Learning a Discriminative Weighted Finite-State Transducer for Speech Recognition

Maider Lehr and Izhak Shafran

***Abstract*—Weighted finite-state transducers (WFSTs) have been widely adopted as efficient representations of a general speech recognition model. The WFST for speech recognizer is typically assembled or composed from the several components—the language model, the pronunciation mapping and the acoustic model—which are estimated separately without any end-to-end optimization. This paper examines how the weights of such transducers can be learned in a manner that captures the interaction between the components. The paths in the transducer are represented as n-grams defined over the input and output sequences whose linear weights are learned using a discriminative criterion. The resulting linear model factors into two weighted finite-state acceptors (WFSAs) which can be applied as corrections to the input and the output side of the initial WFST. This formulation allows duration cues to be incorporated seamlessly. Empirical results on a large vocabulary Arabic GALE task demonstrate that the proposed model improves word error rate substantially, with a gain of 1.5%–1.7% absolute. Through a series of experiments, we analyze the contributions from and interactions between acoustic, duration, and language components to find that duration cues play an important role in a large-vocabulary Arabic speech recognition task. Although this paper focuses on speech recognition, the proposed framework for learning the weights of a finite transducer is more general in nature and can be applied to other tasks such as utterance classification.**

***Index Terms*—Acoustic modeling, discriminative learning, duration modeling, finite-state transducers, language modeling, learning finite-state transducers.**

I. Introduction

WEIGHTED finite-state transducers (WFSTs) have been widely adopted as efficient representations of a general speech recognition model and associated search space [1]. The WFST model for speech recognition is assembled or composed from separately estimated components. The paths of the resulting WFST represent mappings from acoustic sounds to word sequences and the weights of the paths represent scores associated with the word sequences including the contribution from all the components. Typically, the composite WFST is utilized directly in decoding speech without any attempt to optimize the weights of the composite WFST. One exception is the modification of weights related to the output word sequence through discriminative language model [2]. The composite WFST provides an opportunity to optimize the weights associated with not only the word sequence but also the acoustic state sequence, a step towards end-to-end optimization. Leaving the observation probabilities associated with the acoustic states unchanged, the focus of this paper is to learn the weights of the speech recognition transducer while taking into account both the input acoustic state sequences as well as output word sequences. The proposed approach can be applied more widely to learn weights of transducers in the context of utterance classification [3], speaker classification [4], or components of machine translation [5]. As such, we describe the approach in general terms using the framework of weighted finite-state transducers.

In order to set the context for this paper, we briefly review the weighted finite-state transducer for speech recognition in Section II. Next, we examine related work on discriminative models for speech recognition in Section III, including discriminative language models and the proposed extension in Section III-C. After describing our approach, we report experimental results on a large vocabulary Arabic speech recognition task and tease apart the impact of different factors in Section IV. Finally, we conclude with a summary.

II. Review: A Weighted Finite-State Transducer for Speech Recognition

The components of a typical speech recognition system—the language model $\mathcal{G}$, the pronunciation model $\mathcal{L}$, the decision trees for mapping phone sequences to clustered allophone sequences $\mathcal{C}$, and the topology of hidden Markov models $\mathcal{H}$—can be represented efficiently as weighted finite-state transducers [1]. While decoding, the task of searching through the combination of all intermediate symbol sequences is considerably simplified by reducing or eliminating the pursuit of redundant intermediate sequences. This can be achieved by defining the search space as a compact weighted finite-state transducer, an approach that has been widely adopted with considerable success. The integrated search space is obtained by composing the components of the speech recognition system and then removing redundant paths through determinization and minimization of the resulting transducer:

$$\mathcal{T} = \mathbf{Min}\left(\mathbf{Det}(\mathcal{H} \circ \mathcal{C} \circ \mathcal{L} \circ \mathcal{G})\right)$$

The resulting weighted finite-state transducer maps the acoustic state sequences of the hidden Markov models (HMMs) $\mathcal{S}$ on

Manuscript received January 20, 2010; revised June 01, 2010 and September 21, 2010; accepted October 15, 2010. Date of publication December 13, 2010; date of current version May 13, 2011. This work was supported in part by GALE DARPA Award HR0011-06-2-0001, DARPA Award HR0011-09-1-0041, and NIH 5R01AG027481. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Mark J. F. Gales.

The authors are with the Center for Spoken Language Understanding, Oregon Health and Science University, Portland, OR 97239 USA (e-mail: lehrm@cslu.ogi.edu; zak@cslu.ogi.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASL.2010.2090518

the input side to the word sequences $\mathcal{W}$ on the output side. The weights on a path through the transducer provides a cost, often negative log likelihood, incurred in mapping the input sequence to the output sequence of the path. The task of the decoder then is to score the acoustic frames of the given input utterance with HMM states, apply it to the input side, choose a path with the least cost and read off the word sequence on the output side of the chosen path. This decoding is significantly efficient since the redundant paths are reduced or eliminated in the precompiled search graph [6].

III. Discriminative Models

In recent years, discriminative models have gained considerable popularity in speech recognition since their estimation procedures attempt to improve word accuracy and not just the likelihood fit of the training data. Typically, the parameters of the acoustic and the language model are estimated independently. For example, discriminative language models are trained and applied on lattices from speech recognizers and the estimation ignores any related acoustic information at word or sub-word level (e.g., [2]). Likewise, discriminative estimation of HMM parameters in the form of maximum mutual information estimation (MMIE) or conditional maximum-likelihood estimation (CMLE) [7], and minimum phone error estimation (MPE) [8] are now routinely employed to learn acoustic models but they utilize a weaker language model than the one used in decoding.

