Int J Speech Technol (2007) 10: 183–195
DOI 10.1007/s10772-009-9026-8

Arabic broadcast news transcription system

Mansour Alghamdi · Moustafa Elshafei · Husni Al-Muhtaseb

Received: 31 January 2009 / Accepted: 11 March 2009 / Published online: 1 April 2009

Abstract This paper describes the development of an Arabic broadcast news transcription system. The presented system is a speaker-independent large vocabulary natural Arabic speech recognition system, and it is intended to be a test bed for further research into the open ended problem of achieving natural language man-machine conversation. The system addresses a number of challenging issues pertaining to the Arabic language, e.g. generation of fully vocalized transcription, and rule-based spelling dictionary. The developed Arabic speech recognition system is based on the Carnegie Mellon university Sphinx tools. The Cambridge HTK tools were also utilized at various testing stages.

The system was trained on 7.0 hours of a 7.5 hours of Arabic broadcast news corpus and tested on the remaining half an hour. The corpus was made to focus on economics and sport news. At this experimental stage, the Arabic news transcription system uses five-state HMM for triphone acoustic models, with 8 and 16 Gaussian mixture distributions. The state distributions were tied to about 1680 senons. The language model uses both bi-grams and tri-grams. The test set consisted of 400 utterances containing 3585 words. The Word Error Rate (WER) came initially to 10.14 percent. After extensive testing and tuning of the recognition parameters the WER was reduced to about 8.61% for non-vocalized text transcription.

M. Alghamdi (✉) · M. Elshafei · H. Al-Muhtaseb
King Abdulaziz City of Science and Technology, Riyadh, Saudi Arabia
e-mail: mghamdi@kacst.edu.sa

M. Elshafei · H. Al-Muhtaseb
King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia

Keywords Arabic speech recognition · News transcription · Arabic speech corpus · Phonetic dictionary · Sphinx training · Arabic natural language · HMM

1 Introduction

Automatic Speech Recognition (ASR) is a key technology for a variety of industrial and IT applications. It extends the reach of IT across people as well as applications. Automatic Speech Recognition (ASR) is gaining a growing role for a variety of applications such as; hands-free operation and control, automatic query answering, telephone interactive voice response systems, automatic dictation (speech-to-text transcription), and automatic speech translation. In fact, speech communication with computers, PCs, and household appliances is envisioned to be the dominant human-machine interface in the near future.

The majority of the recent successes in building speech recognition systems for various languages is attributed to the statistical approach for speech recognition (Baker 1975; Huang et al. 2001; Jelinek 1976, 1998). The statistical approach is itself dominated by the powerful statistical technique called Hidden Markov Model (HMM) (Rabiner 1989; Rabiner and Juang 1993). The HMM-based ASR technique has led to many successful applications requiring large vocabulary speaker-independent continuous speech recognition (Huang et al. 2001; Lee 1988; Young 1996).

In the HMM-based technique words in the target vocabulary are modeled as sequences of phonemes, while each phoneme is modeled as a sequence of HMM states. In standard HMM-based systems, the likelihoods, or the emission probability, of a certain frame observation being produced by a state is estimated using traditional Gaussian mixture

models. The use of HMM with Gaussian mixtures has several notable advantages such as a rich mathematical framework, and efficient learning and decoding algorithms.

The HMM-based technique essentially consists of recognizing speech by estimating the likelihood of each phoneme at contiguous, small frames of the speech signal, then a search procedure is used to find, amongst the words in the vocabulary list, the phoneme sequence that best matches the sequence of phonemes of the spoken word.

Two notable successes in the academic community in developing high performance large vocabulary speaker independent speech recognition systems are the HMM tools, known as the HTK toolkit, developed at Cambridge University (HTK speech recognition toolkit, http://htk.eng.cam.ac.uk/; Young 1994; Young et al. 1999); and the Sphinx system developed at Carnegie Mellon University (CMU Sphinx Group, http://www.speech.cs.cmu.edu/sphinx/Sphinx.html; The Sphinx Project Open Source Speech Recognition Engines, http://cmusphinx.sourceforge.net/html/cmusphinx.php; Huang et al. 1993; Lamere et al. 2003; Lee et al. 1990; Ravishankar 1996; Sphinx-4 Java-based Speech Recognition Engine, http://cmusphinx.sourceforge.net/sphinx4/; Sphinx-4 trainer design 2003) over the last two decades. HTK is a general-purpose open-source tool (Young 1994) for building HMM-based models and is provided with good documentations and has been utilized as an additional resource of tools during the development of this project.

The Sphinx tools can be used as well for developing wide spectrum of speech recognition tasks. For example, the Sphinx-II (Huang et al. 1993; Lee et al. 1990) uses the Semi-Continuous Hidden Markov Models (SCHMM) to reduce the number of parameters and the computer resources required for decoding, but has limited accuracy and complicated training procedure. On the other hand Sphinx-III uses the Continuous Hidden Markov Models (CHMM) with higher performance, but requires substantial computer resources (CMU Sphinx Group, http://www.speech.cs.cmu.edu/sphinx/Sphinx.html; The Sphinx Project Open Source Speech Recognition Engines, http://cmusphinx.sourceforge.net/html/cmusphinx.php; Ravishankar 1996). Sphinx-IV, which was developed in Java, can be used for building platform independent speech recognition applications (Lamere et al. 2003; Sphinx-4 Java-based Speech Recognition Engine, http://cmusphinx.sourceforge.net/sphinx4/; Sphinx-4 trainer design 2003).

Development of an Arabic speech recognition is a multi-discipline effort, which requires integration of Arabic phonetics (Alghamdi 2000, 2003), Arabic speech processing techniques (Alghamdi et al. 2002; Elshafei-Ahmed 1991), and Natural language (Elshafei et al. 2002, 2006a, 2006b). Development of an Arabic speech recognition system has recently been addressed by a number of researchers. Al-Otaibi (2001) studied different approaches in building the Arabic speech corpus, and proposed a new technique for labeling Arabic speech. He reported a recognition rate for speaker dependent ASR of 93.78%. The ASR was built using the HTK toolkit. A workshop was held in 2002 at John Hopkins University (Kirchhofl et al. 2003) to define and address the challenges in developing a speech recognition system using Egyptian dialectic Arabic for telephone conversations. They proposed to use Romanization method for transcription of the speech corpus. Billa et al. (2002) addressed the problems of indexing of Arabic news broadcast, and discussed a number of research issues for Arabic speech recognition, e.g., absence of short vowels in written text and the presence of compound words that are formed by the concatenation of certain conjunctions, prepositions, articles, and pronouns, as prefixes and suffixes to the word stem. Solatu (2007) reported advancements in the IBM system for Arabic speech recognition as part of the continuous effort for the GALE project. The system consists of multiple stages that incorporate both vocalized and non-vocalized Arabic Speech model. The system also incorporates training corpus of 1800 hours of unsupervised Arabic speech. There are a number of other attempts to build AASR, but they directed towards either limited vocabulary, or speaker dependant system (Alimi and Ben Jemaa 2002; Alotaibi 2004; Bahi and Sellami 2003; El Choubassi et al. 2003; El-Ramly et al. 2002).

