Discriminative Joint Modeling of Acoustic and Lexical Variations for Spoken Language Processing

Maider Lehr

Presented to the
Center for Spoken Language Understanding
within the Oregon Health & Science University
School of Medicine
in partial fulfillment of the
requirements for the degree
Doctor of Philosophy
in
Computer Science & Engineering

June 2014

Center for Spoken Language Understanding
School of Medicine
Oregon Health & Science University

CERTIFICATE OF APPROVAL

This is to certify that the Ph.D. dissertation of

Maider Lehr

has been approved.

Dr. Izhak Shafran, Thesis Advisor
Research Scientist, Google Inc.

Dr. Brian Roark
Research Scientist, Google Inc.

Dr. Richard Sproat
Research Scientist, Google Inc.

Dr. Brian Kingsbury
Research Staff Member, IBM Research

Dr. Jan van Santen
Professor, OHSU

Dr. Kemal Sonmez
Associate Professor, OHSU

Dedication

To my parents Carlos and Arantxa for their infinite patience

Acknowledgements

First, I would like to thank my advisor Izhak Shafran for giving me the opportunity to pursue my Ph.D. at CSLU/OHSU under his guidance. During all these years Izhak has always been there to share his expertise with me, motivate me, and guide me along this journey. Many thanks are also due to my other committee members, Brian Roark, Richard Sproat, Brian Kingsbury, Jan van Santen and Kemal Sonmez for their suggestions and interest on my work during all these years, and for their feedback on my proposal and dissertation. I also want to acknowledge Emily Tucker Prud'hommeaux and Kyle Gorman for their input that allowed me to complete my research.

I am also grateful for the great people I have met at CSLU. It has been a pleasure to share all these years with Meysam Asgari, Christopher Whelan, Alireza Bayestehtashk, Ethan Selfridge, Eric Morley, Aaron Dunlop, Mahsa Sadat Elyasi, Seyed Hamidreza Mohammadi, Andrew Fowler, Russel Beckley, Géza Kis, Qi Miao, Guillaume Thibault and Ranjani Ramakrishnan. In particular, many special thanks to Emily Tucker Prud'hommeaux for great personal and professional support, Akiko Kusumoto Amano for all her help during my first years at CSLU, and Mahsa Yarmohammadi, Masoud Rouhizadeh, Shiran Dudy and Golnar Sheikhshab, for making with their conversations the days at CSLU more enjoyable. I also want to recognize the departmental staff of CSLU, Patricia Dickerson, Sean Farrell and Robert Stites for their invaluable work that allows us to mainly focus on our research.

Finally, I cannot conclude the acknowledgments without expressing how in debt I am to my brother and my parents for always encouraging me to do my best and do not give up.

Contents

List of Tables

List of Figures

Abstract

Discriminative Joint Modeling of Acoustic and Lexical Variations for Spoken Language Processing

Maider Lehr

Doctor of Philosophy
Center for Spoken Language Understanding
within the Oregon Health & Science University
School of Medicine

June 2014
Thesis Advisor: Dr. Izhak Shafran

Speech recognition systems consist of three components, namely, the acoustic model, the pronunciation model and the language model. The acoustic and language models are typically learned separately and furthermore optimized for different cost functions. This framework has been a result of historical and practical considerations such as the availability of limited amounts of training data and the computational cost. These considerations are currently being overcome. Arguably, learning both models jointly to directly minimize the word error rate will result in a better recognizer.

One of the contributions of this thesis is a detailed investigation of a discriminative framework to jointly learn the parameters of the acoustic, language and duration models (commonly captured with the parameters from the acoustic models). The acoustic state transition parameters, the n-gram language model parameters and the state duration parameters are learned using a reranking framework, which has been previously employed in discriminative language models [156]. We report experiments on the GALE Arabic transcription task, a NIST benchmark, with about 200 hours of training data and two test sets of about 2.5 hours. Our results demonstrate that our model improves the performance by about 1.4-1.6% absolute word error rate over the baseline system.