There have been a few notable exceptions where interaction between acoustic and language models have been explored. Printz and Olsen incorporated a notion of acoustic confusability into language model [9]. They estimated the confusability between words by measuring distance between their hidden Markov models using an approximate close form expression. However, the parameters of the acoustic and language model were not jointly estimated. For a small task, Kuo and Gao estimated direct models of the form $P(W|A)$ for speech recognition and estimated the parameters using maximum entropy criterion [10]. This approach has been recently applied to large vocabulary tasks [11], [12] wherein both acoustic and linguistic features are evaluated to output the best hypothesis for a given input. Examples of features include n-grams, word level templates, the HMM posterior probability, and the dynamic time warp distance between a hypothesized instance and pre-defined templates. Continuing this vein of work, for the purpose of efficient sharing of parameters, Zweig and colleagues extract multiphone units automatically from the training set using maximum mutual information criterion and then use them to compute a direct model [12].

Closer to the approach adopted in this paper, a few researchers have modified the weights on the arcs of the decoding graph after composing the acoustic and the language model components [13], [14]. The key idea is to minimize the cost of the path with minimum word error rate compared to other competing paths. Lin and Yvon define a minimum classification error (MCE) criterion using a sigmoidal function defined over the weights of the graph [13]. After setting the slope and the threshold of the sigmoid, they refine the weights of the graph iteratively to optimize MCE over the training data. They demonstrate effectiveness of their estimation procedure on a small name recognition task using context independent acoustic models. This technique was subsequently applied on a large vocabulary task with context dependent acoustic models [14]. Since the technique is computationally expensive, the estimation was performed only on a subset of training data using utterances for which the misclassification function was below a threshold. They observed statistically significant gains for a baseline with a weaker language model, but not for a baseline with a stronger 4-gram language model.

In contrast to previous work, which only consider acoustic and language components, here we propose a joint model that also incorporates duration features and demonstrates a substantial performance gain on a large vocabulary task over a baseline with a powerful language model [15].

A. Discriminative Linear Model

The model presented in this paper belongs to the family of discriminatively estimated log-linear models, which have been successfully used for discriminative language modeling [2] and is briefly described below.

The decoding task is to map a given speech utterance $x \in \mathcal{X}$ to a word sequence in $y \in \mathcal{Y}$, where $\mathcal{X}$ denotes all possible acoustic inputs and $\mathcal{Y}$ denotes all possible strings, i.e., $\mathcal{Y} = \Sigma^*$, for some vocabulary Σ. Given a function $\mathbf{GEN}$ which enumerates a set of candidates $\mathbf{GEN}(x)$ for an input x (e.g., N-best hypotheses or lattices), a representation Φ mapping each $(x, y) \in \mathcal{X} \times \mathcal{Y}$ to a feature vector $\Phi(x, y) \in \Re^d$ and a parameter vector $\bar{\alpha} \in \Re^d$, the output of a linear model $F(x)$ is computed as follows:

$$F(x) = \underset{y \in \mathbf{GEN}(x)}{\arg\min}\ \Phi(x, y)\bar{\alpha}.$$

The parameters of the log-linear model can be estimated discriminatively by methods such as maximum entropy estimation (e.g., [16]) and perceptron algorithm (e.g., [2]).

For a large-vocabulary speech recognition task, the log-linear model typically contains millions of parameters and is trained over a few hundred thousand utterances, a setting for which perceptron is a convenient option [2]. Let (x_j, r_j) be the jth training example out of M, where x_j and r_j represent the speech and its transcript (or oracle in the lattice being rescored), respectively. The perceptron algorithm iterates over all the utterances in the training examples, one utterance at a time. In each iteration i, for each utterance j, the algorithm updates the parameters $\alpha_j^{(i)}$ when the best-scoring hypothesis $\bar{y}_j$ under the current model differs from r_j. Thus, the perceptron algorithm updates the model so that the score of the oracle hypothesis improves with respect to the competing hypotheses. Since the oracle by definition is the hypothesis with minimum word error rate, the learning algorithm minimizes the word error rate.

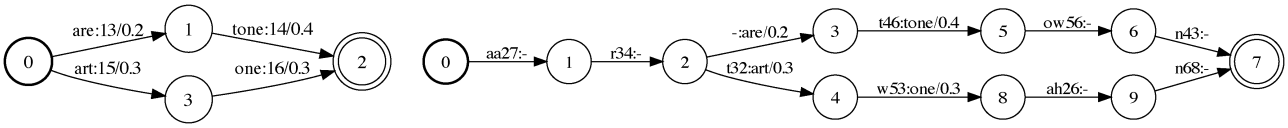

Fig. 1. Output lattice from a speech recognizer as (a) a weighted finite-state acceptor, and (b) a weighted finite-state transducer.

Algorithm 1 Averaged Perceptron $(\mathrm{x}, \mathbf{GEN}(\mathrm{x}), \mathrm{r})$

$\alpha_M^{(0)} \leftarrow 0$
$i \leftarrow 1$
repeat
  $\alpha_0^{(i)} \leftarrow \alpha_M^{(i-1)}$
  for $j = 1$ to M **do**
    $\bar{y}_j \leftarrow \arg\min_{y_j \in \mathbf{GENF}(x_j)} \Phi(x_j, y_j)\alpha_{j-1}^{(i)}$
    if $\bar{y}_j \neq r_j$ **then**
      $\alpha_j^{(i)} \leftarrow \alpha_{j-1}^{(i)} + \Phi(x_j, r_j) - \Phi(x_j, \bar{y}_j)$
    else
      $\alpha_j^{(i)} \leftarrow \alpha_{j-1}^{(i)}$
    end if
  end for
  $i \leftarrow i + 1$
until No Gain in Cross Validation WER
$\bar{\alpha} \leftarrow \sum_{k=1}^{i} \sum_{j=1}^{M} \alpha_j^{(k)}$

The algorithm is terminated when the performance of the model, evaluated after each pass through the training data on a held-out set, does not improve significantly for a few consecutive iterations.

B. Discriminative Language Model: Learning a Weighted Finite-State Acceptor

The proposed model can be motivated easily by viewing the discriminative language model as a weighted finite-state acceptor (WFSA) and its utilization in speech recognition in terms of operations on finite state machines [2]. For clarity and completeness, we summarize their description below.