This paper describes the development and the evaluation of a natural language, large vocabulary, speaker independent Automatic Arabic Speech Recognition (AASR) system. In Sect. 2, we describe the Arabic broadcast news corpus. Then in Sect. 3, we introduce the Arabic phonetic dictionary. A brief description of the main components of the system is given in Sect. 4, while a summary of the training steps is provided in Sect. 5. Finally, Sect. 6 provides a detailed evaluation of the developed AASR system.

2 Arabic broadcast news corpus

The development of a speech recognition system requires in the first place a speech corpus. The developed corpus is based on radio and TV news transcription in the Modern Standard Arabic (MSA). The MSA is widely used and accepted over the entire Arabic region. The audio files were recorded from many Arabic TV news channels, a total of 235 news items. 41 news items cover sport news, and the rest of the items covers mainly economic news. 88 of the news items were by female speakers. The audio items sums up to 7.57 hours of speech. These audio items contain a reasonable set of vocabulary for development and testing the continuous speech recognition system. The recorded speech was divided into 6146 audio files. The length of wave files

varies from 0.8 seconds to 15.1 seconds, with an average file length of 4.43 seconds. Recently, with the increasing interest in Arabic language, the Linguistic Data Consortium (LDC) has produced a number of Arabic speech corpora. However, the available Arabic broadcast news is mainly from one news channel, and still in the raw stage (Solatu 2007).

The audio files were sampled at 16 kHz. Additionally, a 0.1 second silence period is added to the beginning and end of each file. Some of the files have background noise that are of the following types:

1. Background music that accompanies the news headlines. Although this kind of music was deliberately avoided while recording, some files may still have fainting music at the beginning.
2. A few files have relatively high level of background noise. These cases occur when the reporter is in an open location such as a stadium or a stock market.
3. Some of the files contain live translation of foreign speech. The foreign speech is usually at a lower volume but not completely muted.

All the audio files are accompanied by their corresponding orthographic transcription files. The orthographic transcription is a verbatim record of what was actually said. The orthographic transcription form the basis for all other transcriptions and annotations. Full corpus transcription should include as well hesitations, repetitions, false starts and other non speech sounds. All the 6146 files were orthographically transcribed with fully vocalized text. The transcription is meant to reflect the way the speaker utters the words, even if the utterance is grammatically wrong. Thus, grammatical 'errors' were not to be corrected and broken-off words were written down as such (they remained incomplete).

The total words in the corpus is 52,714 words, while the vocabulary is 17,236 words. The transcription of the audio files was first prepared using normal non-vocalized text. Then, an automatic vocalization algorithm was used for fast generation of the Arabic diacritics (short vowels). The algorithm for automatic vocalization is described in detail in Elshafei et al. (2006b). We formulated the problem of generating Arabic vocalized text from non-vocalized text using a Hidden Markov Model (HMM) approach. The word sequence of non-vocalized Arabic text is considered as an observation sequence from an HMM, where the hidden states are the possible vocalized expressions of the words. The optimal sequence of vocalized words (or states) is then obtained efficiently using Viterbi Algorithm. However, the correct letter transcription came to about 90% since, the system was trained on different text subjects. Hand editing was then necessary to bring the transcription to the desired accuracy level.

3 Arabic phonetic dictionary

Table 1 shows the classification of the Arabic consonants, while Table 2 shows the phoneme set used in training and their corresponding symbols. Table 2 shows also illustrative examples of the vowels usage. A detailed description of the Arabic phone set can be found in Algamdi (2003), Elshafei-Ahmed (1991).

Phonetic dictionaries are essential components of large-vocabulary natural language speaker-independent speech recognition systems. Lexicon lookup is a simple but efficient way to acquire phonetic word transcriptions. Yet, not every orthographic unit is a plain word. Some speech fragments contain sloppy speaking styles including broken-off words, mispronunciations and other spontaneous speech effects.

Given an alphabet of spelling symbols (graphemes) and an alphabet of phonetic symbols, a mapping should be achieved to transliterate strings of graphemes into strings of phonetic symbols. It is well known that this mapping is difficult because in general, not all graphemes are realized in the phonemic transcription, and the same grapheme may correspond to different phonetic symbols, depending on the context. Grapheme-to-phoneme conversion is also a central task in any text-to-speech system (Alghamdi et al. 2002; Elshafei et al. 2002). This work uses mainly a rule-based technique to generate Arabic phonetic dictionaries for a large vocabulary speech recognition system. In Ali et al. (2008) the authors presented a rule-based approach to generate Arabic phonetic dictionaries for a large vocabulary speech recognition system. The system used classic Arabic pronunciation rules, common pronunciation rules of Modern Standard Arabic, as well as morphologically driven rules.

A full network of alternative phonetic transcriptions is generated on the basis of orthographic information. Arabic provides multiple phonetic transcriptions for most of the standard words. Lexicon lookup is also used for foreign words. The pronunciation rules and the phone set were validated by test cases. The tool takes care of the following issues:

1. Choosing the correct phoneme combination based on the location of the letters and their neighbors.
2. Providing multiple pronunciations for words that might be pronounced in different ways according to:
   a. The context in witch the words is uttered.
   b. Words that have multiple readings due to dialect issues.
   c. Foreign names.

We defined a set of rules based on regular expressions to define the phonemic definition of words. The tools scans the word letter by letter, and if the conditions of a rule for a specific letter are satisfied, then a selected replacement for that letter is added to a tree structure that represents all the possible pronunciations for that words.

Table 1 IPA classification of Arabic phonemes (Garofolo et al. 1997)

			Bilibial	Libio-dental	Inter-dental	Alveodental Alveolar	Palatal	Velar	Oropha-ryngeal Uvular	Pharyn-geal	Glottal
Stops	Voiced	Pharyngealized				D ض					
			b ب			d د					
	Unvoiced	Pharyngealized				T ط			q ق		
						t ت		k ك			E ء
Fricative	Voiced	Pharyngealized			∂ ظ				γ غ		
					∂ ذ	z ز				? ع	
	Unvoiced	Pharyngealized				S ص			x خ		
				f ف	Θ ث	s س	∫ ش			H ح	h هـ
Affricative							dz ج				
Nasals	Voiced		m م			n ن					
Resonants	Voiced	Pharyngealized				L ř ر ل					
			W و			l r ر ل	y ى				

Each rule has the following structure:

LETTER:

(precondition) . (post-condition) -> replacement

Where LETTER represents the current letter in the word, precondition and post-condition are regular expressions that represent other letters surrounding the current letter, and replacement is the replacement phoneme or phonemes. The number of pronunciations in the developed phonetic dictionary is 28,682 entries. A sample from the developed phoneme dictionary is listed below.