Continuing with the joint modeling framework, next, we apply it to learn pronunciation variations particular to African American Vernacular English (AAVE) speech. Popular speaker adaptation methods adapt the acoustic models quickly using small amounts of data, for example, by estimating a few linear transforms. Such transformations are incapable of appropriately capturing systematic phonetic transformations. We investigate strategies for learning phonetic transformations jointly with the discriminative language model. We compare our new models on NPR's StoryCorps corpus, which consists of stories from self-identified AAVE and Standard American English (SAE) speakers. The joint discriminative pronunciation and language model improves the performance of the AAVE recognizer by about 2.0% WER of which about 0.5% can be attributed to pronunciation models. Improvements on the SAE data are lower and mainly attributed to the discriminative language model.

Finally, we examine how joint modeling of acoustic and lexical variations can improve the performance of a downstream application, a narrative retelling assessment tool. We develop a conditional random field (CRF) based model to incorporate both variations, and demonstrate gains of 6.3% over a generative baseline in the F-score of detecting story elements on a clinical task, the Wechsler Logical Memory test.

Chapter 1

Introduction

Recent advances in speech recognition technology and the growing societal adoption of mobile devices have led to an unprecedented demand for natural spoken interfaces in a wide variety of commercial applications. Automatic mobile virtual assistants such as Apple's Siri[1], Google Now[2] and recent Microsoft's Cortana[3] are becoming popular. We expect voice search on our smartphones to reliably and accurately locate the pertinent documents, images, videos, music or other forms of content on the web. A large number of companies are devoting considerable resources to allow users to speak to appliances such as television or car navigation equipment to control them. Spoken language systems have matured sufficiently that they are beginning to be employed in tutoring systems for children and for second language learners. This provides a personalized and easy way for students to practice at their own pace, making education more accessible to everyone, including those with different forms of learning disabilities. Speech applications are also finding inroads into healthcare, where physicians now routinely employ them for generating their examination reports.

Spoken language applications typically consist of a cascade of systems that are developed or trained separately. In this cascade, the speech recognizer is often the first system, tasked with inferring the spoken words, which are then fed into the downstream components. The downstream components are often developed separately mostly due to

[1] `www.apple.com/ios/siri/`

[2] `https://support.google.com/websearch/answer/2842392?hl=en`

[3] `http://www.techradar.com/news/phone-and-communications/mobile-phones/cortana-everything-you-need-to-know-about-microsoft-s-siri-rival-1183607`

the availability of larger amounts of annotated text data than speech data. However, this way of breaking up an application is suboptimal due to several reasons:

- The components are usually stochastic and the inferred output inevitably contains errors which are propagated to the downstream components.
- The components are trained independently and optimized using objective functions that are often not directly related to the performance measures of the application.
- Furthermore, the underlying assumption that the components are independent may not be true.

Several solutions have been proposed in the literature to alleviate the shortcomings of the separately trained components. The impact of the errors from the previous component in the cascade can be reduced by not relying exclusively on just one output from the previous component but a distribution of outputs with associated weights [110, 50, 87]. In certain applications, it may be possible to optimize the parameters of the models in all the components to maximize the performance of the full system [162, 197].

Even within the speech recognizer, the different components, the acoustic model, the pronunciation model and the language model, are estimated separately. In brief, the acoustic model assigns probability to the acoustic realization of a given word sequence, while the language model assigns probability to the word sequence in a given language. The pronunciation model is often a lookup table that maps words to its phonemic pronunciations. The three models are combined to create a full recognition model that maps observed acoustic sequences to word sequences. From this recognition model, the process of decoding extracts the most likely word sequence for a given input acoustic sequence.

In most systems, the acoustic and the language models are estimated independently, typically to optimize different cost functions. The acoustic models are estimated to minimize a surrogate differentiable function that approximates word error rate while the language models are estimated to maximize the probability that the model generates the training data. With the availability of large amounts of transcribed speech and cheap computational cycles, the historical reasons for this practical choice are no longer valid. There has been recently considerable interest in estimating them jointly [81, 53, 203, 205, 206]. We investigate the problem of jointly modeling the transition probabilities in the acoustic and the language models, along with the probabilities of duration cues.