Word lattices provide a compact representation of competing hypotheses and they are, in fact, a WFSA. Given an acoustic input x, let $\mathcal{L}_x$ be a word-lattice generated by the baseline recognizer, as shown in Fig. 1(a). The lattice $\mathcal{L}_x$ is an acyclic and deterministic WFSA, representing a probability distribution P_x over all strings $y \in \Sigma^*$, i.e., all possible transcriptions of x under the baseline recognizer. The weights represent the combination of acoustic and language model scores in the baseline recognizer. By casting the weights of the WFSA in log semi-ring, where the weights are negative log likelihood (lower weights correspond to better likelihood), the probabilistic computation of the language model can be performed easily in terms of standard finite-state operations.

The estimated discriminative language model can be represented as a deterministic WFSA, $\mathcal{D} = \Sigma^*$, using a failure class in the implementation (see [1] for details). The weights $w_{\mathcal{D}}[\pi]$ for all path $\pi \in \Pi_{\mathcal{D}}$ can be related to the linear form through

$$w_{\mathcal{D}}[\pi] = \sum_{j=1}^{d} \Phi_j\left(x, l[\pi]\right) \alpha_j$$

where $l[\pi]$ is the concatenation of all the labels of the path π, $\Phi_j(x, y)$ for $j > 0$ is the count of the jth n-gram in y (represents number of times the path traverses the n-gram arcs in $\mathcal{D}$) and α_j is the parameter associated with that n-gram (weight on n-gram arcs in $\mathcal{D}$). Then, the output of the discriminative language model can be computed in terms of operations on a WFSA:

$$\arg\min_{\pi} \Phi\left(x, l[\pi]\right) \bar{\alpha} = \mathbf{BestPath}(\mathcal{L}_x \circ \mathcal{D})$$

where we incorporate the scaling factor α_0 corresponding to the log probability of the baseline recognizer into $\mathcal{L}_x$.

While the WFSA view is useful, the discriminative language model can be understood and applied without resorting to finite state operation. For rescoring a hypothesis y, a feature vector $\Phi(x, y)$ consisting of n-gram subsequences from y is extracted. The resulting feature vector could potentially contain a few hundred thousand components in a large vocabulary task. Often, the feature vector for each hypothesis y includes, say in the zeroth component of $\Phi(x, y)$, the log probability of y given x as evaluated by the baseline recognizer and contains total contributions from both acoustic and language models. The feature vectors for all hypotheses are evaluated using the discriminative language model (a linear model), and the best scoring hypothesis is chosen as the output of the rescoring pass.

Discriminative language models have been empirically demonstrated to improve speech recognition in English [2]. In morphologically rich languages with free word order, additional effort is required to overcome their large vocabulary sizes, which involves factorizing the morphological components appropriately before recognition gains can be observed [16], [17].

C. Joint Discriminative Model: Learning a Weighted Finite-State Transducer

We propose a joint acoustic, duration, and language model based on the observation that the baseline recognizer provides more information about competing hypotheses than just the word sequences.

Consider the general finite-state model of a speech recognizer, composed of weighted finite-state components [1]. This transducer represents a mapping from the acoustic state sequence to the word sequence and incorporates all the intermediate mappings. The lattice generated by a baseline recognizer can be viewed as an *a posteriori* version of this transducer, whose weights represent the posterior distribution conditioned on the given (decoded) acoustic input utterance. Thus, the

lattice generated by a decoder in a general case is a weighted finite-state transducer (WFST) and not just a WFSA.

Given an acoustic input x, let the output side of a path in the lattice $\mathcal{L}_x$ represent word sequences $y \in \mathcal{Y}$ (as before) and the input side represent acoustic state sequences $s \in \mathcal{S}$ associated with the path as identified by the back-trace of the Viterbi algorithm in the baseline recognizer and illustrated in Fig. 1(b). The state sequence may be augmented by prosodic features such as duration, or phonological features such as pronunciation variants from the Viterbi path, which we include in the acoustic component for ease of description in this section.

In our joint model, we expand the feature representation, from $\Phi(x, y)$ to $\Phi(x, s, y)$, to include features defined over the input state sequence s in addition to those defined over y. The lattice $\mathcal{L}_x$ may, in general, have multiple state sequences $s \in \mathcal{S}_y$ associated with an output sequence y. Once again, given a function $\mathbf{GEN}$ which enumerates a set of candidates $\mathbf{GEN}(x)$ for an input x (e.g., N-best hypotheses or lattices), a state sequence s associated with y, a representation Φ mapping each $(x, s, y) \in \mathcal{X} \times \S \times \mathcal{Y}$ to a feature vector $\Phi(x, s, y) \in \Re^d$ and a parameter vector $\bar{\alpha} \in \Re^d$, the output of our linear model $F(x)$ is very similar to the earlier case except for the state sequences s associated with hypotheses y of an utterance x.

$$F(x) = \underset{s,y \in \mathbf{GEN}(x)}{\arg\min} \Phi(x, s, y)\bar{\alpha}.$$

The features for our model consist of n-grams extracted from not only y but also s. For each path, we concatenate the n-grams from y with those from s. Thus, $\Phi(x, s, y) = [\Phi_w(x, s, y)\Phi_s(x, s, y)]$, where subscripts w and s denote the lexical and state-specific features corresponding to the speech utterance x and a path mapping the input state sequence s to the output sequence y. In a large vocabulary system, the number of acoustic states (automatically clustered allophone-states) are only a few thousand in number, at least one to two orders of magnitude lower than the size of the word vocabulary. Thus, the number of parameters in the joint model will be marginally smaller than those of discriminative language models.

