well the training data, and hence generalize poorly to other cases. Merely using more data does not solve this problem, as the models rarely focus on critical, discriminatory aspects of speech. In general, ASR has exploited the global HMM structure that examines all the speech and textual data before making a decision; this avoids premature errors (as in earlier expert system ASR), but allows the weaknesses of the approach to remain more obscure. Incremental advances to such a system have been made over the last several years, but its overall complexity makes it difficult to identify where key weaknesses lie. It is difficult to look at the errors that HMM ASR makes, and decide which models or approaches are at fault. As a result, researchers have tended to tinker with minor details, with the result being gradual, incremental improvements. Error rates for databases of conversational speech (e.g., Switchboard, CallHome) remain unacceptably high. For practical applications, errors must be reduced further.

References

[1] D. Ververidis, C. Kotropoulos, Emotional speech recognition: resources, features, and methods, Speech Commun. 48 (2006) 1162–1181.

[2] T. Hain, P. Woodland, G. Evermann, M. Gales, X. Liu, G. Moore, D. Povey, L. Wang, Automatic transcription of conversational telephone speech, IEEE Trans. Speech Audio Process. 14 (2005) 1173–1185.

[3] M. Benzeghiba, R. De Mori, O. Deroo, S. Dupont, T. Erbes, D. Jouvet, L. Fissore, P. Laface, A. Mertins, C. Ris, R. Rose, V. Tyagi, C. Wellekens, Automatic speech recognition and speech variability: a review, Speech Commun. 49 (2007) 763–786.

[4] X. Huang, K-F. Lee, On speaker-independent, speaker-dependent, and speaker-adaptive speech recognition, IEEE Trans. Speech Audio Process. 1 (1993) 150–157.

[5] Y. Zhao, An acoustic–phonetic-based speaker-adaptation technique for improving speaker-independent continuous speech recognition, IEEE Trans. Speech Audio Process. 2 (1994) 380–394.

[6] J. Kreiman, Speaker modeling for speaker adaptation in automatic speech recognition, in: K. Johnson, J. Mullennix (Eds.), Talker Variability in Speech Processing, Academic Press, San Diego, 1997, pp. 167–189.

[7] J.-L. Gauvin, C.-H. Lee, Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains, IEEE Trans. Speech Audio Process. 2 (1994) 291–298.

[8] J.-T. Chien, C.-H. Lee, H.-C. Wang, A hybrid algorithm for speaker adaptation using MAP transformation and adaptation, IEEE Signal Process. Lett. 4 (1997) 167–169.

[9] C. Leggetter, P. Woodland, Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models, Comput. Speech Lang. 9 (1995) 171–185.

[10] M. Gales, Maximum likelihood linear transformations for HMM-based speech recognition, Comput. Speech Lang. 12 (1998) 75–98.

[11] R. Kuhn, et al., Eigenvoices for speaker adaptation, in: Proceedings of the International Conference on Spoken Language Processing, 1998, pp. 1771–1774.

[12] T. Claes, J. Dologlou, L. ten Bosch, D. van Compernolle, A novel feature transformation for vocal tract length normalization in automatic speech recognition, IEEE Trans. Speech Audio Process. 6 (1998) 549–557.

[13] C.-H. Lee, On stochastic feature and model compensation approaches for robust speech recognition, Speech Commun. 25 (1998) 29–47.

[14] R. Lippmann, B. Carlson, A robust speech recognition with time-varying filtering, interruptions, and noise, in: IEEE Workshop on Speech Recognition, 1997, pp. 365–372.

[15] H. Hermansky, N. Morgan, RASTA processing of speech, IEEE Trans. Speech Audio Process. 2 (1994) 578–589.

[16] M. Rahim, B.-H. Juang, W. Chou, E. Buhrke, Signal conditioning techniques for robust speech recognition, IEEE Signal Process. Lett. 3 (1996) 107–109.

[17] M. Gales, Predictive model-based compensation schemes for robust speech recognition, Speech Commun. 25 (1998) 49–74.

