Syllable language models for Mandarin speech recognition: Exploiting character language models

Xunying Liu[a)]
Cambridge University Engineering Department, Cambridge, United Kingdom

James L. Hieronymus
International Computer Science Institute, Berkeley, California 94704

Mark J. F. Gales and Philip C. Woodland
Cambridge University Engineering Department, Cambridge, United Kingdom

(Received 9 December 2011; revised 27 July 2012; accepted 28 October 2012)

Mandarin Chinese is based on characters which are syllabic in nature and morphological in meaning. All spoken languages have syllabiotactic rules which govern the construction of syllables and their allowed sequences. These constraints are not as restrictive as those learned from word sequences, but they can provide additional useful linguistic information. Hence, it is possible to improve speech recognition performance by appropriately combining these two types of constraints. For the Chinese language considered in this paper, character level language models (LMs) can be used as a first level approximation to allowed syllable sequences. To test this idea, word and character level *n*-gram LMs were trained on 2.8 billion words (equivalent to 4.3 billion characters) of texts from a wide collection of text sources. Both hypothesis and model based combination techniques were investigated to combine word and character level LMs. Significant character error rate reductions up to 7.3% relative were obtained on a state-of-the-art Mandarin Chinese broadcast audio recognition task using an adapted history dependent multi-level LM that performs a log-linearly combination of character and word level LMs. This supports the hypothesis that character or syllable sequence models are useful for improving Mandarin speech recognition performance.
 [http://dx.doi.org/10.1121/1.4768800]

PACS number(s): 43.72.Ne [MHJ] Pages: 519–528

I. INTRODUCTION

The Chinese language is based on characters which are syllabic in nature and morphological in meaning.[1] There are many characters which have the same pronunciations. The presence of these homographs makes it difficult to know what the spoken word is without context or what the character sequence is. This ambiguity is demonstrated, for example, by the practice of signing that Chinese people use to show how to spell their names when they are introduced. By showing their names in written form using characters, the listener knows how their names are spelled, otherwise it would be difficult to determine which characters constituted the spelling of the name. As Chinese characters have a broad meaning in themselves, knowing the character sequence in a name, named entity, or sentence can evoke the underlying meaning.

Written Chinese has no word boundaries marked by spaces or other symbols, making the task of finding the correct word sequence difficult. The reader must infer the word boundaries from the context. In this process ambiguity can occur, even among native speakers. In tests of word boundary marking by native speakers, agreement on the boundary locations occurs approximately 75% of the time.[2] Therefore, the character error rate (CER) is the commonly used evaluation metric for state-of-the-art Mandarin speech recognition systems.[3–5]

All languages have constrained syllable constructions and syllable sequence rules which enhance intelligibility.[6] These phonological and pragmatic constraints can be exploited for Chinese speech recognition. Syllable level constraints are not as restrictive as word sequences, but they can provide additional useful linguistic information. It is thus possible to leverage both types of constraints by appropriately combining the two. It is hoped that the combined system will reflect the Chinese language better and improve speech recognition performance. This paper examines this proposition.

In order to implement Mandarin syllable sequence modeling, it is necessary to be able to estimate the probability for each possible syllable sequence. In a speech recognition system, the resulting statistical models assign the prior probability of a syllable sequence. They are commonly referred to as language models (LMs) at the syllable level. One issue with this method is that syllable segmented and labeled Chinese speech is expensive to produce and generally unavailable in large quantities. An alternative is to use character level LMs as an indirect way of modeling syllable sequences.[7–12]

This approach is motivated by two factors. First, ordinary character level texts are already available in the large quantities required for LM training. They are far more accessible than syllable level annotated texts. Second, the maximum onset principle, which allows onset extension by attaching

[a)]Author to whom correspondence should be addressed. Electronic mail: xl207@eng.cam.ac.uk

the previous syllable's final consonants to the following one, is mostly blocked for Mandarin. Mandarin Chinese has no consonant clusters and the syllable structure is

$$(C)V(N)(R),$$

where C is an optional consonant, V is a vowel, and N is an optional final nasal consonant /n/ or /ng/ and R is an optional rhotic coda /r/. The nasal /ng/ can only be a final consonant. As a result the majority of Mandarin syllables should have only one segmentation and phonetic realization associated with their natural character boundaries in continuous speech, except for those ending in /n/ or /r/ followed by others beginning with a vowel. Phonotactically this is very different from other syllabic languages, for example, English, which uses complex prevocallic and postvocallic consonant clusters and thus onset extension can occur more often.

Language models play a crucial role in automatic speech recognition (ASR) systems. They assign the prior probability to a hypothesized word sequence in a speech recognizer. Back-off n-gram models remain the dominant language modeling approach for state-of-art ASR tasks.[13] In these systems, LMs are often, as in this paper, constructed by combining component n-gram models trained on a diverse collection of text sources prior to probability interpolation in the form of a mixture model.[14–16] Individual data sources are considered to be more appropriate for different tasks or genres, for example, broadcast news or conversational telephone speech. The interpolation weights indicate the "usefulness" of each source for a particular task. To further improve robustness to varying styles or tasks, unsupervised test-set adaptation to, for example, a particular broadcast show, may be used.[10,17–21]

As directly adapting n-gram word probabilities is impractical on limited amounts of data, conventional LM adaptation schemes only involve updating the context independent, linear interpolation weights associated with component models.[14,16] However, this approach can only adapt LMs to a particular genre, epoch or other higher level attributes. Local factors that determine the contribution of sources on a context dependent basis, including the modeling resolution, generalization ability, topic coverage, and style, are poorly modeled. To handle this issue, context dependent LM interpolation and adaptation can be used.[10,17,20,21]

One key issue with incorporating character level LMs into a word based recognition system is the appropriate form of combination technique to use. The LIMSI-CNRS speech research group recently constructed a Mandarin character based recognition system.[11] They experimented with three techniques: (1) using character based recognition alone, (2) ROVER (Ref. 22) based character level hypothesis combination with a word based system, and (3) using character level LMs to rescore hypothesis candidates previously generated by a word based system as an implicit form of combination. However, these methods were unsuccessful in improving the character error rate (CER) performance against the word based standard recognition system.

In this paper two major categories of techniques are investigated: hypothesis level, and model level combination. The former exploits the consensus among component systems using voting as well as confidence measures, as used in ROVER (Ref. 22) and confusion network combination.[23] The second category uses linear or log-linear model combination to combine standard word and character based LMs at the distribution level instead of the hypothesis level. In machine learning, these two methods are often in turn referred to as a mixture of experts (MoE), equivalent to a probabilistic *union* that improves generalization, and a product of experts (PoE), a probabilistic *intersection* that increases discrimination.[24] A combination of these two methods can also be used to leverage the strengths of both.