The discriminative joint model can be represented in terms of finite-state machines, but this time they can be factored into two WFSTs. The parameters $\Phi_w(x, s, y)$ corresponding to the features of y can be converted to $\mathcal{D}$, the discriminative language model, as before. Similarly, the parameters $\Phi_s(x, s, y)$ corresponding to the acoustic component can be converted to another WFST, $\mathcal{E}$. Then, the output of the model $F(x)$

$$\underset{s,y \in \mathbf{GEN}(x)}{\arg\min} \Phi_s(x, s, y)\bar{\alpha}_s + \Phi_w(x, s, y)\bar{\alpha}_w$$

can be computed using finite-state operations.

$$F(x) = \mathbf{BestPath}(\mathcal{E} \circ \mathcal{L}_x \circ \mathcal{D}).$$

While rescoring, the lattice $\mathcal{L}_x$ is composed on the input side with $\mathcal{E}$ (the acoustic component), on the output side with $\mathcal{D}$ (the language component), and then projected to the output side. The minimum distance path of the minimized WFSA is the output of the joint discriminative model. In the experiments reported in this paper, $\mathcal{L}_x$ is constrained so that it contains only one state sequence s for each y. However, in the most general case, multiple state sequences may be associated with each y and the evidence from all the associated states sequences s needs to be summed probabilistically. This can be achieved through standard FSM operations, specifically, minimization and determinization.

$$F(x) = \mathbf{BestPath}\left(\mathbf{Min}\left(\mathbf{Det}\left((\mathcal{E} \circ \mathcal{L}_x \circ \mathcal{D})\right)\right)\right).$$

The proposed model can also be implemented without any finite-state operations. While evaluating n-best hypotheses during rescoring, n-gram features corresponding to lexical, acoustic states, and duration are extracted. The resulting feature vector is evaluated for all hypotheses using the discriminative linear model to pick the one with the lowest cost.

Note, the proposed model differs significantly from an earlier attempt to correct the general model $\mathcal{G}$ discriminatively [14]. In their approach, they compute the sentence error rate for each hypothesis y, smooth the errors using a sigmoidal function and then update the weights using a gradient descent. Apart from the differences in the parameter estimation, their models do not utilize local context (e.g., n-gram), which can provide important cues related to the errors. The factorization of a linear model into two WFST has been investigated before in the context of utterance classification [3], where the two factors were cascaded to process appropriately modified inputs. Instead, here we use the factorization for learning the mapping of one sequence to another. The proposed model is also more efficient than the *flat direct model* [11]. In their model, the features are functions of words and the multi-phones within their span, potentially causing data sparsity. Instead, we factor the feature space efficiently into two separate domains corresponding to word sequences and state sequences. When the span of the input and the output symbols on a path of $\mathcal{L}_x$ are synchronized, features similar to the *flat direct model* [12] could be extracted and included in the proposed discriminative model, but then it will no longer factor as described above and hence will not be amenable to finite-state operations easily.

IV. Empirical Results and Analysis

A. Task

Corpus: The proposed model was evaluated empirically on GALE Arabic Transcription task. The training data for our models consisted of about 200 hours ($\approx$ 1.5 M words) of Arabic broadcast conversations. The models were evaluated on the 20-fold cross-validation of the training data as well as on two independent test sets, namely, the 2007 GALE development set and the 2007 GALE evaluation set, distributed by NIST. The development set consisted of 55 shows, totaling 2.6 hours ($\approx$ 18 K words) from 10 BN and 10 BC sources and the evaluation set contains 65 shows, totaling 2.8 hours ($\approx$ 20 K words).

Baseline System: The baseline acoustic models, trained on about 1000 h of Arabic broadcast data, contain 45 phones including long vowels, a three-state left-to-right HMM topology for phones, 5000 clustered pentaphone states (its context includes word boundaries and pauses), a linear discriminant transform, and a semi-tied covariance transform, described in detail in [18]. The acoustic features consist of 13 coefficient perceptual linear coding vectors with speaker-specific vocal tract length normalization (VTLN). The baseline 4-gram language model with 122 M n-grams, was estimated by interpolating 14 components. The vocabulary is relatively large at 737 K and the associated dictionary has only single pronunciations. To reduce

TABLE I
FEATURES CORRESPONDING TO THE WORD AND STATE SEQUENCES ASSOCIATED WITH A GIVEN HYPOTHESIS

Φ_w	Φ_s	Φ_d
<s>,are = 1	*<s>,1000* = 1	*<s>,*1000_2 = *1*
are, tone = 1	*1000,4546* = 2	1000_2*,*4546_1 = *2*
...	...	...

TABLE II
COMPARISON OF DISCRIMINATIVE MODELS WITH LANGUAGE, ACOUSTIC AND DURATION COMPONENTS, MEASURED (WER) ON 20-FOLD CROSS-VALIDATION, DEV07, AND EVAL07

System	Params	F_U	F_C	Xval	Dev07	Eval07
Baseline	-	-		26.6	21.2	24.0
Φ_{w2}	1.1M	54K	34M	26.2	20.9	23.8
Φ_{w2}, Φ_{s2}	1.4M	84K	539M	25.7	20.6	23.4
$\Phi_{w2}, \Phi_{s2}, \Phi_{d2}$	3.4M	456K	2.1G	**25.1**	**19.5**	**22.5**

the computational cost of decoding the training data, speaker adaptation and cross system adaptation was not used and hence the results of the baseline reported in this paper correspond to an intermediate VTLN stage in [18], the IBM GALE system for the second project year. Discriminative models are most effective when the models used for training and testing are matched and so the proposed models are compared with the performance of the same VTLN model on the test sets.

Generation of Competing Hypotheses: For training the proposed model, each utterance x was decoded with the baseline recognizer to generate weighted lattices or WFSTs, encoded with word sequences on the output side, acoustic states with their time marks on the input side, and the associated log probabilities for costs. The baseline language model contains transcripts corresponding to the training data for the discriminative model. In order to avoid biasing the resulting lattices, the decoding was performed using 20-fold cross-validation, where the transcripts from 19 folds were used in the language model while decoding the held-out fold. From each lattice, 100-best unique hypotheses were used to form the competing hypotheses. For each hypothesis, we also extracted the single best state sequence and associated durations.