آبَارِ E AE: B AE: R IX N; E AE: B AA: R IX N; E AA: B AA: R IX N

آخَرْ E AE: KH AA R; E AA: KH AA R

آخَرَ E AE: KH AA R AA; E AA: KH AA R AA

آخِرْ E AE: KH IX R

آلافٍ E AE: L AE: F IH N

آلافْ E AE: L AE: F

آلافِ E AE: L AE F IH

Table 2 The phoneme list used in training

الرمز الصوتي	الحرف	الرمز الصوتي	الحرف
/AE/	ـَ ◀ بَ	/HH/	ح
/AE:/	ـَا ◀ بَاب	/KH/	خ
/AA/	ـَ ◀ خَ	/D/	د
/AA:/	ـَا ◀ صَاض	/DH/	ذ
/AH/	ـَ ◀ قد	/R/	ر
/UH/	ـُ ◀ بُ	/Z/	ز
/UW/	ـُو ◀ دُون	/S/	س
/UX/	ـُ ◀ غُصن	/SS/	ص
/IH/	ـِ ◀ بنت	/DD/	ض
/IY/	ـِي ◀ فِيل	/TT/	ط
/IX/	ـِ ◀ صِنف	/DH2/	ظ
/AW/	ـَو ◀ لَوم	/AI/	ع
/AY/	ـَي ◀ ضَيف	/GH/	غ
/UN/	ـُ ◀ تُنجي	/F/	ف
/AN/	ـَ ◀ نَم	/V/	ف ◀ فيزا
/IN/	ـِ ◀ مِما	/Q/	ق
/E/	ء	/K/	ك
/B/	ب	/L/	ل
/T/	ت	/M/	م
/TH/	ث	/N/	ن
/JH/	جيم فصحة	/H/	هـ
/G/	جيم مصرية	/W/	و
/ZH/	جيم معطشة	/Y/	ي

4 System description

In this section we describe the various components of the Arabic broadcast news transcription system. Figure 1 illustrates the main components of the AASR system.

The Front-End: This sub-system provides the initial step in converting sound input into a feature vectors to be usable by the rest of the system. The recorded speech is sampled at a rate of 16 ksps. The analysis window is 25.6 msec (about 410 samples), with consecutive frames overlap by 10 msec. Each window is pre-emphasized and is multiplied by a Hamming window (CMU Sphinx Group, http://www.speech.cs.cmu.edu/sphinx/Sphinx.html; The Sphinx Project Open Source Speech Recognition Engines, http://cmusphinx.sourceforge.net/html/cmusphinx.php; Huang et al. 2001). The basic feature vector uses the Mel Frequency Cepstral Coefficients (MFCC). The Mel-Frequency scale is a linear frequency spacing below 1000 Hz and a logarithmic spacing above 1000 Hz. The MFCCs are obtained by taking the Discrete Cosine Transform (DCT) of the log power spectrum from Mel spaced filter banks (Alghamdi 2000). Thirteen Mel frequency cepstra are computed, $x(0), x(1), \ldots, x(12)$, for each window of 25 ms, with adjacent windows overlapped by 15 ms.

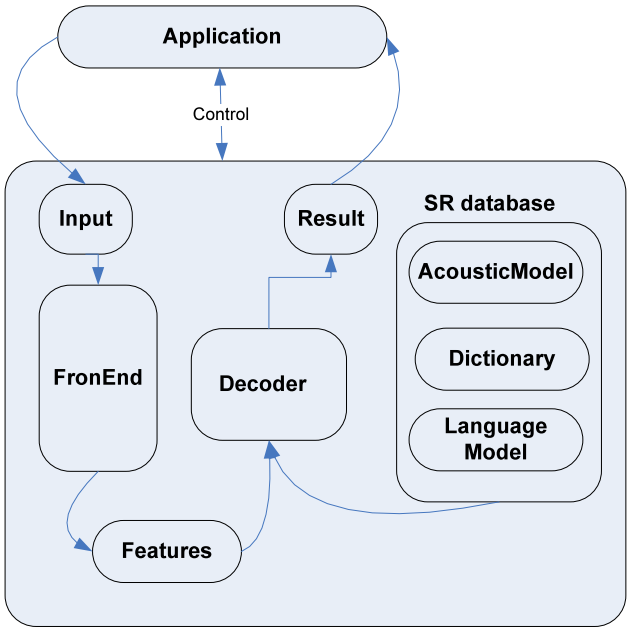

Fig. 1 Speech recognition system's architecture

The basic feature vector is highly localized. To account for the temporal properties, Two other derived vectors are con-

structed from the basic MFCC coefficients: a 40-ms differenced MFCCs, and a second order differenced MFCCs, giving a feature vector dimension of 39.

The SR database: This sub-system contains the details that describe the recognized language itself. This subsystem is where most of the adjustments are made in order to support the Arabic Language recognition. It consists of three main modules:

The Acoustic Model: This module provides the Hidden Markov Models (HMMs) of the Arabic triphones to be used to recognize speech. The basic HMM model used in this work is a 5-state forward model as shown in Fig. 2.

The Language Model: This module provides the statistical language model of the natural Arabic language based on the transcription of the entire corpus.

The Dictionary: This module serves as an intermediary between the Acoustic Model and the Language Model. It contains the words available in the language and the pronunciation of each word in terms of the phonemes available in the acoustic model.

The Decoder: This sub-system performs the actual recognition task. When speech is entered into the system, the Front-End converts the incoming speech into feature vectors as described earlier. The Decoder takes these features, in addition to the acoustic models, the phonetic dictionary, and the language model, and searches for the most likely sequence of words, given the sequence of feature vectors for the speech signal.

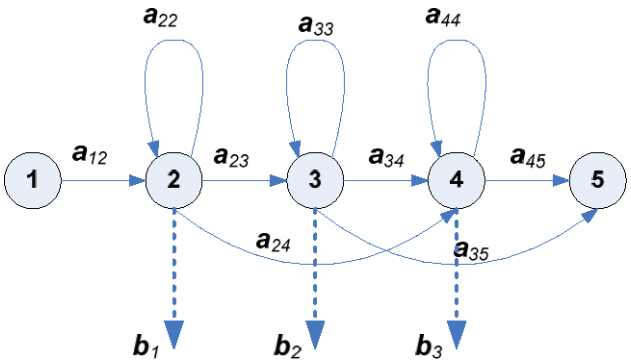

Fig. 2 The 5-states HMM triphone model

5 Training steps

Training the complete speech recognition engine consists of building two models; the acoustic model, and the language model.