As the underlying LM configuration becomes more complex, extensive software changes to special purpose tools, for example, SRILM or HTK toolkit,[25,26] is often required. An alternative approach to combine and adapt LMs is to use weighted finite state transducers (WFSTs).[10,27–30] As this approach is entirely based on a set of well-defined automaton operations, minimal changes to decoding tools are required. It is highly flexible and can be used for a wide range of combination configurations. It not only supports the use of global, context independent weights in LM combination, but also a more general case when context dependent weights are employed. Thus LM adaptation using history context dependent interpolation and adaptation can be conveniently implemented.

The precise nature of component language models determines which of the two combination schemes is more appropriate. Since character LMs represent additional syllable sequence constraints that word based models cannot provide, a log-linear combination is thought to be more appropriate rather than a linear combination for this purpose. This hypothesis is confirmed by experimental results presented later in this paper. A carefully constructed WFST based log-linear model combination consistently gave the lowest character error rate among all the schemes investigated in the paper.

The rest of the paper is organized as follows. First, ROVER based hypothesis level combination is reviewed in section II. Model level combination schemes to incorporate character sequence level information into word based LMs are then introduced in Sec. III. Generic WFST based LM combination methods are proposed in Sec. IV. Context dependent LM adaptation is then presented in Sec. V. An efficient on-the-fly WFST decoding approach is proposed in Sec. VI. Experimental results on a state-of-the-art Mandarin broadcast speech transcription task using the CU-HTK Mandarin large vocabulary continuous speech recognition system are presented and analyzed in Sec. VII, together with the implications which can be drawn from them. Section VIII gives the conclusion and suggests possible future work.

II. HYPOTHESIS LEVEL SYSTEM COMBINATION

One commonly used form of hypothesis level combination is ROVER.[22] Hypotheses from a total of S component systems are iteratively aligned to create word transition networks. An interpolation between voting counts and confidence scores is then used to find the optimal word sequence within the network. For any set of confusions in the network this is given by

$$\hat{w} = \arg\max_{w_s}\left\{\alpha\frac{N_{1:S}(w_s)}{S} + (1-\alpha)c_w^{(s)}\right\}, \quad (1)$$

where $N_{1:S}(w_s)$ is number of systems that output word w_s, $c_w^{(s)}$ is the confidence score assigned by the sth system, and α is a tunable parameter to balance the contribution between voting counts and confidence scores. When component systems use different word segmentation schemes, a direct combination between their outputs is problematic, for example, in Chinese where different character to word segmentations are used. Hence, for the Mandarin speech recognition tasks considered here, the most successful approach is to perform a character level combination,[3–5,11] as is also considered in this paper. This requires the mapping of outputs from a standard word based system to sub-word, character level. The confidence score of each word is assigned to each character it contains. One major issue with character level ROVER is it does not preserve a consistent character to word segmentation in the final outputs, and thus affects further processing of the recognition outputs, for example, in speech translation tasks.[4] In general, hypothesis level combination methods such as ROVER also require the error rate performance of the component sub-systems to be close in order to be effective in combination.[4,11]

III. LANGUAGE MODEL COMBINATION

As discussed in Sec. I, model level combination techniques may be used to incorporate character sequence level constraints into a word based recognition system. These techniques can be further classified into MoE based linear,[15,16] and PoE based log-linear model combination.[24,31,32] While each has its own characteristics, it is possible to leverage from the strength of both methods. The rest of this section discusses various forms of model level combination schemes.

A. Linear model combination

As a *union* of all the individual probabilistic experts, linear model combination tends to give a broader distribution than individual components alone. Hence, this form of model combination may help overcome the sparsity issue when training individual component models and thus improve generalization. Let w_i denote the ith word of a L word long sequence $\mathcal{W} = \langle w_1, w_2, \ldots, w_i, \ldots, w_L\rangle$. The LM log-probability for the complete word sequence is given by

$$\ln P(\mathcal{W}) = \sum_{i=1}^{L} \ln\left(\sum_{m=1}^{M}\lambda_m P_m(w_i|h_i^{n-1})\right), \quad (2)$$

where h_i^{n-1} represents the ith word's history of at most $n-1$ words, $\langle w_{i-n+1}, \ldots, w_{i-1}\rangle$, and λ_m is the global, context independent weight for the mth component under a positive and sum-to-one constraint. These weights indicate the "usefulness" of each source for a particular task. To reduce the mismatch relative to the target domain, these weights may be tuned to minimize the perplexity on held-out data.

B. Log-linear model combination

In contrast, log-linear interpolation provides an *intersection* of individual probabilistic experts. It yields a high likelihood only when all component models agree. This form of model combination exploits the consensus among product experts. Hypotheses with very different log-likelihood ranking among component models will be penalized. For the above example, the log-linearly interpolated LM probability on word sequence level is

$$\ln P(\mathcal{W}) = \sum_{i=1}^{L}\sum_{m=1}^{M}\lambda_m \ln P_m(w_i|h_i^{n-1}) - \ln(Z), \quad (3)$$

where Z is a normalization term to ensure the sequence level interpolated probability to be a valid distribution. Its exact computation for general forms of log-linear models is non-trivial. Analytical solutions are available only for certain forms of density functions used for component models. It may be ignored when the interpolation is performed at complete word sequence level under a discriminative framework, for example, *maximum entropy* models and *logistic regression*,[31,32] as implemented in this paper. Here, λ_m is the context free log-linear weight for the mth component. They are no longer subject to a positive and sum-to-one constraint. They may be optimized under a discriminative framework as in maximum entropy models.[31] For a simple two way log-linear combination between word and character based LMs considered in this paper, these weights are fixed as equal in all experiments.

C. Multi-level model combination

As discussed in Sec. I, the precise nature of component language models determines which of the two combination schemes is more appropriate. For example, when building word level LMs, in order to improve context coverage and generalization, a linear interpolation between component LMs trained over a diverse set of text sources can be used. When introducing additional sub-word, character level linguistic constraints to increase discrimination, word and character level LMs can be log-linearly combined.[8,10] In order to achieve a good balance between generalization and discrimination, it is also possible to leverage both forms of combination using a product between mixtures of experts.

IV. LANGUAGE MODEL COMBINATION USING WEIGHTED FINITE STATE TRANSDUCERS

As discussed in Sec. I, in current ASR systems language models are often constructed by training n-gram component models[13] from a set of diverse sources representing data from different genre, epoch, or other higher level attributes. In order to incorporate more linguistic constraints, it is also possible to train and combine LMs that model different unit sequences, for example, syllables and words.[8,10] Interpolated LMs with context independent weights are normally constructed using special purpose tools, for example, the SRILM or HTK toolkits.[25,26] In order to capture local variations in modeling resolution, generalization, topics, and style among

component LMs, history context dependent LM interpolation and adaptation can be used.[10,20,21] These techniques often require extensive software changes.

An alternative approach considered in this paper is to combine and adapt LMs using WFSTs.[10,27–30] As this approach is entirely based on well-defined WFST operations, minimal changes to the decoding tools are required. It is highly flexible and can be used for a wide range of combination configurations. It not only supports the use of global, context independent weights in LM combination, but also a more general case when context dependent weights are employed. Thus LM adaptation using history context dependent interpolation can be conveniently implemented.