Parameter Estimation: The parameters of the models were estimated using the perceptron algorithm, iteratively, until no improvements were observed for five consecutive iterations as evaluated on the held-out set (one fold). The oracle hypotheses were chosen as the reference for the perceptron algorithm. The maximum-likelihood (ML) scores were not used as features in the discriminative model since the ML scores are good predictors of hypotheses with least word error rate (WER) and are likely to overwhelm the other features [19]. Instead, the ML scores were interpolated with the scores from the discriminative models for each utterance using an interpolation weight α_0, which was optimized over 20-fold cross-validation.

Encoding Duration: Duration can be encoded at the word, phone, or acoustic state level. When the vocabulary is large such as in inflected languages with free word order, a large number of words are likely to be observed very infrequently in the training data, making it difficult to estimate robust word-level duration models for them. Phones, on the other hand, are too coarse as units for representing duration. Clustered allophone states provide a reasonable compromise between these two extremes and so we chose to model their durations. Note, the contexts for the allophones include pauses and word boundaries which allows the allophones to capture pre-pausal lengthening that is well studied in phonetics.

Even though durations are inherently continuous, they are quantized by the recognizer to a resolution of 0.01 s for a 100 frames/s system. Empirically, the range of durations observed for each state is limited even in 200 h of speech. So, we encode each observed duration of a clustered state as a separate feature with its own model parameter.

Features Space: The features of our model can be understood easily by considering an example utterance.

- Word sequence (output side):

$$< s > are\ tone < /s >$$

- Clustered-state sequence (input side):

$$< \mathrm{s} >, 1000, 1000, 4546, 4789, 1000, 1000, 4546, \ldots$$

The feature space of our joint model is illustrated in Table I, where Φ_w, Φ_s, Φ_d represent the n-gram subsequences or features on y, acoustic states, and their durations, respectively. The sub-sequence—1000, 1000, 4546—is represented as a duration augmented state sequence bigram—1000_2, 4546_1—where 1000_2 denotes a single run of length two for state 1000 followed by a run of length one for state 4546. In the example sequence, the bigram feature is assigned a value of two since the bigram occurs twice. Thus, in the bigram case, the duration feature space spans the product of clustered bigram state space augmented with every possible observed run length.

B. Experimental Results

We evaluated the discriminative linear model with a simple implementation using the best state sequence s for a given y, instead of summing over all state sequences associated with y. The models were evaluated on 20-fold cross-validation of the training set (Xval) and on two independent test sets (Dev07 and Eval07) in three conditions using: 1) only the lexical features Φ_{w2}, 2) only the acoustic and language features $[\Phi_{s2}, \Phi_{w2}]$, and then 3) augmenting 2) with duration features $[\Phi_{s2}, \Phi_{w2}, \Phi_{d2}]$. For training, 100-best unique hypotheses were extracted and the resulting models were evaluated on 1000-best unique hypotheses. Although the number of potential bigram features is large, the average perceptron algorithm estimates non-zero weights for only those that are observed in the training data and are useful in discriminating the oracle from competing hypotheses. The number of such active features or parameters for each model is listed in second column in Table II. To provide an insight into how often those parameters were used in the test set, the number of unique features F_U and their coverage F_C are listed in third and fourth columns, respectively. The performance on Xval is lower than Dev07 and Eval07 because the WER was measured on the surface form, without the equivalent representations of words (e.g., akin to *gonna*==*going to* in Arabic), which was available from NIST for Dev07 and Eval07. The gains reported in the table are statistically significant with respect to the baseline when they are evaluated using standard NIST statistical tests.

The bigram discriminative language model has about 1 M parameters and provides a consistent gain of about 0.2%–0.4% across the three sets, similar to the gains observed in inflectional languages such as Czech and Turkish [16], [17]. The addi-

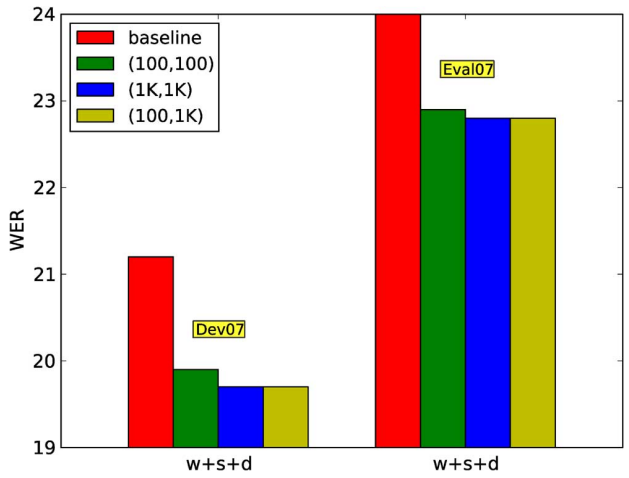

Fig. 2. Impact on Dev07 and Eval07 performance (WER) of the model when the number of competing hypotheses is changed. Legend represents $(N_{\text{train}}, N_{\text{test}})$ as explained in the text.

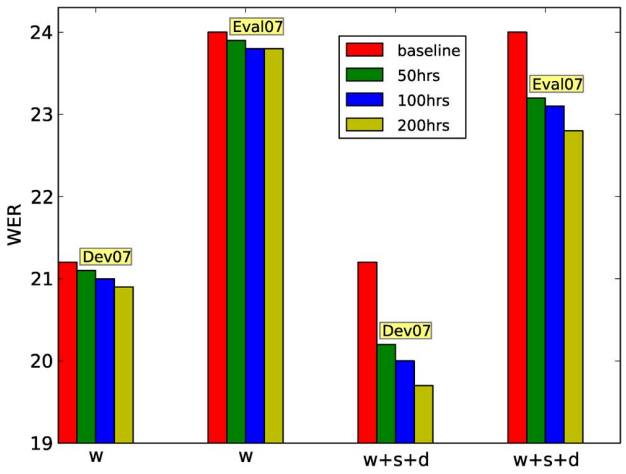

Fig. 3. Impact of the training data size on the performance of Dev07 and Eval07 with Φ_w, and $\Phi_w + \Phi_s + \Phi_d$.