5.1 Acoustic model training

The training procedure consists of three phases as shown in Fig. 3. Each phase consists of three steps: model definition, model initialization, and model training. In the first phase, Context-Independent (CI) phoneme models are built. Baum-Welch re-estimation algorithm is used iteratively to estimate the transition probabilities of the CI HMM models (Rabiner 1989; Rabiner and Juang 1993). In this phase the emission probability distribution of each state is taken to be a single normal distribution.

During the second phase, an HMM model is built for each triphone, that is a separate model for each left context and right context for each phoneme. During this context-dependant (CD) phase, triphones are added to the HMM set. In the model definition stage, all the possible triphones will be created, and then the triphones below a certain frequency are excluded. After defining the needed triphones, states are given serial numbers as well (continuing the same count). The initialization stage copies the parameters from the CI phase. Similar to the previous phase, the model training stage consists of iterations of the Baum-Welch algorithm (6 to 10 times) followed by a normalization process.

The number of tri-phones in the training database is 10326. Table 3 gives the number of tri-phones for each Arabic phoneme according to the current speech corpus. For example, the /AA/ was found to have 96 cases of different left/right contexts.

The performance of the model generated by the previous phase is improved by tying some states of the HMMs. These

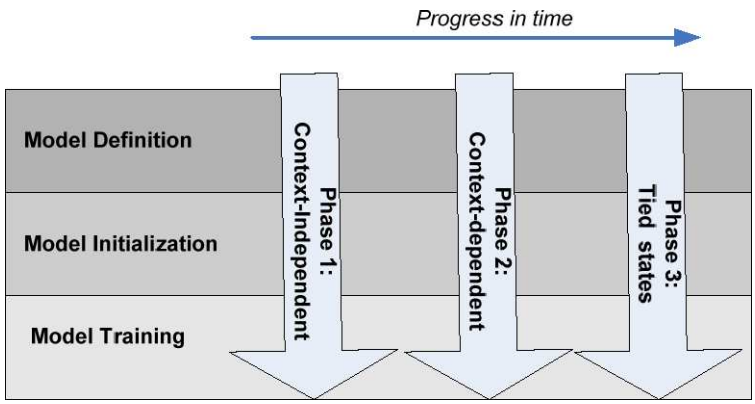

Fig. 3 Acoustic model-building steps

Table 3 Number of tri-phones for each phoneme in the AASR

Phone	Triphones	Phone	Triphones
AA	96	IX:	51
AA:	70	IY	372
AE	542	JH	181
AE:	389	K	225
AH	64	KH	130
AH:	40	L	560
AI	289	M	344
AW	77	N	454
AY	104	Q	238
B	324	R	460
D	356	S	302
DD	137	SH	144
DH	65	SS	156
DH2	41	T	393
E	479	TH	106
F	286	TT	161
GH	83	UH	487
H	258	UW	257
HH	195	UX	70
IH	657	W	187
IX	85	Y	218
IX:	51	Z	192

tied states are called senons (Bellagarda and Nahamoo 1988; Digalakis et al. 1996; Hwang et al. 1993; Hwang and Huang 1993). In the third training phase, the number of distributions is reduced by combining similar state distributions. The process of creating these senons involves classification of phonemes according to their acoustic properties (Singh et al. 1999). A senon is also called a tied-state and is obviously shared across the triphones which contributed to it. In the last phase, the senons probability distributions are re-estimated and presented by a Gaussian mixture model by iterative splitting of the Gaussian distributions. In this reported work, the emission probabilities of the senons are modeled and tested with mixtures of 8 and 16 diagonal covariance Gaussian distributions.

5.2 Language model

The probability $P(W)$ of a sequence of words $W = w_1, w_2, \ldots, w_L$ is computed by a Language Model (LM). In general $P(W)$can be expressed as follows:

$$P(W) = P(w_1, w_2, \ldots, w_L) = \prod_{i=1}^{L} P(w_i | w_1, \ldots, w_{i-1}). \tag{1}$$

In a bigram model the most recent word is used to construct the condition probability of the next word, while in a trigram model the most recent two words of the history are used to condition the probability of the next word. The probability of a word sequence using bigrams is given by (Clarkson and Rosenfeld 1997; Huang et al. 2001):

$$P(W) \approx \prod_{i=1}^{L} P(w_i | w_{i-1}). \tag{2}$$

For the trigram model

$$P(W) \approx \prod_{i=1}^{L} P(w_i | w_{i-2}, w_{i-1}). \tag{3}$$

Speech recognition systems treat the recognition process as one of maximum a-posteriori estimation, where the most likely sequence of words is estimated, given the sequence of feature vectors for the speech signal. The score of a particular word sequence W evaluated by a given utterance X is a weighted summation of the acoustic score and language score (CMU Sphinx Group, http://www.speech.cs.cmu.edu/sphinx/Sphinx.html; The Sphinx Project Open Source Speech Recognition Engines, http://cmusphinx.sourceforge.net/html/cmusphinx.php):

$$score(W|X) = \log P(X|HMM(W)) + \beta \log P(W). \tag{4}$$

The argument on the right hand side of (4) has two components: the probability of the utterance given the acoustic model of the word sequence, and the probability of the sequence of words itself, $P(W)$. The first component is provided by the acoustic model. The second component is estimated using the language model.

The language probability is raised to an exponent for recognition. Although there is no clear statistical justification for this, it is frequently explained as "balancing" of language and acoustic probability components during recognition and is known to be very important for good recognition. Here β is the language weight. Experimental values of β typically lie between 6 and 13 (CMU Sphinx Group, http://www.speech.cs.cmu.edu/sphinx/Sphinx.html; The Sphinx Project Open Source Speech Recognition Engines, http://cmusphinx.sourceforge.net/html/cmusphinx.php).

Similarly, it has also been found useful to include a *word insertion penalty* (WIP) parameter which is a fixed penalty for each new word hypothesized by the decoder. It is effectively another multiplicative factor in the language model probability computation (before the application of the language weight). This parameter has usually ranged between 0.2 and 0.7, depending on the task. These two parameters are tuned on a test set after training of the acoustic model.