[18] M. De Wachter, M. Matton, K. Demuynck, P. Wambacq, R. Cools, D. Van Compernolle, Template-based continuous speech recognition, IEEE Trans. ASLP 15 (2007) 1377–1390.

[19] F. Itakura, Minimum prediction residual principle applied to speech recognition, IEEE Trans. ASSP 23 (1975) 67–72.

[20] H. Sakoe, S. Chiba, Dynamic programming algorithm optimization for spoken word recognition, IEEE Trans. ASSP 26 (1978) 43–49.
[21] H. Silverman, D. Morgan, The application of dynamic programming to connected speech segmentation, IEEE ASSP Mag. 7 (3) (1990) 7–25.
[22] D. O'Shaughnessy, Speech Communications, IEEE Press, New York, 2000.
[23] A. Stolcke, et al., Recent innovations in speech-to-text transcription at SRI-ICSI-UW, IEEE Trans. ASLP 14 (2006) 1729–1744.
[24] W. Campbell, J. Campbell, T. Gleason, D. Reynolds, W. Shen, Speaker verification using support vector machines and high-level features, IEEE Trans. ASLP 15 (2007) 2085–2094.
[25] R. Solera-Ureña, D. Martín-Iglesias, A. Gallardo-Antolín, C. Peláez-Moreno, F. Díaz-de-María, Robust ASR using support vector machines, Speech Commun. 49 (2007) 253–267.
[26] K. Shutte, J. Glass, Speech recognition with localized time-frequency detectors, in: IEEE Automatic Speech Recognition and Understanding Workshop, 2007, pp. 341–344.
[27] H. Jiang, X. Li, C. Liu, Large margin hidden Markov models for speech recognition, IEEE Trans. ASLP 14 (2006) 1584–1595.
[28] K. Yu, M. Gales, Bayesian adaptive inference and adaptive training, IEEE Trans. ASLP 15 (2007) 1932–1943.
[29] C. Ma, C.-H. Lee, A study on word detector design and knowledge-based pruning and resocring, in: Proceedings of the Interspeech, 2007, pp. 1473–1476.
[30] I. Bromberg, et al., Detection-based ASR in the automatic speech attribute transcription project, in: Proceedings of the Interspeech, 2007, pp. 1829–1832.
[31] B. Meyer, M. Wachter, T. Brand, B. Kollmeier, Phoneme confusions in human and automatic speech recognition, in: Proceedings of the Interspeech, 2007, pp. 1485–1488.
[32] K. Chen, M. Hasegawa-Johnson, A. Cohen, S. Borys, S.-S. Kim, J. Cole, J.-Y. Choi, Prosody dependent speech recognition on radio news corpus of American English, IEEE Trans. ASLP 14 (2006) 232–245.
[33] M. Rajamanohar, E. Fosler-Lussier, An evaluation of hierarchical articulatory feature detectors, in: IEEE Automatic Speech Recognition and Understanding Workshop, 2005, pp. 59–64.
[34] O. Scharenborg, V. Wan, R. Moore, Towards capturing fine phonetic variation in speech using articulatory features, Speech Commun. 49 (2007) 811–826.
[35] W. Jeon, B.-H. Juang, Speech analysis in a model of the central auditory system, IEEE Trans. ASLP 15 (2007) 1802–1817.
[36] S. Axelrod, V. Goel, R. Gopinath, P. Olsen, K. Visweswariah, Discriminative estimation of subspace constrained Gaussian mixture models for speech recognition, IEEE Trans. ASLP 15 (2007) 172–189.
[37] E. McDermott, T. Hazen, J. Le Roux, A. Nakamura, S. Katagiri, Discriminative training for large-vocabulary speech recognition using minimum classification error, IEEE Trans. ASLP 15 (2007) 203–223.
[38] C. White, J. Droppo, A. Acero, J. Odell, Maximum entropy confidence estimation for speech recognition, in: Proceedings of the ICASSP, 2007, pp. 1485–1488.
[39] C. Meyer, H. Schramm, Boosting HMM acoustic models in large vocabulary speech recognition, Speech Commun. 48 (2006) 532–548.
[40] J. Ming, T. Hazen, J. Glass, D. Reynolds, Robust speaker recognition in noisy conditions, IEEE Trans. ASLP (2007) 1711–1723.
[41] O. Scharenborg, Reaching over the gap: a review of efforts to link human and automatic speech recognition research, Speech Commun. 49 (2007) 336–347.
[42] S. Davis, P. Mermelstein, Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences, IEEE Trans. ASSP 28 (1980) 357–366.
[43] H. Hermansky, Perceptual linear predictive (PLP) analysis of speech, J. Acoust. Soc. Am. 87 (1990) 1738–1752.
[44] J. Campbell, Speaker recognition: a tutorial, Proc. IEEE 85 (1997) 1437–1462.