Furthermore, even though speech recognizers have been employed in a variety of tasks or domains, where the sorts of sentences spoken and the style of speaking them vary considerably, each application requires careful tuning. A mismatch between training data and the application can degrade performance considerably. These mismatches may occur in one or more components of the speech recognizer, and adaptation methods are employed to tune each component separately. For example, domain adaptation is employed for language models [6, 10, 93], acoustic condition adaptation for acoustic models [135], and accent, dialect or speaking style adaptation for pronunciation models [179, 104, 44, 187]. In this context, we address the problem of porting a standard recognizer to a new dialect using two approaches: Explicitly by discriminative learning phonetic transformations, and implicitly with a discriminative acoustic state transition model, both of which are jointly trained with a discriminative language model.

Finally, we examine a spoken language application—scoring narrative retellings—where we investigate solutions to mitigate the effect of errors in the output of the speech recognizer. Automatic evaluation of narratives has wide variety of applications ranging from tutoring systems to neuropsychological examinations. We examine the problem of detecting a word or a phrase in the retelling which may be mutated due to lexical variants that the speaker chooses or the acoustic confusions that the ASR system outputs.

1.1 Research objectives

The key objectives of this thesis are to develop solutions to address the three problems mentioned above, and that are enumerated next.

1. Develop a framework for estimating jointly the parameters of the acoustic model and the language model to minimize the word error rate directly. The focus of this work lies in estimating the transition probabilities in the acoustic models, which are

typically ignored in a large vocabulary system, along with the weights of a standard discriminative language model. We also investigate the utility of incorporating duration information within this framework [97, 98].

2. Investigate methods to improve recognition performance on dialectal variants. We investigate two alternatives for the discriminative modeling of dialectal variations—one focused on learning phonetic transformations explicitly and the other focused on representing them implicitly at the acoustic state level by fine tuning state transition probabilities. In both cases, we learn the parameters jointly with the language model to directly minimize the word error rate.

3. Address the task of automatically scoring narrative retellings using models that take into account both the lexical choices from the user and the errors from the speech recognizer. Our goal is to develop a joint model that captures both variants within a discriminative estimation framework [96, 99].

1.2 Organization of the thesis

Before delving into the topic of the thesis, in Chapter 2, I provide the necessary background on fundamentals of speech recognition that are pertinent to the rest of the thesis, which can be easily skipped by those who are familiar with the technology. That chapter describes the components of speech recognizers in more detail, the metric employed to evaluate recognizers, weighted finite state transducers and their use in speech recognizers as well as the recipes we use for training a recognizer and decoding speech. Next, in Chapter 3, I look into the first research objective enumerated above. The joint discriminative model framework is evaluated extensively on a GALE Arabic transcription benchmark task. In Chapter 4, I focus on the second research objective, where I describe the extension of the joint discriminative model for adapting a general English speech recognizer to improve the recognition performance on African American Vernacular English (AAVE) dialectal speech. Chapter 5 is devoted to the third research objective, the automated scoring of narrative retellings from a clinical task, the Wechsler Logical Memory task. Finally, in the last chapter, Chapter 6, I conclude my thesis summarizing its key contributions and pointing to some future directions.

Chapter 2

Background

2.1 Components, output and evaluation of an ASR

The task of automatic speech recognition (ASR) is to decode an utterance x and hypothesize the spoken word sequence w. Mathematically, the process of decoding consists of searching over all word sequences and picking the word sequence $\hat{w}$ with the maximum posterior probability given the input utterance x.

$$\hat{w} = \underset{w \in \mathcal{T}}{\operatorname{argmax}}\, P(w|x) \tag{2.1}$$

The decoding is performed efficiently by searching over a compact representation of the acoustic realizations of all possible word sequences in a language or a task, the decoding graph $\mathcal{T}$. The paths in the decoding graph are weighted by the probability distribution of different word sequences. These weights are not obtained from a single model. Instead, they are obtained by combining different independent models, each one representing a specific source of knowledge. This is achieved by factoring the posterior probability $P(w|x)$ using Bayes' rule, and the process of generating the output is often referred to as the *maximum a posteriori* (MAP) decoding.

$$\hat{w} = \underset{w \in \mathcal{T}}{\operatorname{argmax}}\, \frac{P(x|w)P(w)}{P(x)} = \underset{w \in \mathcal{T}}{\operatorname{argmax}}\, P(x|w)P(w) \tag{2.2}$$