The creation of a language model from a training text consists of the following steps as depicted in Fig. 4:

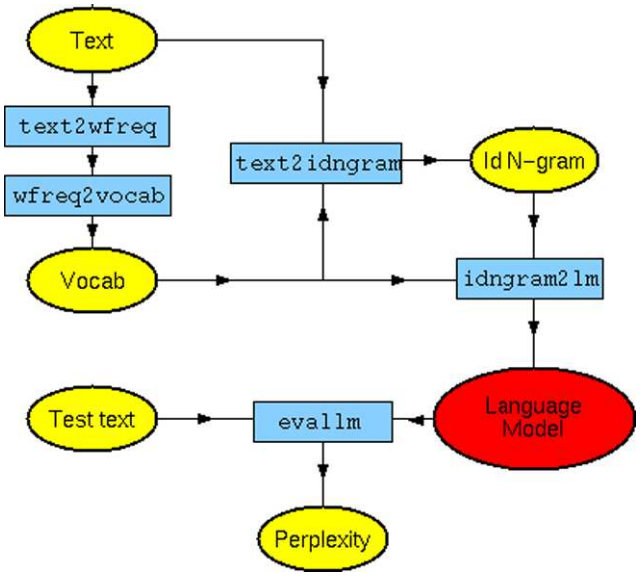

Fig. 4 Steps for creating and testing language model (Clarkson and Rosenfeld 1997)

- Compute the word unigram counts.
- Convert the word unigram counts into a task vocabulary.
- Generate a binary id 3-gram of the training text, based on this.
- Convert the Id N-gram into a binary format language model.

In this work (KACST v1.09) the number of unigrams is 17237, the number of bigrams is 42660, and the number of trigrams is 501481.

6 Evaluation of the AASR system

In this section we present an extensive evaluation of the developed AASR system. First, in Sect. 6.1 we introduce the performance metrics used in this evaluation. In Sect. 6.2 we provide a benchmark performance of the CMU ASR for the English language. In Sect. 6.3 we present the performance of the base AASR before performance tuning, followed by another section covering the performance tuning. Finally in Sect. 6.5 we present the performance of the improved version after tuning.

6.1 Performance metrics

We developed a tool which compare the Arabic recognition result with the reference text. The tool can be set to compare fully vocalized text or the non-vocalized text. The tool compares the two texts**,** line by line and computes the number of substitution errors (S), deletion errors (D) and insertion errors (I). The percentage correct is defined as

$$\text{Percent Correct} = \frac{N - D - S}{N} \times 100\% \tag{5}$$

where N is the total number of labels in the reference transcriptions. Notice that this measure ignores insertion errors. For many purposes, the percentage accuracy defined as

$$\text{Percent Accuracy} = \frac{N - D - S - I}{N} \times 100\%. \tag{6}$$

The reported WER in this work is defined to be

$$\begin{aligned}\text{WER} &= 100 - \text{Percent Accuracy} \\ &= \frac{D + S + I}{N} \times 100\%. \end{aligned} \tag{7}$$

6.2 Benchmarking recognition performance

Before we evaluate the performance of the Arabic speech recognition system, it is imperative to review the performance of the same recognition engine (CMU Sphinx) as reported in a number of publications. The Sphinx engine was tested under many recognition tasks, including isolated digits, connected digits, small vocabulary, medium vocabulary (1000, 5000, and 20,000 words) (Price et al. 1988), and large vocabulary (64,000 words) (Garofolo et al. 1997). Table 4 and Fig. 5 provide a summary of the sphinx performance.

For large vocabulary systems, the performance of the decoder was tested on the DARPA Hub-4 Broadcast news project (Ortmanns et al. 1998; Placeway et al. 1997; Siegler et al. 1997). The HUB-4 Broadcast News Speech Corpus contains a total of 104 hours of broadcasts from various television networks and radio networks with corresponding transcripts. The acoustic models used for this test had 5000 tied states with 32 Gaussians per state. A trigram LM with 4.7M bigrams and 15.5M trigrams covering a vocabulary of 64,000 words was used.

6.3 Evaluation of KACST v1.09

This section presents the evaluation of the baseline AASR, KACST V1.09, before recognition tuning. This base system uses 5 states HMM with three emitting states. The state probability distribution uses continuous density of 8 Gaussian mixture distributions. The state distributions were tied to about 1636 senons. The language model uses bigrams and trigrams as explained in the previous sections. The size of the vocabulary is 17,234 words. The number of entries in the phonetic dictionary is 28,682 entries. The system was trained on about 7.0 hours of speech. The above AASR release system was tested on test corpus of 400 utterances, 3585 words, representing about half an hour of the entire corpus. The test utterances were not included in the training set. Two filler sounds were included in the filler dictionary. After initial inspection of a subset of the sound files, 75 utterances were marked to have either noise or inhalation. The transcription was modified to include noise or the

Table 4 Sphinx performance versus vocabulary size, http://cmusphinx.sourceforge.net/sphinx4/

Vocabulary Size	11	79	1,000	5,000	60,000
WER	0.661	1.30	2.746	7.323	18.845

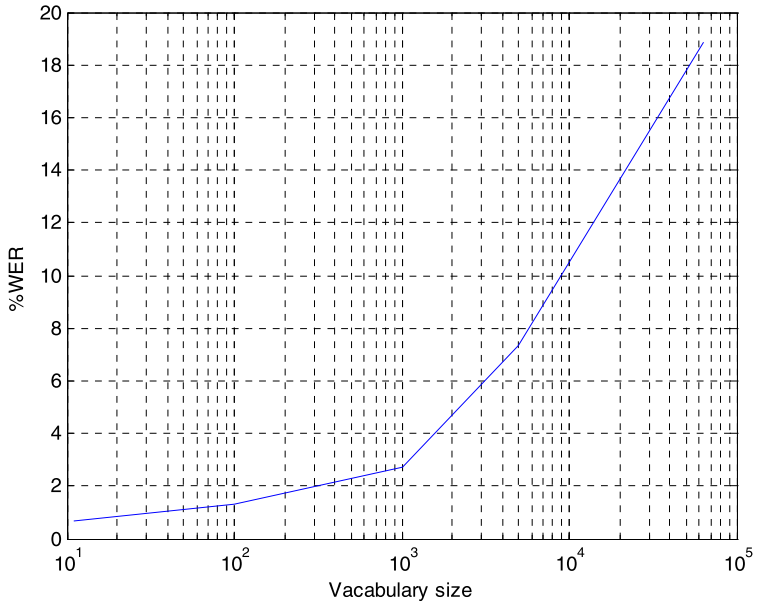

Fig. 5 Typical performance in terms of WER versus vocabulary size for Sphinx 3&4 engines

inhalation words and used to train the models for the filler words.

The initial test result is given below:

Number of correctly recognized words	= 3182
% word recognition accuracy	= 88.76%
Number of word Insertions	= 83
Number of word substitution	= 362
Number of word deletion	= 41
Word Error Rate (WER)	= 13.66%

A sample of the original text and the corresponding recognition result are shown in Table 5.