[45] Y. Liu, P. Fung, State-dependent phonetic tied mixtures with pronunciation modeling for spontaneous speech recognition, IEEE Trans. ASLP 12 (2004) 351–364.
[46] M. Gales, S. Young, The application of hidden Markov models in speech recognition, Found. Trends Signal Process. 1 (2007) 1738–1752.
[47] X. Huang, Y. Ariki, M. Jack, Hidden Markov Models for Speech Recognition, Edinburgh University Press, Edinburgh, 1990.
[48] K.-F. Lee, Automatic Speech Recognition: The Development of the SPHINX System, Kluwer, Boston, 1989.
[49] A. Poritz, Hidden Markov models: a guided tour, in: ICASSP, 1988, pp. 7–13.
[50] L. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, Proc. IEEE 77 (2) (1989) 257–286.
[51] T. Moon, The expectation–maximization algorithm, IEEE Signal Process. Mag. 13 (6) (1996) 47–60.
[52] F. Jelinek, Continuous speech recognition by statistical methods, Proc. IEEE 64 (1976) 532–556.
[53] R. Blahut, Fast Algorithms for Digital Signal Processing, Addison-Wesley, Reading, MA, 1985.
[54] H.-L. Lou, Implementing the Viterbi algorithm, IEEE Signal Process. Mag. 12 (5) (1995) 42–52.
[55] L. Liporace, Maximum likelihood estimation for multivariate observations of markov sources, IEEE Trans. Inf. Theory 28 (1982) 729–734.
[56] L. Bahl, F. Jelinek, R. Mercer, A maximum likelihood approach to continuous speech recognition, IEEE Trans. PAMI 5 (1983) 179–190.
[57] M. Mohri, F. Pereira, M. Riley, Weighted finite state transducers in speech recognition, Comput. Speech Lang. 16 (2002) 69–88.
[58] A. Rosenberg, L. Rabiner, J. Wilpon, D. Kahn, Demisyllable-based isolated word recognition systems, IEEE Trans. Audio Speech Signal Process. 31 (1983) 713–726.
[59] A. Lazaridès, Y. Normandin, R. Kuhn, Improving decision trees for acoustic modeling, in: Proceedings of the International Conference on Spoken LP, 1996, pp. 1053–1056.
[60] X. Huang, H. Hon, M. Hwang, K. Lee, A comparative study of discrete, semicontinuous, and continuous hidden Markov models, Comput. Speech Lang. 7 (1993) 359–368.
[61] B.-H. Juang, L. Rabiner, Mixture autoregressive hidden Markov models for speech signals, IEEE Trans. ASSP 33 (1985) 1404–1413.
[62] H. Bourlard, H. Hermansky, N. Morgan, Towards increasing speech recognition error rates, Speech Commun. 18 (1996) 205–231.
[63] O. Siohan, Y. Gong, J.-P. Haton, Comparative experiments of several adaptation approaches to noisy speech recognition using stochastic trajectory models, Speech Commun. 18 (1996) 335–352.
[64] V. Deng, M. Aksmanovik, Speaker-independent phonetic classification using HMMs with mixtures of trend functions, IEEE Trans. Speech Audio Process. 5 (1997) 319–324.
[65] M. Russell, W. Holmes, Linear trajectory segmental HMM's, IEEE Signal Process. Lett. 4 (1997) 72–74.
[66] I. Hetherington, PocketSUMMIT: small-footprint continuous speech recognition, in: Proceedings of the Interspeech, 2007, pp. 1465–1468.
[67] K. Church, Parsing in Speech Recognition, Kluwer, Dordrecht, 1987.
[68] S. Katz, Estimation of probabilities from sparse data for the language model component of a speech recognizer, IEEE Trans. ASSP 35 (1987) 400–401.
[69] H. Lin, J. Bilmes, D. Vergyri, K. Kirchhoff, OOV detection by joint word/phone lattice alignment, in: IEEE Automatic Speech Recognition and Understanding Workshop, 2007, pp. 341–344.
[70] K. Truong, D. van Leeuwen, Automatic discrimination between laughter and speech, Speech Commun. 49 (2007) 144–158.
[71] J. Fiscus, J. Ajot, J. Garofolo, The Rich Transcription 2007 Meeting Recognition Evaluation, in: Joint Proceedings of Multimodal Technologies for Perception of Humans, 2007.
[72] S. Tranter, D. Reynolds, An overview of speech diarization systems, IEEE Trans. ASLP 14 (2006) 1557–1565.
[73] Y.-F. Liao, Z.-H. Chen, Y.-T. Juang, Latent prosody analysis for robust speaker identification, IEEE Trans. ASLP 15 (2007) 1870–1883.
[74] L. ten Bosch, K. Kirchhoff, Bridging the gap between human and automatic speech recognition, Speech Commun. 49 (2007) 331–335.
[75] TIMIT Acoustic–Phonetic Continuous Speech Corpus, National Institute of Standards and Technology Speech Disc 1-1.1, NTIS PB91-505065, 1990.