A WFST is a finite state machine that associates weights such as probabilities, durations, penalties, or any other quantity that accumulates linearly along paths within a directed graph, to each pair of input and output symbol sequences. A set of classic finite automaton operations to combine, optimize, and compact WFSTs during search are available. Many types of modeling information used in speech recognition systems, such as HMM topology, lexicon, and n-gram LMs, involve a stochastic finite-state mapping between symbol sequences. WFSTs provide a generic and well-defined framework to represent and manipulate them. Unless otherwise stated, tropical semi-ring based WFSTs (Refs. 27–30) are used in this paper.

More precisely, n-gram LMs can be represented by weighted finite state acceptors. These are special cases of WFSTs when the input and output symbol sequences are identical. Consider two simple back-off bigram language model fragments that are associated with a three word sub-vocabulary $\{A, B, C\}$ and trained on two different text sources. Their WFST representations, $\mathcal{L}_G^{(1)}$ and $\mathcal{L}_G^{(2)}$, are shown in Figs. 1(a) and 1(b). In both transducers, n-gram log probabilities appear as negated arc weights. The 1-gram back-off weights are represented by non-emitting epsilon arcs without output symbols, as marked with "<e>" in the figure.

Assuming component LMs model the same type of symbol sequences, for example, words, the WFST representation of the linearly combined LM given in Eq. (2) of Sec. III A can be derived using a component level *composition* between the n-gram and interpolation weight transducers prior to a final log semi-ring based n-gram level *union* operation. Hence,

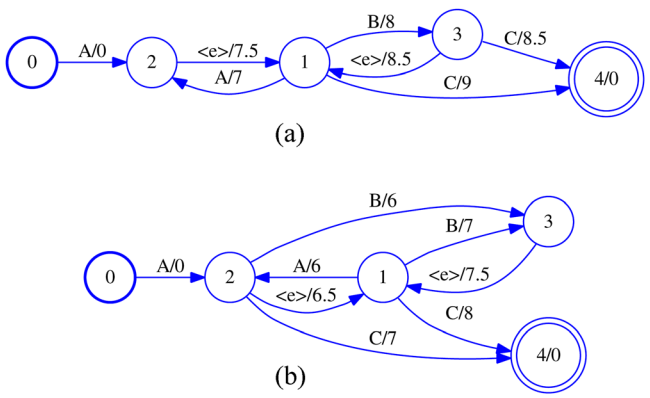

FIG. 1. (Color online) WFST representation of two 2-g back-off LMs.

$$\mathcal{L} = \left(\mathcal{L}_G^{(1)} \circ \mathcal{L}_\phi^{(1)}\right) \cup \cdots \left(\mathcal{L}_G^{(m)} \circ \mathcal{L}_\phi^{(m)}\right) \cup \cdots \left(\mathcal{L}_G^{(M)} \circ \mathcal{L}_\phi^{(M)}\right),$$

where $\mathcal{L}_G^{(m)}$ is the n-gram model transducer, and $\mathcal{L}_\phi^{(m)}$ the interpolation weight transducer for the mth component. Here "$\circ$" and "$\cup$" denote the composition and union operations, respectively. Taking the component LMs of Figs. 1(a) and 1(b) as examples, their context independent interpolation weights, $\lambda_1 = 0.3$, $\lambda_2 = 0.7$ (1.220 and 0.360 as negated natural log) may be represented by the two transducers in Fig. 2. They proportionately reflect the probability contribution from the two component LMs on a context independent basis, as the second model has more 2-grams than the first one. In order to improve the efficiency of the combined LM WFST during search, it may be further compressed via standard WFST *determinization* and *minimization* operations.

It is also possible to linearly combine LMs modeling sequences of different linguistic units, for example, syllables, or characters, and words. In order to have compatible transducer symbols during combination, the syllable, or character level component WFSTs must be first *composed* with a lexicon transducer, which provides the character to word mapping, and then *projected* onto the word level. Take the three word vocabulary $\{A, B, C\}$ associated with the two 2-gram LMs shown in Figs. 1(a) and 1(b) as an example. Assuming word A has only a single character $a1$, while word B is made up of two characters $b1$ and $b2$, and C a three character word containing $c1$, $c2$ and $c3$, the WFST representation of such lexicon is shown in Fig. 3 below.

In contrast, the log-linear model combination given in Eq. (3) of Sec. III B may be efficiently implemented using a sequence of WFST *composition* operations between component n-gram model transducers after an *arithmetic scaling* of arc costs by their respective log-linear weights. This is given by

$$\mathcal{L} = \left(\mathcal{L}_G^{(1)} \times \lambda_1\right) \circ \cdots \left(\mathcal{L}_G^{(m)} \times \lambda_2\right) \circ \cdots \left(\mathcal{L}_G^{(M)} \times \lambda_M\right).$$

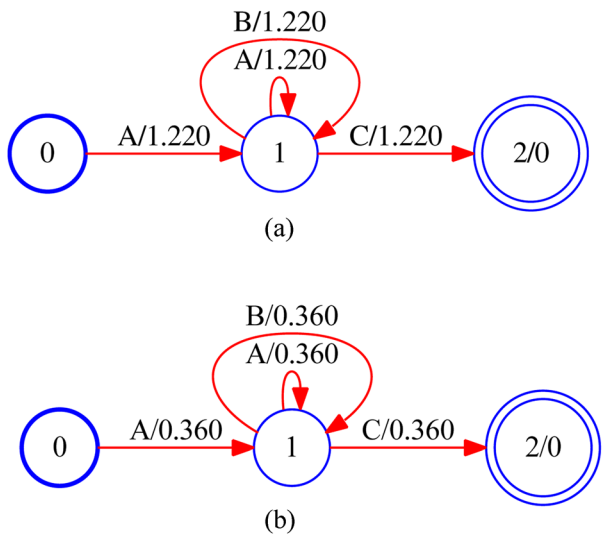

FIG. 2. (Color online) WFST representation of context independent linear interpolation weights for component LMs in Figs. 1(a) and 1(b).

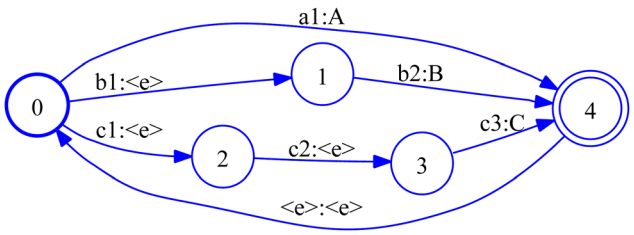

FIG. 3. (Color online) WFST representation of a lexicon which provides a mapping from character sequences to words.

The multi-level combination described in Sec. III C may be represented as *composition* between *union*-ed LM WFSTs.