Analysis of errors indicates that the number of substitutions is high (362 words). The analysis shows that many of the word substitution errors are due to slight differences (deletion/substitution) of diacritical marks. For example,

Original: بِبنَاءِ مِصفَاةٍ نَفطِيَّة بِطَاقةِ أربَع مِئَةِ ألفِ **بَرمِيل** يَومِيًّا بِالتَّعَاوُن مَع نَظِيرَتِهَا السُّعُودِيَّة أَرَامكو

Recognized: بِبنَاءِ مِصفَاةٍ نَفطِيَّة بِطَاقةِ أربَع مِئَةِ ألفِ **برمِيلٍ** يَومِيًّا بِالتَّعَاوُن مَع نَظِيرَتِهَا السُّعُودِيَّة أَرَامكو

Clearly, the recognized sentence would be considered faultless by a native Arabic reader, however, the error analysis indicates that there is two diacritical marks substitution errors in the word "**بَرمِيل**".

Since MSA text is written without diacritical marks, the error analysis was carried out once more after removing all the diacritical marks.The recognition results for the non-vocalized text is shown below

Number of correctly recognized words	= 3306
% word recognition accuracy	= 92.22%
Number of word Insertions	= 83
Number of word substitution	= 238
Number of word deletion	= 41
WER	= 10.1%

6.4 Further enhancements

This section summaries several trials for tuning the recognition parameters to enhance the recognition accuracy of the trained model.

1. On a first trial, 16 Gaussian mixtures where used instead of 8 Gaussians. Increasing the number of Gaussian mixtures is supposed to increase both the accuracy and sen-

Table 5 Recognition result of the fully vocalized transcription

Recognition result	Original text
فقط الثَّنَائِيَّة بِشَأن التِّجَارَةِ الحُرَّة مَعَ الولايَات المُتَّحِدَة	المُفَاوَضَات الثُّنَائِيَّة بِشَأن التِّجَارَةِ الحُرَّة مَعَ الولايَات المُتَّحِدَة
تَعَهَّدَ العَاهِلُ السُّعُودِيِّ المَلِك عَبدُ اللَّه بن عَبدِ العَزِيز بِبَذل قُصَارَى جُهدِهِ مِن أجل رَخَاءِ المُوَاطِنِينَ السُّعُودِيين	تَعَهَّدَ العَاهِلُ السُّعُودِيِّ المَلِك عَبدُ اللَّه بن عَبدِ العَزِيز بِبَذل قُصَارَى جُهدِهِ مِن أجل رَخَاءِ المُوَاطِنِينَ السُّعُودِيين
بِبنَاءِ مِصفَاةٍ نَفطِيَّة بِطَاقَةِ أربَع مِئَةِ ألفِ بَرمِيل يَومِيًّا بِالتَّعَاوُن مَع نَظِيرَتِهَا السُّعُودِيَّة أرَامكو	بِبنَاءِ مِصفَاةٍ نَفطِيَّة بِطَاقَةِ أربَع مِئَةِ ألفِ بَرمِيل يَومِيًّا بِالتَّعَاوُن مَع نَظِيرَتِهَا السُّعُودِيَّة أرَامكو
بَدَأت الشَّرِكَةُ السُّعُودِيَّة لِلصِّنَاعَاتِ الأسَاسِيَّة سَابِك بن	بَدَأت الشَّرِكَةُ السُّعُودِيَّةُ لِلصِّنَاعَاتِ الأسَاسِيَّة سَابِك
	!NH
وَضِمنَ المَجمُوعَةِ إذَا ذَاتِهَا تَدخُل هُولندَا مِئَةِ كَأس العَالم فِي ألمَانيَا بِأفضَل سِجِلٍ فِي التَّصفِيَاتِ الأُورُوبِيَّة	وَضِمنَ المَجمُوعَةِ ذَاتِهَا تَدخُل هُولندَا نِهَايَةِ كَأس العَالم فِي ألمَانيَا بِأفضَل سِجِلٍ فِي التَّصفِيَاتِ الأُولَمبِيَّة
وَذَلِكَ بِفَضل النُّمُوِّ القَوِيِّ الاِقتِصَاد فِي بُلدَانِ مِثلَ الصِّين وَالولايَاتِ المُتَّحِدَة	وَذَلِكَ بِفَضل النُّمُوِّ القَوِيِّ لِلاِقتِصَاد فِي بُلدَانِ مِثلَ الصِّين وَالولايَاتِ المُتَّحِدَة
قَالَ النَاطِقُ الرَّسمِي بِاسمِ وِزَارَةِ النَّفطِ العِرَاقِيَّة عَاصِم جِهَاتِ	قَالَ النَاطِقُ الرَّسمِي بِاسمِ وِزَارَةِ النَّفطِ العِرَاقِيَّة عَاصِم جِهاد
تَحتَ التَّأسِيس برَأس مَالٍ يَصِلُ إلَى نَحو مِلَيَار وَخَمس مِئَة مِلَيَار رِيَال	تَحتَ التَّأسِيس برَأس مَالٍ يَصِلُ إلَى نَحو مِلَيَار وَخَمس مِئَة مِلِيُون رِيَال

Table 6 Recognition accuracy for varying training parameters

	Base V1.09	V1.09.1	V1.09.2	V1.09.3
No. Senons	1500	1000	2000	1500
No. GM terms	8	8	8	16
Accuracy	% Acc = 86.44 $D = 41$, $S = 362$, $I = 83$	% Acc = 85.69 $D = 46$, $S = 391$, $I = 76$	% Acc = 85.16 $D = 46$, $S = 396$, $I = 90$	% Acc = 81.37 $D = 40$, $S = 472$, $I = 156$

sitivity of the language model. However, with the small amount of audio data used, increasing the size of the Gaussian mixtures leads to a slight decline in accuracy, with a noticeable increase in insertion and substitution cases. Deletion, however, got slightly reduced, as summarized in Table 6. The degraded performance is due to poor training of the Gaussian mixer probabilities.

2. The effect of increasing the number of tied state distributions (senons) was also performed. A thumb rule figure for the number of senons is given in Table 7.

   In one build we used 1000 senons, and in another one we used 2000 senons. The result is also reported in Table 6. We explored other numbers of senons up to 3000, but there was no improvement over the base system. It is clear that the number of senons used in the base KACST v1.09 is the best for the size of corpus we have.
3. We also examined the effect of other recognition (decoding) parameters such as the language model weight, beam width, and the word insertion penalty (wip). The language model weight parameter decides how much relative importance is given to the actual acoustic probabilities of the words in the hypothesis. A low language weight gives more chance for words with high acoustic probabilities to be hypothesized, at the risk of hypothesizing spurious words. A value between 6 and 13 is recommended, and by default it is 9.5.