About the Author—DOUGLAS O'SHAUGHNESSY has been a Professor at INRS-EMT (formerly, INRS-Telecommunications), a constituent of the University of Quebec, in Montreal, Canada, since 1977. During this same period, he has also taught as Adjunct Professor at McGill University in the Department of Electrical and Computer Engineering. For the periods 1991–1997 and 2001–present, he has been Program Director of INRS-EMT as well. Dr. O'Shaughnessy has worked as a Teacher and Researcher in the Speech Communication field for more than 30 years. His interests and research include automatic speech synthesis, analysis, coding, enhancement, and recognition. His research team is currently working to improve various aspects of automatic voice dialogues.

Dr. O'Shaughnessy was educated at the Massachusetts Institute of Technology, Cambridge, MA (B.Sc. and M.Sc. in 1972; Ph.D. in 1976). He is a Fellow of both the Acoustical Society of America (1992) and of the IEEE (2006). From 1995 to 1999, he served as an Associate Editor for the IEEE Transactions on Speech and Audio Processing, and has been an Associate Editor for the Journal of the Acoustical Society of America since 1998. He also served as a member of the IEEE Technical Committee for Speech Processing during 1981–1985, and was recently re-elected to that post. Dr. O'Shaughnessy was the General Chair of the 2004 International Conference on Acoustics, Speech and Signal Processing (ICASSP) in Montreal, Canada. He just finished a three-year elected term as a Member-at-Large of the IEEE SPS Board of Governors, and was a recent member of the IEEE SPS Conference Board.

Dr. O'Shaughnessy has served on several Canadian research grant panels: for FCAR and for NSERC, as well as for NSF. He has also served on organizing technical committees for ICSLP and Eurospeech. He is the Author of the textbook Speech Communications: Human and Machine (first edition in 1986 from Addison-Wesley; completely revised edition in 2000 from IEEE Press). In 2003, with Li Deng, he Co-authored the book Speech Processing: A Dynamic and Optimization-Oriented Approach (Marcel Dekker Inc.). He has presented tutorials on speech recognition at ICASSP-96 in Atlanta, ICASSP-2001 in Orlando, and at ICC-2003 in Anchorage. He has published more than 30 articles in the major speech journals, is a regular presenter at the major speech conferences of Eurospeech and ICSLP, and has had papers at almost every ICASSP since 1986 (more than 130 conference papers).