V. LANGUAGE MODEL ADAPTATION

In order to improve robustness to varying style or tasks, unsupervised test-time LM adaptation to a particular broadcast show, for example, may be used.[18,19] As directly adapting n-gram probabilities is impractical on limited amounts of data, standard LM adaptation schemes only involve updating the context independent, linear interpolation weights of Eq. (2).[14,16]

However, this approach can only adapt LMs to a particular genre, epoch or other higher level attributes. Local factors that determine the contribution of sources on a context dependent basis, such as modeling resolution, generalization, topic coverage, and style, are poorly modeled. Take the 2-gram distribution $P(C|B)$ in Fig. 1 as an example, the first component LM of Fig. 1(a) gives a 2-gram log-probability of -8.5, while a lower score of -15.5 is assigned by the second one via a back-off path in Fig. 1(b). In this case the probability contribution from the two component LMs clearly contradicts the assignment of context independent interpolation weights of 0.3 and 0.7 in Fig. 2. To handle this issue, context dependent LM interpolation and adaptation can be used.[10,17,20,21] A set of discrete context dependent back-off weights are used to dynamically adjust the contribution from component LMs. Thus Eq. (2) is extended to

$$\ln P(\mathcal{W}) = \sum_{i=1}^{L} \ln\left(\sum_{m=1}^{M} \phi_m(h_i^{n-1}) P_m(w_i|h_i^{n-1})\right), \quad (4)$$

where $\phi_m(h_i^{n-1})$ is the mth component's context dependent weight for history h_i^{n-1}.

Both maximum likelihood and discriminative schemes are available to robustly estimate context dependent interpolation weights.[20,21] Considering the maximum *a posteriori* based adaptation as an example, this is given by

$$\hat{\phi}_m(h_i^{n-1}) = \frac{\mathcal{C}_m^{\mathrm{ML}}(h_i^{n-1}) + \tau \hat{\phi}_m(h_i^{n-2})}{\sum_m \mathcal{C}_m^{\mathrm{ML}}(h_i^{n-1}) + \tau}, \quad (5)$$

where $\mathcal{C}_m^{\mathrm{ML}}(h_i^{n-1})$ is maximum likelihood (ML) counts for history context h_i^{n-1}, and τ controls the contribution from a hierarchical prior, $\hat{\phi}_m(h_i^{n-2})$, before log-linearly combined with a high resolution training data prior.[21]

To improve robustness to the supervision quality, it is possible to use confidence score weighted sufficient statistics when estimating context independent, and dependent interpolation weights.[10,33] The log-likelihood in Eq. (4) is thus modified as

$$\ln \breve{P}(\mathcal{W}) = \sum_{i=1}^{L} c_i \ln\left(\sum_{m=1}^{M} \phi_m(h_i^{n-1}) P_m(w_i|h_i^{n-1})\right), \quad (6)$$

where c_i is the confidence score for word w_i. By default, when using a null history the above simplifies to confidence score based adaptation of context independent weights in Eq. (2). To further improve robustness during context dependent LM adaptation, it is also possible to impose a count cut-off for different histories, for example, the average word level confidence score computed over the supervision hypotheses. Contexts which do not have sufficient counts will be pruned in weight estimation.

The WFST representation of Sec. III A also holds for LMs constructed using context dependent interpolation weights. The difference between context independent and dependent LM interpolation lies in the precise nature of the weight transducers. Again taking the two component LMs of Figs. 1(a) and 1(b) as examples, the WFST representation of their context dependent interpolation weights are shown in Figs. 4(a) and 4(b). As is shown in the figure, when the history varies, more flexibility is allowed in component LM weighting than for the context independent case of Fig. 2. For the 2-gram $P(C|B)$, a duly higher weight of 0.8 (0.219 as negated natural log) is now assigned to the first component LM.

VI. IMPLEMENTATION ISSUES

In this section implementation issues that can affect the performance of multi-level LM combination and adaptation are discussed.

A. Decoding with multi-level LMs

When character and word based LMs are combined to form a multi-level LM, two decoding strategies can be considered. The first starts from a standard word based recognition and lattice generation stage. A word level LM constructed using a linear combination between component LMs trained

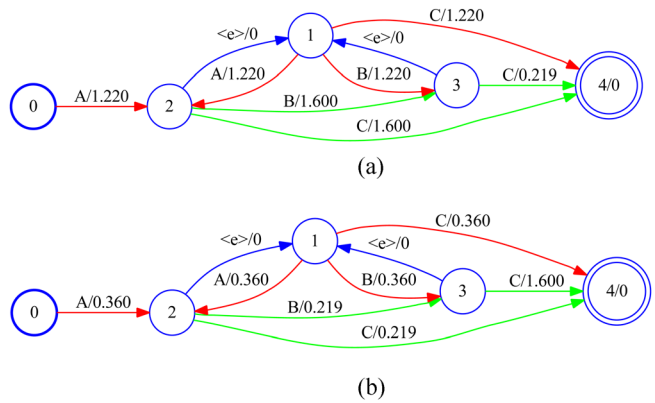

FIG. 4. (Color online) WFST representation of context dependent linear interpolation weights of component LMs shown in Figs. 1(a) and 1(b).

on different text sources is used in decoding. This is followed by expanding the resulting word lattices into sub-word, i.e., character level lattices, via a *composition* with the *inverse* of the lexicon transducer of Fig. 3 (obtained by swapping input and output symbols), which provides word to character sequence mapping. A final composition with a character level LM leads to a log-linear combination between the two and produces the most likely hypothesis.

In contrast, the second option starts from character recognition using a linearly interpolated multi-source character level LM. This is followed by a transduction of the resulting lattices from the sub-word, character level, to the word level by a composition with the lexicon WFST shown in Fig. 3. The character-level LM is then log-linearly combined with a word-level language model via a composition operation to produce the best hypothesis. As discussed in Sec. I, character-level LMs provide weaker linguistic constraint than word-based LMs. As a result, when using this decoding approach, the character lattices generated in the initial stage are prone to have higher lattice oracle error rates[34] due to the necessity of pruning. The increase in lattice error rate can also affect the final recognition performance when the character LM is further combined with a word based LM, as is confirmed in the experimental results presented later in this paper.

B. On-the-fly WFST network expansion

When using context dependent interpolation in LM adaptation and decoding, every broadcast show or snippet, for example, may have its own set of interpolation weights. When modeling a large number of contexts using the transducer topology of Fig. 4, the composition operation between component n-gram and their weight WFSTs can lead to a significant network expansion. This is highly inefficient and makes the subsequent network compression operations very expensive. The same issue exists during the composition between component n-gram transducers in the log-linear combination of Sec. III B. Hence, it is preferable to dynamically perform the composition, union, and compression operations in one single on-the-fly step. Related approaches have been previously shown to be effective for the composition between one single back-off n-gram LM and a lexicon.[35] The basic idea is to only create a new path on request during search, if and only if it carries context information different from others. For context dependent adaptation, the LM state associated with context history is *jointly* determined by component n-gram models and interpolation weights in the form of a context pair. Using this on-the-fly lattice expansion algorithm, redundant paths representing unused lower order back-off distributions will be automatically filtered out.