   Similarly, though with lesser impact, is the word insertion penalty (wip), which is a fixed penalty for each new word hypothesized by the decoder. This parameter

Table 7 Number of Senones versus training data size in hours[a]

Amount of training data (hours)	No. of senones
1–3	500–1000
4–6	1000–2500
6–8	2500–4000
8–10	4000–5000
10–30	5000–5500

[a]http://www.cs.cmu.edu/~rsingh/sphinxman/FAQ.html#1

Table 8 Sensitivity analysis of various recognition parameters

	Base V1.09	V1.09.4	V1.09.5
LM weight	9.5	12	10.5
Accuracy	%Acc = 86.44, $D = 41$, $S = 362$, $I = 83$	%Acc = 85.24, $D = 58$, $S = 399$, $I = 72$	%Acc = 86.16, $D = 42$, $S = 375$, $I = 79$
Word Insertion Penalty	0.7	0.4	0.3
Accuracy	%Acc = 86.44, $D = 41$, $S = 362$, $I = 83$	%Acc = 86.47, $D = 42$, $S = 362$, $I = 81$	%Acc = 86.53, $D = 42$, $S = 361$, $I = 80$
Beam Pruning	1.0e-55	1.0e-45	1.0e-65
Accuracy	%Acc = 86.44, $D = 41$, $S = 362$, $I = 83$	%Acc = 81.12, $D = 45$, $S = 512$, $I = 120$	%Acc = 86.86, $D = 41$, $S = 349$, $I = 81$

has usually ranged between 0.2 and 0.7, depending on the task.

4. *Beam Pruning*. Each utterance is processed in a time-synchronous manner, one frame at a time. At each frame the decoder has a number of currently active HMMs to match with the next frame of input speech. But it first discards or deactivates those whose state likelihoods are below some threshold, relative to the best HMM state likelihood at that time. The threshold value is obtained by multiplying the best state likelihood by a fixed beam width. The beam width is a value between 0 and 1, the former permitting all HMMs to survive, and the latter permitting only the best scoring HMMs to survive. Table 8 summarizes the results of recognition tuning.

   Based on the above sensitivity table, we fixed the Beam pruning parameter to 1.0e-65, and the wip = 0.3; The best results we obtained for vocalized text was the following:

Number of correctly recognized words	= 3193
% word recognition accuracy	= 89.86%
Number of word Insertions	= 79
Number of word substitution	= 350
Number of word deletion	= 42
WER	= 13.14%

   For the non-vocalized text, we obtained the following

Number of correctly recognized words	= 3306
% word recognition accuracy	= 92.52%
Number of word Insertions	= 79
Number of word substitution	= 226
Number of word deletion	= 42
WER	= 9.68%

6.5 AASR KACST V1.10

It is clear from the above tests that the limiting factor is the limited data available (7.5 hours), and the need for a more thorough inspection of the recorded speech and its associated transcription. Accordingly, once again we went through extensive inspection of the training errors and the recognition errors one by one. Among the causes of errors:

- High background noise, music, or a second speaker
- Speaker hesitation, inhalation, and other non speech sounds.
- Bad recording (saturated volume, sudden truncation of utterance)
- Bad transcription (unmatched text, wrong/missing words, wrong diacritical marks)

The discovered errors in the transcription or the sound files were corrected. Filler words were also added to the transcriptions if necessary. These corrections led to substantial improvement in the performance and marked as KACST v1.10, in which we used the best tuning parameters.

The following the summary of the performance of KACST v1.10

Number of correctly recognized words	= 3248
% word recognition accuracy	= 90.78%
Number of word Insertions	= 59
Number of word substitution	= 300
Number of word deletion	= 30
WER	= 10.87%

For the non-vocalized text, we obtained the following

Number of correctly recognized words	= 3329
% word recognition accuracy	= 93.04%
Number of word Insertions	= 59
Number of word substitution	= 219
Number of word deletion	= 30
WER	= 8.61

This WER is very much comparable or better than the reported accuracy of the SPHINX English systems with same vocabulary size.

7 Conclusion

This paper reports the first phase towards building a high performance Arabic news transcription system. This phase of work includes establishing an infrastructure for research in Arabic speech and natural Arabic language processing. The work includes building an Arabic broadcast news speech corpus with full vocalized transcription, build an Arabic phonetic dictionary, and an Arabic language statistical model. The AASR system was trained using 7.0 hours of speech, and tested using half an hour of speech (400 utterances).

The WER of fully vocalized transcription was 10.87% and correct word accuracy was 90.78%. For non-vocalized text transcription, the WER was 8.61%, and the correct word recognition accuracy was 93.04%. These results are comparable or better than the reported English recognition results for tasks of the same vocabulary size.

Acknowledgements This work was supported by a grant #AT-24-94 by King Abdulaziz City of Science and Technology. The authors would like also to thank King Fahd University of Petroleum and Minerals for its support in carrying out this project.

References

Alghamdi, M. (2000). *Arabic phonetics*. Riyadh: Attaoobah.

Algamdi, M. (2003). KACST Arabic phonetics database. In *The fifteenth international congress of phonetics science* (pp. 3109–3112). Barcelona.

Alghamdi, M., Elshafei, M., & Almuhtasib, H. (2002). Speech units for Arabic text-to-speech. In *The fourth workshop on computer and information sciences* (pp. 199–212).

Ali, M., Elshafei, M., Alghamdi, M., Al-Muhtaseb, H., & Al-Najjar, A. (2008). Generation of Arabic phonetic dictionaries for speech recognition. In *The 5th international conference on innovations in information technology*, United Arab Emirates, December 2008.

Alimi, A. M., & Ben Jemaa, M. (2002). Beta fuzzy neural network application in recognition of spoken isolated Arabic words. *International Journal of Control and Intelligent Systems, Special Issue on Speech Processing Techniques and Applications*, *30*(2).

Alotaibi, Y. A. (2004). Spoken Arabic digits recognizer using recurrent neural networks. In *Proceedings of the fourth IEEE international symposium on signal processing and information technology* (pp. 195–199), 18–21 Dec. 2004.

Al-Otaibi, F. A. H. (2001). *Speaker-dependant continuous Arabic speech recognition*. M.Sc. Thesis, King Saud University.

Bahi, H., & Sellami, M. (2003). A hybrid approach for Arabic speech recognition. In *ACS/IEEE international conference on computer systems and applications*, 14–18 July 2003.

Baker, J. K. (1975). Stochastic modeling for automatic speech understanding. In R. Reddy (Ed.), *Speech recognition* (pp. 521–542). New York: Academic Press.

Bellagarda, J., & Nahamoo, D. (1988). Tied-mixture continuous parameter models for large vocabulary isolated speech recognition. In *Proc. IEEE international conference on acoustics, speech, and signal processing*.

Billa, J., Noamany, M., Srivastava, A., Liu, D., Stone, R., Xu, J., Makhoul, J., & Kubala, F. (2002). Audio indexing of Arabic broadcast news. In *Proceedings (ICASSP '02). IEEE international conference on acoustics, speech, and signal processing* (Vol. 1, pp. I-5–I-8).