The term $P(x|w)$ is the acoustic likelihood, computed for a given input utterance x conditioned on a candidate word sequence w, while the term $P(w)$ is independent of the input and captures the prior probability of the word sequence. Hence, the two models are referred to as the acoustic model (AM) and the language model (LM), respectively. The acoustic model is evaluated by passing the input utterance as a sequence of high-dimensional real-valued vectors, and as a result the acoustic likelihoods have a large dynamic range compared to the prior probabilities from the language model. In practice, to reduce the disproportionate influence and incorrect independence assumptions of the acoustic models the likelihoods are scaled by a constant factor uniformly across all utterances and word sequence candidates [7, 40]. The probability from the denominator $P(x)$ is the same for all candidate word sequences, and so it is left out in the search.

In large vocabulary speech recognizers, the acoustic models are not modeled at the word level since a large proportion of words in the vocabulary are not seen frequently enough in the training data to allow robust estimation of their parameters. Instead, the acoustics are modeled in terms of smaller units. Phones (p_i) by themselves are too few and too coarse to capture all the contextual variations, for example, the difference in /t/ at the beginning (*tight*) and middle of a word (*butter*). These differences in realization are captured by taking into account the immediate phone contexts. These context-dependent phones (p_d) however are too numerous to be robustly estimated from the typical training data. For example, in English, which has about 40 phones, taking the two left and right contexts into account increases the inventory of acoustic units to $40^{(2+1+2)}$. They are instead mapped to a few thousand allophones using decision trees [199], as is explained later in Section 2.1.2.

Putting this all together, during recognition, the probability $P(x|w)$ is modeled in terms of several factors that are obtained by Bayes' rule and simplified using conditional independence assumptions.

$$P(x|w) = \sum_{p_i} \sum_{s} P(x, s|p_d) P(p_d|p_i) P(p_i|w) \tag{2.3}$$

The term $P(p_i|w)$ is the probability of the context independent phone sequence given the word sequence w. The mapping from word sequence to phone sequence is performed using a lexicon. The sequence of phones are mapped to context-dependent phones p_d deterministically. In turn, the states of the context-dependent phones are mapped to allophone states deterministically using the learned decision trees.

The sum over all the acoustic state sequences $\sum_s$ and phonetic sequences $\sum_{p_i}$ is approximated with Viterbi approximation [189, 148] by considering only the best acoustic state sequence for each phonetic sequence, and best phonetic sequence for each word sequence.

$$P(x|w) = \underset{p_i}{\operatorname{argmax}} \underset{s}{\operatorname{argmax}} P(x, s|p_d)P(p_d|p_i)P(p_i|w) \tag{2.4}$$

Figure 2.1 illustrates an ASR system with its components. (The illustration shows the simple case when acoustic models represent context independent phones).

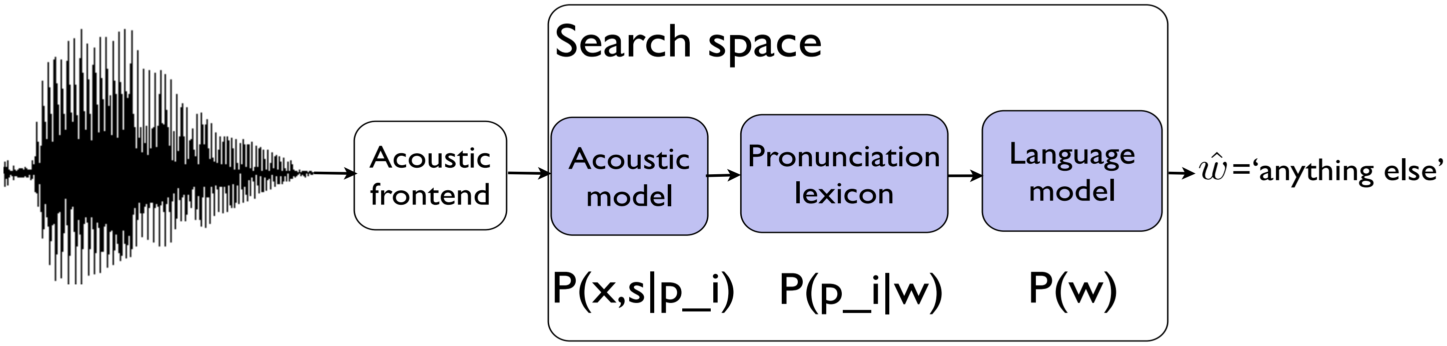


Figure 2.1: Automatic speech recognition (ASR) system.