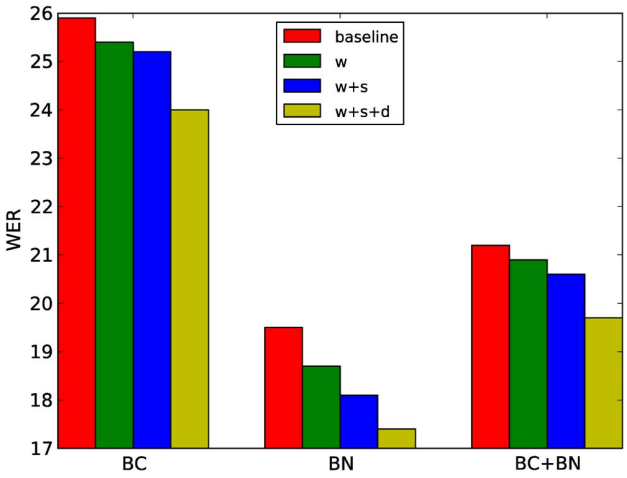

Fig. 4. Performance on Broadcast Conversations (BC) and Broadcast News (BN) portion of Dev07 with models trained on BC.

tion of the acoustic component to the model improved the gain to about 0.6%–0.9%. The number of unique features encountered in the test increased only modestly, from 54 K to 84 K, since there are about 6 phones and 18 acoustic states for each word on the average. However, the acoustic features are exercised more often than lexical features, about 539 M compared to 34 M, contributing disproportionately to the total score from the log-linear model. The number of duration bigram features are large as expected from the simple encoding of duration. The resulting model further boosts the gain to about 1.5%–1.7% over the baseline recognizer. In the conventional recognizer, the transition matrix of HMMs of allophones estimated by maximum likelihood has little impact on the performance of a large vocabulary system since it is dwarfed by the large dynamic range of the likelihood of the high-dimensional observation space. The transition matrix is typically assigned a uniform probability for all allowable transitions. In contrast, the gains observed in our model are a result of jointly estimating the duration parameters along with transition weights of $\mathcal{H}$ and $\mathcal{G}$ corresponding to features Φ_s and Φ_w using a discriminative criterion.

C. Analysis

In this section, we analyze the influence of empirical factors. For clarity of legends, the plots use a simplified notation, specifically, w, $w+s$, $w+s+d$ to denote the feature space corresponding to Φ_w, $[\Phi_w \Phi_s]$, and $[\Phi_w \Phi_s \Phi_d]$.

Effect of the Number of Competing Hypotheses: The size of the lattice or the number of competing hypotheses being considered in the training data and rescored during testing impacts both the accuracy and the computational cost.

The number of competing hypotheses in the training and test set can be varied independently and we explored three settings, represented in Fig. 2. The legend in the graph shows pairs, $(N_{\text{train}}, N_{\text{test}})$, where the first number refers to the size of hypotheses in the training set and the second to that in the test set. The number of hypotheses in training effects computational cost disproportionately more than the number of hypotheses in testing, since the perceptron algorithm has to iterate over all the hypotheses. As a compromise between the settings (100, 100) and (1 K, 1 K), we also investigated (100, 1 K) setting. As the number of hypotheses in training and testing is increased from 100 to 1 K, the performance of the model improves steadily. The models with duration features exhibit a gain of about 0.2%–0.3% in Dev07 and 0.1%–0.2% in Eval07. From these experiments, it is clear that most of the performance gain can be obtained by training with 100 hypotheses and testing with 1000 hypotheses.

Effect of the Training Data Size: The amount of training data impacts the performance of discriminative models. In order to understand the trend, we measured the performance of the integrated models when they are trained with 200 h, 100 h, and 50 h of data. The results, illustrated in Fig. 3, shows that the performance steadily improves as the size of training data is increased and does not appear to plateau for 200 hours and so we expect further gains with the complete 1000 hours or more of training data.

Effect of the Differences in Genre: Recall that the discriminative models presented in this paper were trained on 200 hours of Broadcast Conversations (BC). However, the test data also includes Broadcast News (BN). This provides an opportunity to test the ability of the different features to generalize across genre. The performance gains, shown in Fig. 4, with lexical Φ_w

TABLE III
DETAILED PERFORMANCE (WER) COMPARISON, MEASURED ON 20-FOLD CROSS-VALIDATION, DEV07, AND EVAL07, USING MODELS WITH DIFFERENT COMBINATIONS OF ACOUSTIC, DURATION, AND LEXICAL n-GRAMS FEATURES