Clarkson, P., & Rosenfeld, R. (1997). Statistical language modeling using the CMU-Cambridge toolkit. In *Proceedings of the 5th European conference on speech communication and technology*, Rhodes, Greece, Sept. 1997.

Digalakis, V., Monaco, P., & Murveit, H. (1996). Genones: Generalized mixture tying in continuous hidden Markov model-based speech recognizers. *IEEE Transactions on Speech and Audio Processing*, *4*(4), 281–289.

El Choubassi, M. M., El Khoury, H. E., Alagha, C. E. J., Skaf, J. A., & Al-Alaoui, M. A. (2003). Arabic speech recognition using recurrent neural networks. In *Proceedings of the 3rd IEEE international symposium on signal processing and information technology* (*ISSPIT*) (pp. 543–547), Dec. 2003.

El-Ramly, S. H., Abdel-Kader, N. S., & El-Adawi, R. (2002). Neural networks used for speech recognition. In *Radio science conference (NRSC 2002). Proceedings of the nineteenth national* (pp. 200–207), March 2002.

Elshafei-Ahmed, M. (1991). Toward an Arabic text-to-speech system. *The Arabian Journal of Science and Engineering*, *16*(4B), 565–583.

Elshafei, M., Almuhtasib, H., & Alghamdi, M. (2002). Techniques for high quality text-to-speech. *Information Science*, *140*(3–4), 255–267.

Elshafei, M., Al-Muhtaseb, H., & Alghamdi, M. (2006a). Statistical methods for automatic diacritization of Arabic text. In *Proceedings 18th national computer conference NCC'18*, Riyadh, March 26–29, 2006.

Elshafei, M., Al-Muhtaseb, H., & Alghamdi, M. (2006b). Machine generation of Arabic diacritical marks. In *Proceedings of the 2006 international conference on machine learning; models, technologies, and applications (MLMTA'06)*, June 2006, USA.

Garofolo, J., Voorhees, E., Auzanne, C., Stanford, V., & Lund, B. (1997). Design and preparation of the 1996 HUB-4 broadcast news benchmark test corpora. In *Proceedings of the DARPA speech recognition workshop* (pp. 15–21). Chantilly: Morgan Kaufmann.

Hagen, S. (2007). The IBM 2006 GALE Arabic ASR system. In *ICASSP*, 2007.

Huang, X., Alleva, F., Hon, H. W., Hwang, M. Y., & Rosenfeld, R. (1993). The SPHINX-II speech recognition system: an overview. *Computer Speech and Language*, *7*(2), 137–148.

Huang, X., Acero, A., & Hon, H. (2001). *Spoken language processing*. Englewood Cliffs: Prentice-Hall.

Hwang, M. Y., & Huang, X. (1993). Shared-distribution hidden Markov models for speech recognition. *IEEE Transactions on Speech and Audio Processing*, *1*(4), 414–420.

Hwang, M. Y., Huang, X. D., & Alleva, F. (1993). Predicting unseen triphones with senones. In *Proc. IEEE international conference on acoustics, speech, and signal processing*.

Jelinek, F. (1976). Continuous speech recognition by statistical methods. *Proceedings of the IEEE*, *64*(4), 532–555.

Jelinek, F. (1998). *Statistical methods for speech recognition*. Cambridge: MIT Press.

Kirchhofl, K., Bilmes, J., Das, S., Duta, N., Egan, M., Ji, G., He, F., Henderson, J., Liu, D., Noamany, M., Schoner, P., Schwartz, R., & Vergyri, D. (2003). Novel approaches to Arabic speech recognition: report from the 2002 John-Hopkins summer workshop. In *ICASSP 2003* (pp. I-344–I-347).

Lamere, P., Kwok, P., Walker, W., Gouvea, E., Singh, R., Raj, B., & Wolf, P. (2003). Design of the CMU Sphinx-4 decoder. In *Proceedings of the 8th European conference on speech communication and technology* (pp. 1181–1184), Geneve, Switzerland, Sept. 2003.

Lee, K. F. (1988). *Large vocabulary speaker-independent continuous speech recognition: the SPHINX system*. PhD Thesis, Carnegie Mellon University.

Lee, K. F., Hon, H. W., & Reddy, R. (1990). An overview of the SPHINX speech recognition system. *IEEE Transactions on Acoustics, Speech and Signal Processing*, *38*(1), 35–45.

Ortmanns, S., Eiden, A., & Ney, H. (1998). Improved lexical tree search for large vocabulary speech recognition. In *Proc. IEEE int. conf. on acoustics, speech and signal proc.*

Placeway, P., Chen, S., Eskenazi, M., Jain, U., Parikh, V., Raj, B., Ravishankar, M., Rosenfeld, R., Seymore, K., Siegler, M., Stern, R., & Thayer, E. (1997). The 1996 HUB-4 Sphinx-3 system. In *Proceedings of the DARPA speech recognition workshop*. Chantilly: DARPA, Feb. 1997. http://www.nist.gov/speech/publications/darpa97/pdf/placewa1.pdf.

Price, P., Fisher, W. M., Bernstein, J., & Pallett, D. S. (1988). The DARPA 1000-word resource management database for continuous speech recognition. In *Proceedings of the international conference on acoustics, speech and signal processing* (Vol. 1, pp. 651–654). New York: IEEE.

Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2).

Rabiner, L., & Juang, B. H. (1993). *Fundamentals of speech recognition*. Englewood Cliffs: Prentice-Hall.

Ravishankar, M. K. (1996). *Efficient algorithms for speech recognition*. PhD Thesis (CMU Technical Report CS-96-143), Carnegie Mellon University, Pittsburgh, PA.

Siegler, M., Jain, U., Raj, B., & Stern, R. M. (1997). Automatic segmentation, classification and clustering of broadcast news audio. In *Proc. DARPA speech recognition workshop*, Feb. 1997.

Singh, R., Raj, B., & Stern, R. M. (1999). Automatic clustering and generation of contextual questions for tied states in hidden Markov models. In *Proc. IEEE int. conf. on acoustics, speech and signal proc.*

Sphinx-4 trainer design (2003). http://www.speech.cs.cmu.edu/cgi-bin/cmusphinx/twiki/view/Sphinx4/Train%erDesign.

Young, S. (1994). *The HTK hidden Markov model toolkit: design and philosophy* (Tech. Rep. CUED/FINFENG/, TR152). Cambridge University Engineering Department, UK, Sept. 1994.

Young, S. (1996). A review of large-vocabulary continuous-speech recognition. *IEEE Signal Processing Magazine*, 45–57.

Young, S. J., Kershaw, D., Odell, J. J., Ollason, D., Valtchev, V., & Woodland, P. C. (1999). *The HTK book.* Entropic.