2.1.1 Acoustic model

The acoustics of large vocabulary statistical automatic speech recognizers are modeled with context dependent hidden Markov models (HMMs) [9, 61, 148, 62, 57] (Figure 2.2 illustrates the HMM topology for a context dependent phone). Specifically, context dependent HMMs represent the joint probability distribution $P(x, s)$ over observation and acoustic state sequences x and s, respectively. The observations are vectors of acoustic features such as Mel frequency cepstrum coefficients (MFCCs) [27, 78] or perceptual linear predictions (PLPs) [54]. The joint probability is expressed in terms of the emission or observation probability $P(x_t|s_t)$, and the state transition probability $P(s_{t+1}|s_t)$ at time t:

$$P(x,s) = P(s_1) \prod_{t=1}^{t=T} P(x_t|s_t) \prod_{t=1}^{T-1} P(s_{t+1}|s_t) \tag{2.5}$$

where $P(s_1)$ is the initial state distribution.

The distribution of the acoustic vectors is represented with Gaussian mixture models (GMMs) [149] where $P(c|s)$ is the mixture weight, μ_{sc} the mean vector and Σ_{sc} the covariance matrix for the cth Gaussian of each state s.

$$P(x|s) = \sum_c \frac{P(c|s)}{\sqrt{(2\pi)^d |\Sigma_{sc}|}} \exp^{\frac{-1}{2}(x-\mu_{sc})^T \Sigma_{sc}^{-1}(x-\mu_{sc})} \tag{2.6}$$

In recent work, deep neural networks (DNNs) have been found to be more effective for modeling the observation probability than GMMs. They can capture more complex feature spaces with non-linear transformations [55, 29, 160]. In the case of the GMMs when observation vectors from the neighborhood are concatenated with the current vector, a linear transformation is applied to map the high dimensional vector into a lower dimensional space where the features are decorrelated.

In large vocabulary speech recognizers, typically the transition probabilities of the HMMs are modeled with only one parameter, the self-transition probability $a_{ii} = P(s_{t+1} = i|s_t = i)$, since the transition out of the state can be trivially computed from it ($P(s_{t+1} = j|s_t = i) = 1 - a_{ii}$ with $i \neq j$).

HMMs are very popular for sequence labeling because of their elegant mathematical framework and the well-defined algorithms that exist for their training and decoding. The mathematical elegance stems from two crucial conditional independence assumptions.

- Observations are conditionally independent given the hidden state. In speech recognition, often, the neighboring acoustic features are concatenated with the current features and the stacked vector is treated as the current observation [167].

- The hidden states have a Markov dependency. States are independent of previous states given the most recent state. This assumption poses difficulty in modeling segmental effects such as duration. Explicit duration models are often used to alleviate this weakness [147, 200].

It has been argued that the conditional independence assumption poses a severe limitation on HMMs [40]. The adjacent frames are, in fact, not conditionally independent due to the mechanics of speech production and influence of speaker characteristics on the speech signal.

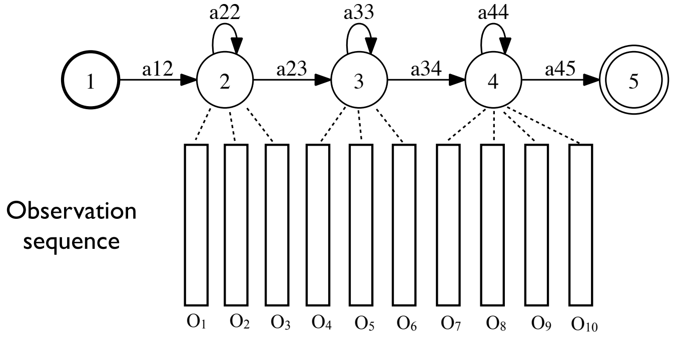


Figure 2.2: Acoustic modeling with 3-state left-to-right HMM topology.