VII. EXPERIMENTS AND RESULTS

A. Baseline system description

The CU-HTK Mandarin ASR system[36] was used to evaluate the performance of multi-level language models. Two versions of the speech recognition system were used. One used words as the basic recognition unit (as is usually done) and the other used characters. This was done to determine whether character based recognition, followed by applying the word level LM and then breaking the words into characters again would be better than word based recognition with word level LM's, then breaking the words into characters and applying character level LM constraints. In theory the results should be comparable, but in practice they are not due to the increase in lattice error rate when performing character recognition first, as discussed earlier in Sec. VI A.

1. Acoustic model training

Context dependent phone models, for example, triphones, are the most common acoustic unit used in state-of-the-art speech recognition systems. The CU-HTK Mandarin system uses 46 base phones to which five tones (four tones and the neutral tone) are added to the vowels. The result is 124 tonal phones. A recognition dictionary of 63 k words with an average of 1.02 pronunciations per word is used. There are 52 k multiple character Chinese words, 6 k single character words, and 5 k frequent English words.

Speech data for acoustic model training consisted of 1673 hours of broadcast news (BN) and broadcast conversation (BC) data released under the DARPA GALE program. Human transcribed speech is first force-aligned with a standard word based speech recognition system using a dictionary to assign phone sequences to each word. ML and then later discriminative training techniques,[37] were used to train three state left to right Gaussian mixture hidden Markov (HMM) based triphone models.

To handle the data sparsity issue, phonetic decision trees[38] were used in HMM state tying for the CU-HTK Mandarin ASR system. MLLR speaker adapted,[39] gender dependent, MPE discriminatively trained[37] cross-word triphone HMM acoustic models were used in decoding. The acoustic models were trained on 39 dimensional HLDA (Refs. 40 and 41) projected PLP (Ref. 42) features with cepstral mean normalization and appended log pitch parameters.

2. Language model training

A total of 4.3 billion characters from 27 text sources were used in LM training. After a left to right longest first based character to word segmentation, 2.8 billion words of texts in total were used to train a word level interpolated 4-gram back-off LM.[13] Information on corpus size, cut-off settings, and smoothing or discounting schemes for text sources are given in Table I. The context independent linear interpolation weights of the word level 4-gram LM were perplexity tuned on the reference transcription of the GALE development set dev07 combined with two additional held out sets, bn06 of 3.4 h of BN data, and bc05 of 2.5 h of BC data. These are shown in the fifth column of Table I. A similar rank ordering of sources weights was also obtained for the character level language model. Due to data sparsity, only 5-gram and 6-gram LMs were built for character level models. Their cut-off settings are shown in brackets. For data sources that are closer in genre to the test data, minimum cut-offs and modified KN smoothing[43] were used. These include two audio transcription sources, bcm and bnm, and additional

TABLE I. Text source size, 2/3/4-gram cut-off settings, smoothing scheme used in training (5/6-gram cut-offs for character level LMs in brackets), perplexity tuned interpolation weights using held-out data.

Comp. LM	#Char. (M)	#Word (M)	Train config.	Intplt. weight
bcm	14.26	9.21	kn/111(11)	0.260058
bnm	12.29	7.41	kn/111(11)	0.147834
gigaxin	483.65	362.74	kn/112(22)	0.132539
phoenix	144.57	91.38	kn/112(22)	0.107920
gigacna	891.13	604.98	gt/123(33)	0.072665
voarfa	63.54	35.31	kn/112(22)	0.072299
ibmsina2	382.34	253.59	kn/112(22)	0.055601
bbndata	301.39	186.3	kn/112(22)	0.046213
galeweb	556.41	390.8	kn/122(22)	0.045918
agilece	336.78	204.5	kn/112(22)	0.031497
ntdtv	36.44	24.75	kn/112(22)	0.010216
ibmsina1	78.39	51.89	kn/112(22)	0.003814
papersjing	197.75	135.69	kn/112(22)	0.003220
tdt4	2.98	1.76	kn/112(22)	0.003005
tdt2+3	15.87	9.35	kn/112(22)	0.001689
xinhuachina	105.88	76.57	kn/112(22)	0.001587
sriwebconv	163.16	114.6	kn/112(22)	0.001081
gigaafp	40.28	27.24	kn/112(22)	0.000770
cctvcnr	47.31	29.59	kn/112(22)	0.000751
hub4m	0.38	0.22	kn/111(11)	0.000533
chradio	91.55	54.86	kn/112(22)	0.000468
papersyue	52.48	34.14	kn/112(22)	0.000275
gigalhzb	29.16	19.73	kn/112(22)	0.000019
papershu	50.67	34.85	kn/112(22)	0.000012
pdaily	114.51	68.89	kn/112(22)	0.000012
papersning	51.99	33.9	kn/112(22)	0.000006
dongailbo	12.82	8.02	kn/112(22)	0.000000

web data from major TV channels such as Phoenix TV and Voice of America. For example, a cut-off of "111" were used for the bnm and bcm sources. This setting implies that there has to be one occurrence of any bigram, trigram or fourgram if any of them were to be retained. For the largest corpora of newswire genre and Taiwanese origin, giga-cna, more aggressive cut-offs and good Turing discounting were used.

3. Character recognition system

A character based speech recognition system was also implemented by using a character dictionary with multiple pronunciations and a character level LM sequence model during decoding. There are 6 k characters in the dictionary and 5 k frequent English words with approximately 1.1 pronunciations per character. The resulting character lattices were then converted into word lattices using a dictionary mapping character sequences to words before a word level LM is applied, as previously described in Sec. VI A. The word level LM, which provides more constraints, is used to refine the ranking of hypotheses. Finally the words are broken into characters and a character error rate is determined.

4. Test sets

Three Mandarin Chinese broadcast speech test sets of mixed BN and BC genre were used in the experiments: 2.6 h dev07, 1 h dev08, and 2.6 h p2ns. For all results presented in this paper, statistical significance tests were performed at a significance level $\alpha = 0.05$.

B. Performance of character and word level LMs

The model sizes of the final interpolated 4-gram word and 6-gram character level LMs are shown in Table II. The total log-probability scores on the combined held out data reference bn06 + bc05 + dev07 assigned by these two LMs are shown in the sixth column of the table. On average the word based system produces approximately 1.5 characters per word. Hence, a 6-gram character based LM would be appropriate to compare with a 4-gram word level baseline. As expected, with a stronger constraint, the word based model gave a better log-probability than the character based one. The perplexity metric is commonly used to measure the predictive power of LMs. However, as these two LMs considered here model different linguistic units, a direct comparison between word and sub-word level perplexity scores is not meaningful. One possible solution is to approximate the sub-word, or character, level perplexity for the word based LM. The number of sub-word units, instead of the word level sequence length, is used in perplexity computation. This approximate character level perplexity score is in the last column of the first line for the word based system. Consistent with the trend observed on log-likelihood, the word based model also has a lower approximated character level perplexity by 9% relative.