System	#Feats.	Xval	Dev07	Eval07
Baseline Recognizer	-	26.6	21.2	24.0
Φ_{w1}	164K	26.34	21.09	23.87
Φ_{w1}, Φ_{s1}	169K	25.85	20.65	23.58
Φ_{w1}, Φ_{s2}	159K	25.72	20.61	23.37
Φ_{w1}, Φ_{d1}	271K	25.40	19.82	22.77
Φ_{w1}, Φ_{d2}	1.78M	25.37	19.70	22.86
$\Phi_{w1}, \Phi_{s1}, \Phi_{d1}$	276K	25.35	19.89	22.78
$\Phi_{w1}, \Phi_{s2}, \Phi_{d1}$	253K	25.25	19.77	22.67
$\Phi_{w1}, \Phi_{s1}, \Phi_{d2}$	1.79M	25.32	19.76	22.66
$\Phi_{w1}, \Phi_{s2}, \Phi_{d2}$	1.85M	25.21	19.68	22.57
Φ_{w2}	1.13M	26.22	20.9	23.76
Φ_{w2}, Φ_{s1}	1.18M	25.72	20.49	23.57
Φ_{w2}, Φ_{s2}	1.47M	25.66	20.60	23.40
Φ_{w2}, Φ_{d1}	1.45M	25.26	19.74	22.77
Φ_{w2}, Φ_{d2}	3.06M	25.30	19.67	22.75
$\Phi_{w2}, \Phi_{s1}, \Phi_{d1}$	1.54M	25.24	19.53	22.73
$\Phi_{w2}, \Phi_{s2}, \Phi_{d1}$	1.59M	25.19	19.71	22.68
$\Phi_{w2}, \Phi_{s1}, \Phi_{d2}$	3.02M	25.22	19.75	22.49
$\Phi_{w2}, \Phi_{s2}, \Phi_{d2}$	3.15M	25.16	19.61	22.65
Φ_{s1}	4,87K	25.94	20.61	23.53
Φ_{s2}	41,26K	25.76	20.55	23.46
Φ_{d1}	107.23K	25.49	19.77	22.89
Φ_{d2}	1.72M	25.44	19.66	22.82
Φ_{s1}, Φ_{d1}	112.1K	25.41	19.77	22.78
Φ_{s2}, Φ_{d1}	1.35K	25.30	19.56	22.70
Φ_{s1}, Φ_{d2}	1,68M	25.36	19.77	22.64
Φ_{s2}, Φ_{d2}	1.71M	25.25	19.70	22.56
$\Phi_{w1,w2}, \Phi_{s1,s2}, \Phi_{d1,d2}$	3.4M	25.12	19.50	22.49

and acoustic states Φ_s is higher in Broadcast News portion of Dev07 while the duration model provides similar gains across both genre.

Interaction Between Acoustic, Language, and Duration Components: Next, we investigated the interaction between acoustic, duration, and language features at the utterance level for different orders of n-gram. The results, reported in Table III, are grouped separately for different orders of lexical, acoustic, and duration n-grams. In the table, for example, $w2$ and $w1$ indicate lexical n-grams containing bigrams and unigrams respectively. Note, the lexical n-gram features mentioned here are specific to the discriminative language model and this is different from the 4-gram language model, estimated from a much larger corpora, whose scores are included in the likelihoods from the baseline recognizer. For each model, the interpolation weight (α_0) was tuned on Xval to maximize the performance. The results reported in the table show consistent trends except for a few results denoted in gray. For models with unigram lexical features, the performance improves consistently across the three test sets as the order of n-gram for acoustic state and duration features is increased from unigram to bigram with only one exception out of 24 results. Recall, the duration features explicitly encode observed span for the states, as a result its bigrams are numerous. When acoustic and duration features are augmented to lexical bigrams, the performance improves consistently except for four cases out of 24 results. Surprisingly, in this Arabic task, lexical features do not provide significant additional gains beyond that obtained from acoustic state and duration features. The most useful features appear to be bigrams of durations of the clustered states. We observed the best performance when acoustics, durations, and lexical features of hypotheses were encoded as bigrams as well as unigrams in the joint model.

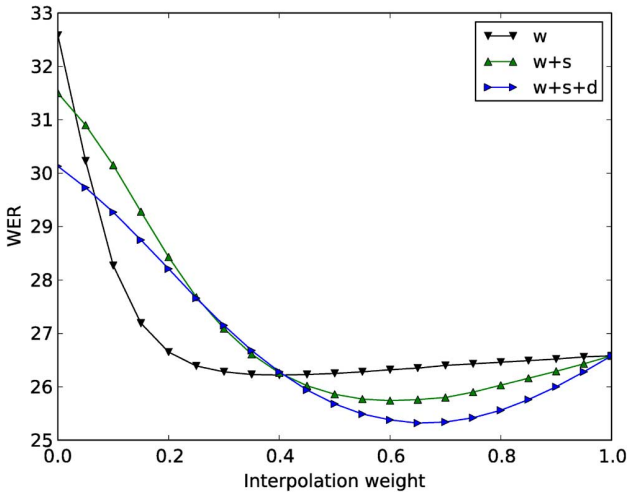

Fig. 5. Effect of the interpolation weight α_0 on the performance of XVal for the three types of features.

Effect of the Interpolation Weight (α_0)*:* For all experiments reported in this paper, we rescored the N-best list by interpolating the costs of the discriminative model (L_{DM}) with the negative log likelihood from the baseline system (L_{ML}) according to the following formula:

$$L = \alpha_0 * L_{\mathrm{ML}} + (1 - \alpha_0) * L_{\mathrm{DM}}.$$

The best performance on the cross-validation set was used as the optimal interpolation weight. As the Fig. 5 shows, the performance on cross-validation set is not overly sensitive to small changes in the interpolation weight and the local minima for joint models that includes duration features is easy to locate.

V. SUMMARY

This paper proposes a joint discriminative model to incorporate acoustic, duration, and language components. We describe how this model generalizes the discriminative language model in [2]. The model factors into two WFSTs which can be applied to the input $(\mathcal{E})$ and the output side $(\mathcal{D})$ of the general model $(\mathcal{G})$ for speech recognition $\mathcal{E} \circ \mathcal{G} \circ \mathcal{D}$ and thus can also be used in first pass recognition. Empirical results demonstrate that the proposed joint model improves performance by 1.5%–1.7% absolute on a GALE Arabic transcription task. The focus of this paper is on learning the weights of the finite-state transducer for speech recognition and as such the observation probabilities have not been modified. For practical reasons, we used observation models from the VTLN baseline system. At the least, since the duration features bring information currently not exploited fully in speech recognition system, we expect the proposed model will provide gains for other observation models such as MMI and MPE.