2.1.2 Context dependent decision trees

The difficulty of estimating the 3-states of a left-to-right HMM representing a context-dependent (CD) phone is typically solved by clustering acoustically similar CD states into far fewer allophone states. These clusters are learned from the data using decision trees [17] whose partitions are defined using linguistically motivated questions about the membership of a phone in different phonetic categories such as vowels, consonants, affricates and so on. To be precise, for each of the three states of each phone, all the sufficient statistics for all the corresponding context-dependent phone-states are pooled. Clustering is performed in a top-down manner, by greedily splitting the states that maximize the likelihood. The likelihood is computed using a single multivariate Gaussian model for each partition or leaf of the tree in terms of the sufficient statistics. Figure 2.3 illustrates the decision tree for the first state of phoneme /aa/. If the phoneme /aa/ is in the beginning of the word (word boundary symbol | to the left) and its phone to the right is liquid, for example, /l/, the observation model (the GMM) whose identifier is 1000 is used to represent the acoustics of the first segment of phoneme /aa/.

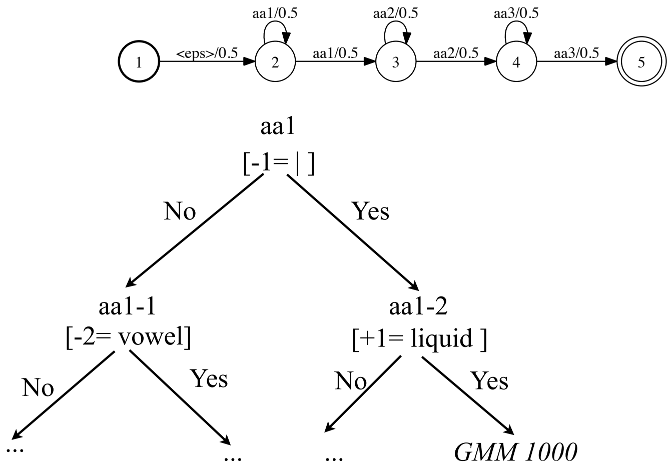


Figure 2.3: Context dependent decision tree for the first state of phoneme /aa/.

In practice, the correspondences between frames and states are not given, so decision trees are learned using an alignment generated by a simpler model such as a context-independent model. Subsequently, the decision trees are improved iteratively using alignments generated from more complex models. Thus, the decision tree maps the sequence of context-dependent phones to allophone states.

2.1.3 Pronunciation lexicon

The pronunciation lexicon is the intermediate layer between the acoustic models (AMs) and language models (LMs) that maps words into their phonemic representation. It is typically a one-to-one mapping from words into their canonical pronunciation. The choice of lexicon depends on the task, and sometimes a few variants are allowed for a handful of words like “and” and “the”. Allowing variants for more words jeopardizes recognition accuracy due to the acoustic confusability introduced. Furthermore, many of the pronunciation variations are implicitly captured by training the AMs with domain data [67].

2.1.4 Language model

Recall that language models (LMs) are the priors $P(w)$ necessary for the *maximum a posteriori* decoding, represented in Equation 2.2. The language models are also employed in other noisy channel models in natural language processing such as statistical machine translation and tagging. In practice, the LMs have a disproportionate influence on the performance of the recognizer and are trained with examples that match the task.

They assign a score to the word candidate that represents the probability of observing the word sequence in the given language or recognition task.

$$p(w) = \prod_{i=1}^{N} P(w_i|w_1, ..., w_{i-1}) \tag{2.7}$$

As it is not feasible to keep all possible histories for the words from the vocabulary, a Markov assumption is applied to reduce the history and retain only a fixed number $(n-1)$ of most recent previous words in an n-gram model. The probability $p(w)$ is approximated to:

$$p(w) \approx \prod_{i=1}^{N} P(w_i|w_{i-n+1}, ..., w_{i-1}) \tag{2.8}$$

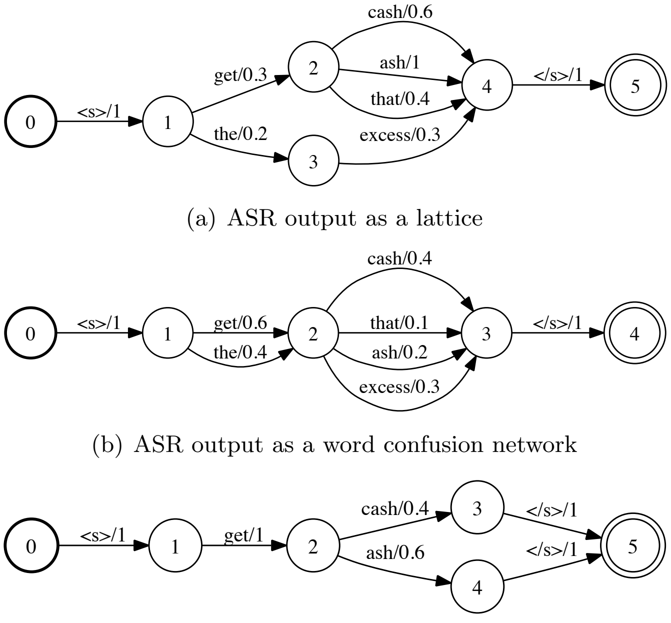
These probabilities are estimated by a maximum likelihood criterion, which is simply the frequency of the word given the history in the training data. During test time, however, a user may utter n-grams not seen in the training data. In order to avoid assigning zero probability to such n-grams, the maximum likelihood trained language models are smoothed using, for example, Kneser-Ney smoothing [21].

As an alternative to the popular n-gram language models people presented recurrent neural network-based LMs. These models have lately shown state-of-the-art performances [115, 173].

2.1.5 ASR output

The spoken words recognized by an ASR system are often processed further, for example, in named entity recognition, spoken language understanding and spoken question-answering systems, or re-evaluated in a multi-stage recognizer. Often, the correct candidate may not be the best (the highest scoring) candidate from the recognition, but it may still be present in the top 50 or 100 candidates. Since there is a significant overlap in words from the top candidates, they are often represented in a more compact form than N-best candidates (Figure 2.4(c)). Figure 2.4 illustrates storing them as lattices 2.4(a) or word confusion networks (WCN) 2.4(b).

The most general form for representing ASR output is a lattice whose paths represent different candidates. The nodes in the lattice correspond to boundaries between words. In word confusion networks, the boundaries between words are aligned using heuristics and words are binned within them, making them more compact than lattices. The binning introduces spurious paths that were absent in the lattices and as a result they have the lowest oracle error rate, that is, they often contain the candidates with the lowest error rate (oracle). The binning of words allows direct comparison between words in each bin and the words with highest posterior probability can be selected to improve ASR performance [109].



(a) ASR output as a lattice

(b) ASR output as a word confusion network

(c) ASR output as an N-best

Figure 2.4: ASR outputs

2.1.6 ASR evaluation

Speech recognition performance is measured in terms of the word error rate (WER), computed by aligning the ASR output with the reference transcription by dynamic programming. The WER includes three kind of errors: Insertions, deletions and substitutions. If the reference transcript contains N words and the ASR output has I insertions, D deletions and S substitutions, the WER is computed as:

$$WER(\%) = 100\frac{S + D + I}{N}\% \tag{2.9}$$

The edit distance metric rewards exact match and this may unnecessarily penalize words with equivalent alternative spellings. This is often alleviated by using the scoring tool developed by the National Institute of Standards and Technology (NIST) [132] which normalizes the typographic form of words or phrases.

2.2 Weighted finite state transducers

Word lattices, described earlier, are often represented as weighted finite state machines. In fact, a generalization of such machines, the weighted finite state transducers (WFSTs) are widely used in speech recognition and also in related spoken language processing tasks.

2.2.1 Formal Definition of weighted finite state transducers

A weighted finite state transducer consists of a set of nodes and directed arcs. The arcs contain input and output symbols, along with a cost for traversing the arc. These elementary costs are then combined to obtain the total cost of traversing one or more arcs. The algebra of combining the costs is defined using a semiring.

A system $(K, \oplus, \otimes, \bar{0}, \bar{1})$ is a semiring if it fulfills the following conditions [123]:

- It is specified by a set of values K, two binary operations $\oplus$ and $\otimes$, and two designated values $\bar{0}$ and $\bar{1}$.