A similar error rate performance difference between the 4-gram word baseline and the 6-gram character level LM can also be found in the second and fifth line of Table III. The word level LM outperformed the character based LM by statistically significant 0.7%–1.1% absolute (7.3%–11.2% relative) across all test sets. The CER performance of various other systems on dev07, dev08, and p2ns is shown in Table III. The performance of the 3-gram word based model, the 4-gram and 5-gram character based models are shown in the first, third, and fourth lines of the table. For the word based LM, increasing the *n*-gram context length from 3-gram to 4-gram gave further CER reductions of 0.2%–0.4% absolute (2.0%–4.0% relative) on all test sets. In contrast, the relative gains on the character based system, for example, between 5-gram to 6-gram models, are only 0.1%–0.2% on dev07 and p2ns.

The weaker constraints of character level LMs can also be shown by measuring the oracle error rate of the lattices it produces. In this process, lattices from the word 4-gram and character 6-gram based systems are examined to test whether

TABLE II. Model sizes of word and character level LMs, their total log-probability and character level perplexity scores on the combined reference of bn06 + bc05 + dev07.

	Model size (M)						
LM	2g	3g	4g	5g	6g	Log prob.	Char. PPlex
word	60	228	56	—	—	−511524	25.61
char.	10	148	111	130	122	−524473	27.91

TABLE III. 1-best CER performance of various LMs on dev07, dev08, and p2ns. ⊕ denotes hypothesis level ROVER and "○" denotes WFST composition operations.

	CER%		
System	dev07	dev08	p2ns
w.3 g	10.0	10.0	9.8
w.4 g	9.8	9.6	9.6
c.4 g	11.5	10.4	10.8
c.5 g	11.1	10.3	10.6
c.6 g	10.9	10.3	10.5
c.6 g ○ w.4 g	10.5	9.9	10.1
w.4 g ⊕ c.4 g	10.4	9.8	10.0
w.4 g ⊕ c.5 g	10.2	9.7	9.9
w.4 g ⊕ c.6 g	10.2	9.6	9.8
w.4 g ○ c.4 g	9.7	9.4	9.5
w.4 g ○ c.5 g	9.7	9.4	9.4
w.4 g ○ c.6 g	9.7	9.4	9.4

the correct characters are present. This measure would serve as a performance upper bound for the underlying recognition system. As discussed in Sec. III, for the baseline system that uses a word level LM, this requires expanding the lattices onto the character level, via a composition with the inverse of the lexicon transducer of Fig. 3. The lattice oracle CER performance of these two systems are shown in Table IV. Again, the word level LM, which provides a stronger constraint during search, consistently produced lower lattice error rates.

As discussed in Secs. I and VI A, the higher lattice oracle error rates that resulted from the weak constraints of character level LMs can also affect the final recognition performance when the they are further combined with a word based LM. This is confirmed by the error rate performance of the "c.6g ○ w.4g" system shown in the sixth line of Table III, derived by a sequence of composition operations between the character level lattices, the lexicon WFST shown in Fig. 3, and finally the baseline word level 4-gram LM WFST, as discussed in Sec. VI A. These effectively provide a log-linear combination between the 6-gram character level and 4-gram word level LMs. Despite using both word and character level constraints, this system is consistently outperformed by using the 4-gram word baseline LM alone across all test sets. In particular, a statistically significant increase in CER of 0.7% and 0.5% absolute were observed on dev07 and p2ns, respectively. These results suggest that using character level LMs to generate initial recognition hypotheses is unsuitable due to its weak constraints. The pruning during recognition seems to remove the correct character sequences and has increased the number of search errors. Instead the sub-word level constraints they provide should be applied in later recognition stages, for example, after standard word level recognition is performed.

TABLE IV. Oracle character level error rate for lattices generated using word or character level LMs on dev07, dev08, and p2ns.

	Lattice Oracle CER%		
LM	dev07	dev08	p2ns
word	1.71	1.70	1.89
char.	2.04	1.89	2.02

C. Performance of ROVER combination

The rest of Table III shows the performance of various other methods combining information from word and character LMs based recognition systems. Performance of three character level ROVER systems combining the 4-gram word LM based system with various character LM based ones are shown in the third section of Table III. The best ROVER configuration is between the 4-gram word and the 6-gram character LM based systems, but it is still outperformed by the 4-gram word level LM by 0.2%–0.4% absolute on dev07 and p2ns. As previously discussed in Sec. II, hypothesis level combination methods such as ROVER require that the error rate of component systems be similar in order to be effective in combination. However, this condition is not satisfied given the significant performance difference between the word and character LM based systems of Table III. Furthermore, as there are only two component systems used in ROVER, the combination decision is purely based on confidence scores as voting now has no effect. Poor confidence scores generated by the character LM based systems can introduce additional errors in combination.

D. Performance of multi-level LMs

The performance of model level combination techniques discussed in Sec. III were then evaluated. Based on the lattice oracle error rates shown in Table IV, the lattices produced by the word based LM are used in a later rescoring stage where various combined LMs are used. As discussed previously in Sec. III, the precise nature of the sub-word, character level LM determines whether a linear or log-linear model level combination would be appropriate to use. A MoE model uses linear interpolation between the word and sub-word *n*-gram models. Rather than increasing the combined model's discrimination, it broadens the underlying statistical distribution and improves its generalization. This was confirmed by the error rate performance of using a linear model combination between character and word level LMs, which was consistently outperformed in practice by the standard word based 4-gram LM. Exhaustive tuning of linear interpolation weights on dev08, for example, between the 4-gram word and 6-gram character level LMs showed that the best linear weighting is to use the word based LM's probability only.

In contrast, a PoE model,[24] which uses a log-linear model combination, tends to sharpen the underlying distribution and increase its power of discrimination. As character level LMs provide additional sub-word, syllable level information, it is expected that a log-linear, rather than linear, interpolation would be more suitable for incorporating character level constraints. An equally weighted log-linear interpolation between the 4-gram word and character based LMs using a WFST composition operation described in Sec. III B gave consistent CER reductions of 0.1%–0.2% on all test

sets over the word based standard system. These are shown in the bottom section of Table III. These results suggest including a character level LM provides additional sub-word level linguistic constraints and increased discrimination for Mandarin speech recognition.

E. Performance of adapted multi-level LMs

As discussed in Sec. III C, when using the above multi-level LM combination frame-work, in order to improve context coverage and generalization, both the word and character level LMs are constructed by a linear combination of component *n*-gram LMs trained over a diverse set of text sources shown in Table I, prior to a log-linear combination between word and character level mixture models. In order to improve robustness to varying styles or tasks, unsupervised LM adaptation can be used, as discussed in Sec. V. Hence, it would be interesting to investigate the gains from the log-linear combination based multi-level LM shown in Table III are retained after LM adaptation. This is evaluated and shown in Table V using confusion network (CN) (Ref. 23) decoding at the lattice generation stage of the CU-HTK Mandarin ASR system. The CN outputs and associated confidence scores generated by the unadapted baseline LM, and the confidence score weighted log-likelihood criterion in Eq. (6) presented in Sec. V were used to adapt various LMs.