The analysis of the model suggests the following five points. First, the performance of the joint model improves as the amount of the training data increases from 50 h to 100 h and 200 h with no sign of saturation, leading to the expectation that further gains may be obtained with additional training

data. Second, even though the joint models were trained on Broadcast Conversations, the model with duration features performed on Broadcast News as well as on Broadcast Conversations, exhibiting a good generalization across genre. Third, empirical results show that most gains from the joint model can be obtained by training with only 100 competing hypotheses while testing with 1000 competing hypotheses. This is good news since the cost of training using iterative procedure is more than that of testing this log-linear model. Fourth, the joint model is not overly sensitive to the interpolation weight and a robust operating point can be located with ease. Fifth, the joint model captures the interaction between duration cues and lexical n-grams. The gains from the duration features appear to be higher with lexical bigrams than with lexical unigrams or when the lexical features are used in isolation. In contrast, the duration of allophones modeled through transition matrix of HMMs, estimated by maximum likelihood, has failed to provide any gains in large vocabulary recognition. The likelihood from the transition matrix is dwarfed by those from the high-dimensional observation component of the HMMs. The key reason for the success of our duration model is that it is jointly estimated with other components of the speech recognizer specifically the word transitions $\mathcal{G}$, and the acoustic state transitions $\mathcal{H}$. Furthermore, the parameters are estimated to minimize the word error rate directly.

ACKNOWLEDGMENT

The authors would like to thank the Advanced Large Vocabulary Speech Recognition Group at IBM Watson Research Center (Yorktown Heights, NY) for making their tools and models available for this work and specifically B. Kingsbury for facilitating the collaboration. They would also like to thank B. Roark for several useful discussions and for the implementation of perceptron code.

REFERENCES

[1] M. Mohri, F. Pereira, and M. Riley, "Weighted finite-state transducers in speech recognition," *Comput. Speech Lang.*, vol. 16, no. 1, pp. 69–88, 2002.

[2] B. Roark, M. Saraclar, and M. Collins, "Discriminative n-gram language modeling," *Comput. Speech Lang.*, vol. 21, no. 2, pp. 373–392, 2007.

[3] M. Saraclar and B. Roark, "Utterance classification with discriminative language modeling," *Speech Commun.*, vol. 48, no. 3–4, pp. 276–287, 2006.

[4] I. Shafran, M. Riley, and M. Mohri, "Voice signatures," *IEEE Autom. Speech Recognition Understanding*, pp. 31–36, 2003.

[5] S. Kumar, Y. Deng, and W. Byrne, "A weighted finite state transducer translation template model for statistical machine translation," *Nat. Lang. Eng.*, vol. 12, no. 1, pp. 35–75, 2006.

[6] S. Kanthak, H. Ney, M. Riley, and M. Mohri, "A comparison of two LVR search optimization techniques," in *Proc. ICSLP*, 2002.

[7] V. Valtchev, J. J. Odell, P. C. Woodland, and S. J. Young, "MMIE training of large vocabulary recognition systems," *Speech Commun.*, vol. 22, no. 4, pp. 303–314, 1997.

[8] D. Povey and P. C. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," in *Proc. ICASSP*, 2002.

[9] H. Printz and P. Olsen, "Theory and practice of acoustic confusability," *Comput. Speech Lang.*, pp. 131–164, 2002.

[10] H.-K. J. Kuo and Y. Gao, "Maximum entropy direct models for speech recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 3, pp. 873–881, May 2006.

[11] G. Heigold, G. Zweig, X. Li, and P. Nguyen, "A flat direct model for speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Washington, DC, 2009, pp. 3861–3864, IEEE Computer Society.

[12] G. Zweig and P. Nguyen, "Maximum mutual information multi-phone units in direct modeling," in *Proc. ICASSP*, 2009, pp. 3861–3864.

[13] S.-S. Lin and F. Yvon, "Discriminative training of finite state decoding graphs," in *Proc. Interspeech*, 2005.

[14] H.-K. J. Kuo, B. Kingsbury, and G. Zweig, "Discriminative training of decoding graphs for large vocabulary continuous speech recognition," in *Proc. ICASSP*, 2007, vol. 4, pp. 45–48.

[15] M. Lehr and I. Shafran, "Discriminatively estimated joint acoustic, duration and language model for speech recognition," in *Proc. IEEE ICASSP*, 2010, pp. 5542–5545.

[16] I. Shafran and K. Hall, "Corrective models for speech recognition of inflected languages," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2006, pp. 390–398.

[17] E. Arisoy, M. Saraclar, B. Roark, and I. Shafran, "Syntactic and sub-lexical features for Turkish discriminative language models," in *Proc. ICASSP*, 2010, pp. 5538–5541.

[18] H. Soltau, G. Saon, B. Kingsbury, H.-K. Kuo, D. Povey, and A. Emami, "Advances in Arabic speech transcription at IBM under DARPA GALE program," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, no. 5, pp. 884–894, Jul. 2009.

[19] C. Sutton, M. Sindelar, and A. McCallum, "Reducing weight undertraining in structured discriminative learning," in *Proc. HLT-NAACL*, 2006.

Maider Lehr received the B.S. degree in telecommunication engineering from the University of the Basque Country, Bilbao, Spain, in 2002. She is currently pursuing the Ph.D. degree in computer science at the Center for Spoken Language Understanding (CSLU), Oregon Health and Science University (OHSU), Portland.

She worked in speech technology at VICOMTech, San Sebastian, Spain, before joining OHSU. Her research interests include speech recognition and processing and statistical learning methods.

Izhak Shafran received the Ph.D. degree in electrical engineering from the University of Washington, Seattle.

He is currently an Assistant Professor at Oregon Health and Science University (OHSU), Portland, and is a member of the Center for Spoken Language Understanding (CSLU). After the Ph.D. degree, he joined AT&T Labs Research in the Speech Algorithms Group. Before joining OHSU, he was a Visiting Professor at the University of Paris-South and then a research faculty in the Center for Language and Speech Processing (CLSP) at the Johns Hopkins University, Baltimore, MD. His research interests are in modeling speech and analyzing conversational speech.