The CER performance of the baseline word level 4-gram LM is shown in the first line of Table V. The performance of using the character level 6-gram model is shown in the second line of Table V. Consistent with the 1-best decoding results shown in Table III, with a stronger constraint, the word level 4-gram baseline significantly outperformed the character 6-gram LM by 0.4%–1.2% absolute. When combining character and word level constraints using an equally weighted log-linear interpolation of Eq. (3) and the WFST representation presented in Sec. III B, consistent performance improvements were obtained over the word level baseline. This is shown in the third line of Table V. It gave a statistically significant CER reductions of 0.5%–0.3% on dev08 and p2ns, respectively.

The second section of Table V shows the performance of three adapted LMs using the estimation scheme and WFST representation given in Secs. III and V. The 1-best outputs from the un-adapted word level baseline system was used as the supervision in perplexity based LM adaptation. Standard LM adaptation using context independent interpolation weights gave CER reductions of 0.1%–0.3% absolute across three test sets (fourth line of Table V). Using three-word history based context dependent adaptation of Eq. (4), further CER improvements of 0.1% absolute were obtained for all test sets (fifth line of Table V). Context dependent adaptation of both word and character level LMs before a final log-linear combination gave the best performance in the table. Absolute CER reductions of 0.4% and 0.3% on dev08 and p2ns were obtained over the baseline word level LM adapted using context independent interpolation. The total performance improvements over the unadapted word level 4-gram LM baseline are 0.3% on dev07, 0.7% on dev08 (7.3% relative), and 0.5% on p2ns (5.2% relative), respectively, all being statistically significant.

TABLE V. CN performance of language models on bn06, bc05, dev07, dev08, and p2ns. "○" denotes the WFST composition operation.

		CER%		
P2 system	LM adapt.	dev07	dev08	p2ns
w.4g	—	9.7	9.6	9.6
c.6g	—	10.9	10.0	10.3
w.4g ○ c.6g	—	9.5	9.1	9.3
w.4g	CI	9.6	9.3	9.4
w.4g	CD	9.5	9.2	9.3
w.4g ○ c.6g	CD	9.4	8.9	9.1

TABLE VI. CN performance of acoustic rescoring of the lattices generated by various language models on dev07, dev08, and p2ns using re-adapted acoustic models.

		CER%		
P3 system	LM adapt	dev07	dev08	p2ns
w.4g	—	9.3	8.7	9.1
w.4g	CI	9.1	8.6	9.1
w.4g	CD	9.0	8.5	8.8
w.4g ○ c.6g	CD	8.8	8.4	8.6

Table V shows the performance of multi-level combined and adapted LMs at the lattice generation stage. It would also be interesting to examine if the performance improvements can be maintained at the following stage of the CU ASR system where re-adapted acoustic models are used to rescoring the lattices generated by various LMs in Table V. These are shown in Table VI. The performance improvements from the adapted multi-level combined LM (last line of Table VI) over the word level baseline (first line of Table VI) were largely maintained. Statistically significant CER reductions 0.3%–0.5% absolute were obtained over all test sets, in particular, 0.5% absolute (5.5% relative) for dev07 and p2ns.

VIII. CONCLUSION

In this paper character level language models were used as an approximation of allowed syllable sequences that follow Mandarin Chinese syllabiotactic rules. A range of combination schemes were investigated to integrate character sequence level constraints into a standard word based speech recognition system. A generic and flexible weighted finite state transducer based language model combination and adaptation framework was also proposed. Significant error rate gains up to 7.3% relative were obtained on a state-of-the-art Mandarin Chinese broadcast audio recognition task using a history dependently adapted multi-level LM that performs a log-linear combination of character and word level LMs. These results suggest character sequence models are useful for improving Mandarin speech recognition performance. Future research will focus on incorporating character sequence constraints into more complex forms of language models, for example, neural network based LMs.[44] Syllable based acoustic modeling and the use of additional prosodic information, such as stress, will also be investigated.

ACKNOWLEDGMENTS

This work was in part supported by DARPA under the GALE program via a subcontract to BBN Technologies. The paper does not necessarily reflect the position or the policy of the US Government and no official endorsement should be inferred. The authors would also like to thank Andreas Stolcke, Elizabeth Shriberg, and Michael Levit for useful discussion and comments.

[1]J. de Francis, *The Chinese Language: Fact and Fantasy* (University of Hawaii Press, Honolulu, 1984), pp. 1–344.
[2]R. Sproat, C. Shih, N. Chang, and W. Gale, "A stochastic finite-state word-segmentation algorithm for Chinese," Comput. Linguist. **22**(3), 377–404 (1996).
[3]S. M. Chu, D. Povey, Hong-Kwang Kuo, L. Mangu, S. Zhang, Q. Shi, and Y. Qin, "The 2009 IBM GALE Mandarin broadcast transcription system," in *Proceedings of IEEE ICASSP2010*, Dallas (2010).
[4]M. J. F. Gales, X. Liu, R. Sinha, P. C. Woodland, K. Yu, S. Matsoukas, T. Ng, K. Nguyen, L. Nguyen, J.-L. Gauvain, L. Lamel, and A. Messaoudi, "Speech system combination for machine translation," in *Proceedings of IEEE ICASSP2007*, Hawaii (2007).
[5]T. Ng, B. Zhang, K. Nguyen, and L. Nguyen, "Progress in the BBN 2007 Mandarin speech to text system," in *Proceedings of IEEE ICASSP2008*, Las Vegas (2008).
[6]E. O. Selkirk, "The syllable," in *The Structure of Phonological Representations*, edited by H. van der Hulst and Norval Smieth (Fortus, Dordrecht, 1982), Vol. 2, pp. 337–385.
[7]H. Gu, C. Tseng, and L. Lee, "Markov modeling of Mandarin Chinese for decoding the phonetic sequence into Chinese characters," Comput. Speech Lang. **5**, 363–371 (1991).
[8]J. L. Hieronymus, X. Liu, M. J. F. Gales, and P. C. Woodland, "Exploiting Chinese character models to improve speech recognition performance," in *Proceedings of Interspeech'09*, Brighton (2009).
[9]L. S. Lee, C. Y. Tseng, H. Y. Gu, F. H. Liu, C. H. Chang, Y. H. Lin, Y. Lee, S. L. Tu, S. H. Hsieh, and C. H. Chen, "Golden Mandarin (I) — A real-time Mandarin speech dictation machine for Chinese language with very large vocabulary," IEEE Trans. Speech Audio Process. **1**(2), 158–179 (1993).
[10]X. Liu, M. J. F. Gales, J. L. Hieronymus, and P. C. Woodland, "Language model combination and adaptation using weighted finite state transducers," in *Proceedings of IEEE ICASSP2010*, Dallas (2010).
[11]J. Luo, L. Lamel, and J-L. Gauvain, "Modeling characters versus words for Mandarin speech recognition," in *Proceedings of ICASSP2009* (2009).
[12]H. M. Wang, T. H. Ho, R. C. Yang, J. L. Shen, B. R. Bai, J. C. Hong, W. P. Chen, T. L. Yu, and L. S. Lee, "Complete recognition of continuous Mandarin speech for Chinese language with very large vocabulary using limited training data," IEEE Trans. Speech Audio Process. **5**(2), 195–200 (1997).
[13]S. M. Katz, "Estimation of probabilities from sparse data for the language model component of a speech recognizer," IEEE Trans. Acoust., Speech, Signal Process. **35**(3), 400–401 (1987).
[14]P. Clarkson and A. Robinson, "Language model adaptation using mixtures and an exponentially decaying cache," in *Proceedings of ICASSP1997*, Munich (1997), pp. 799–802.
[15]F. Jelinek and R. Mercer, "Interpolated estimation of Markov source parameters from sparse data," in *Pattern Recognition in Practice*, edited by E. S. Gelsema and L. N. Kanal (Norh-Holland, Amsterdam, 1980), pp. 381–402.
[16]R. Rosenfeld, "A maximum entropy approach to adaptive statistical language modeling," Comput. Speech Lang. **10**, 187–228 (1996).
[17]I. Bulyko, M. Ostendorf, and A. Stolcke "Getting more mileage from web text sources for conversational speech language modeling using class-dependent mixtures," in *Proceedings of HLT '03*, Edmonton (2003).
[18]L. Chen, J-L. Gauvain, L. Lamel, G. Adda, and M. Adda, "Language model adaptation for broadcast news trancription," in *Proceedings of ISCA ITRW '01*, Paris (2001).
[19]M. Federico, "Efficient language model adaptation through MDI estimation," in *Proceedings of EuroSpeech '99*, Budapest (1999).
[20]X. Liu, M. J. F. Gales, and P. C. Woodland, "Context dependent language model adaptation," in *Proceedings of Interspeech '08*, Brisbane (2008).
[21]X. Liu, M. J. F. Gales, and P. C. Woodland, "Use of contexts in language model interpolation and adaptation," in *Proceedings of Interspeech '09*, Brighton (2009).
[22]J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recogniser output voting error reduction (ROVER)," in *Proceedings of IEEE ASRU '97* (1997).
[23]G. Evermann and P. C. Woodland, "Posterior probability decoding, confidence estimation and system combination," in *Proceedings of Speech Transcription Workshop 2000* (2000).
[24]G. Hinton, "Training products of experts by minimizing contrastive divergence," Neural Comput. **14**, 1771–1800 (2002).
[25]A. Stolcke, "SRILM—An extensible language modeling toolkit," in *Proceedings of ICSLP '02*, Denver (2002).
[26]S. J. Young, G. Evermann, M. J. F. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. C. Woodland, The HTK Book Version 3.4.1, http://htk.eng.cam.ac.uk/prot-docs/htk_book.shtml (2009) (last viewed 12/16/2011), pp. 1–374.
[27]M. Mohri and M. Riley, "Network optimizations for large vocabulary speech recognition," Speech Commun. **25**(3), 1–12 (1998).
[28]M. Mohri, F. C. N. Pereira, and M. Riley, "The design principles of a weighted finite-state transducer library," Theor. Comput. Sci. **231**, 17–32 (2000).
[29]M. Mohri, F. C. N. Pereira, and M. Riley, "Weighted finite-state transducers in speech recognition," Comput. Speech Lang. **16**(1), 69–88 (2002).
[30]M. Mohri, "Weighted automata algorithms," in *Handbook of Weighted Automata. Monographs in Theoretical Computer Science*, edited by Manfred Droste, Werner Kuich, and Heiko Vogler (Springer, Berlin, 2009), pp. 213–254.
[31]J. Darroch and D. Ratcliff, "Generalized iterative scaling for log-linear models," Ann. Math. Stat. **43**(5), 1470–1480 (1972).
[32]R. Rosenfeld, S. F. Chen, and X. Zhu, "Whole-sentence exponential language models: A vehicle for linguistic-statistical integration," Comput. Speech Lang. **15**(1), 55–73 (2001).
[33]M. Weintraub, Y. Aksu, S. Dharanipragada, S. Khudanpur, H. Ney, J. Prange, A. Stolcke, F. Jelinek, and E. Shriberg, "LM95 project report: Fast training and portability," in *1995 Language Modeling Summer Research Workshop Technical Reports*, Research Note No. 1, Center for Language and Speech Processing, Johns Hopkins University, Baltimore (1996), http://www-speech.sri.com/cgi-bin/run-distill?papers/lm95-report.ps.gz (last viewed 12/16/2011).
[34]P. C. Woodland, C. J. Leggetter, J. J. Odell, V. Valtchev, and S. J. Young, "The 1994 HTK large vocabulary speech recognition system," in *Proceedings of IEEE ICASSP1995*, Detroit (1995).
[35]D. A. Caseiro and I. Trancoso, "A specialized on-the-fly algorithm for lexicon and language model composition," IEEE Trans. Audio, Speech, Lang. Process. **14**(4), 1281–1291 (2006).
[36]R. Sinha, M. J. F. Gales, D. Y. Kim, X. Liu, K. C. Kim, and P. C. Woodland, "The CU-HTK Mandarin broadcast news transcription system," in *Proceedings of IEEE ICASSP2006*, Toulouse (2006).
[37]D. Povey and P. C. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," in *Proceedings of IEEE ICASSP2002*, Orlando (2002).
[38]S. J. Young, J. J. Odell, and P. C. Woodland, "Tree-based state tying for high accuracy acoustic modeling," in *Proceedings of ARPA Human Language Age Technology Workshop* (Morgan Kaufman, Plainsboro, NJ, 1994), pp. 307–312.
[39]C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density HMMs," Comput. Speech Lang. **9**, 171–186 (1995).
[40]N. Kumar, "Investigation of silicon-auditory models and generalization of linear discriminant analysis for improved speech recognition," Ph.D. thesis, John Hopkins University, Baltimore, 1997.
[41]X. Liu, M. J. F. Gales, and P. C. Woodland, "Automatic complexity control for HLDA systems," in *Proceedings of IEEE ICASSP2003*, Hong Kong (2003), Vol. 1, pp. 132–135.
[42]P. C. Woodland, M. J. F. Gales, D. Pye, and S. J. Young, "The development of the 1996 HTK broadcast news transcription system," in *Proceedings of DARPA Speech Recognition Workshop* (Arden House, New York, 1996), pp. 73–78.
[43]S. F. Chen and J. T. Goodman, "An empirical study of smoothing techniques for language modeling," Comput. Speech Lang. **13**(4), pp. 359–394 (1999).
[44]H. Schwenk, "Continuous language models," Comput. Speech Lang. **21**(3), 492–518 (2007).