- The operation $\oplus$ is associative, commutative, and has $\bar{0}$ as identity.

- The operation $\oplus$ is associative and has identity $\bar{1}$

- The operation $\otimes$ distributes with respect to $\oplus$

  $\forall x, y, z \in K : x \otimes (y \oplus z) = (x \otimes y) \oplus (x \otimes z)$. Left distributivity

  $\forall x, y, z \in K : x \otimes (y \oplus z) = (x \otimes y) \oplus (x \otimes z)$. Right distributivity

- The operation $\otimes$ has $\bar{0}$ as annihilator: $\forall k \in K, k \otimes \bar{0} = \bar{0} \otimes k = \bar{0}$

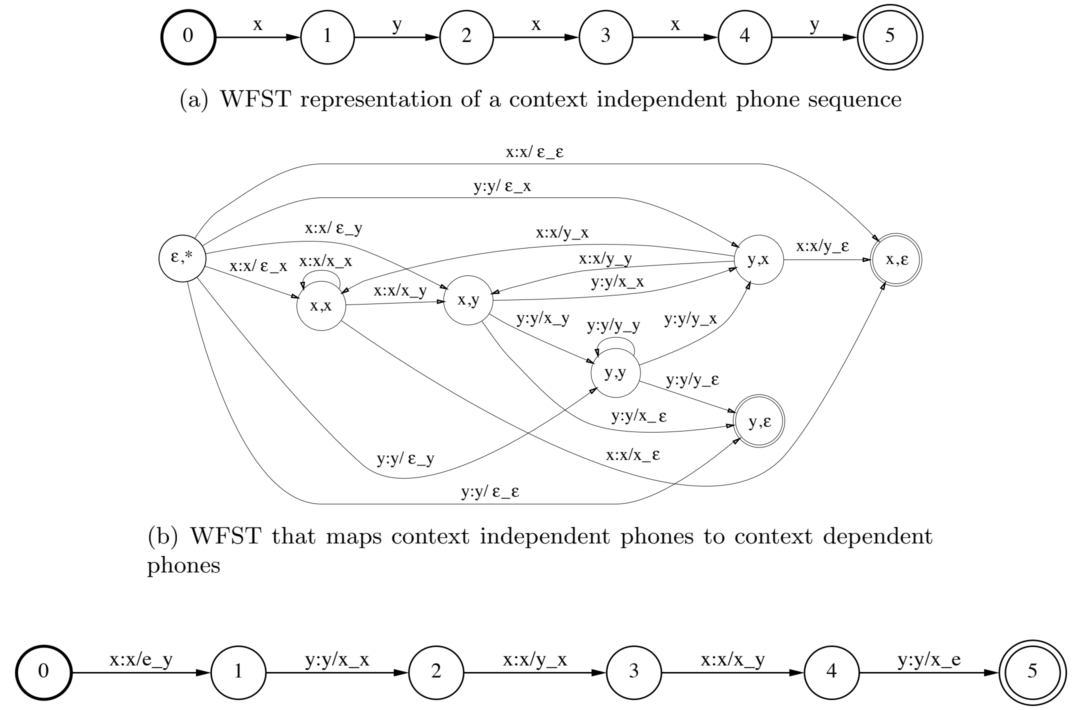
Given a semiring $(K, \oplus, \otimes, \bar{0}, \bar{1})$ a weighted finite state transducer (WFST) can be formally defined as an 8-tuple $(\Sigma, \Delta, Q, I, F, E, \lambda, \rho)$, where Σ and Δ are the finite input and output alphabet, respectively, Q the finite set of states, $I \in Q$ the start state, $F \subseteq Q$ the set of final states, E the set of transitions, and λ and ρ the weights for the initial and final state, respectively.

Weighted finite state acceptors (WFSAs) are a special case of WFTs which contain only input symbols and match or reject input sequences.

2.2.2 Common operations with weighted finite state transducers

- **Composition** is the operation used to combine several WFSTs in a single WFST. Each WFST factor or component represents an intermediate binary relation of symbols. For example, to get a WFST that maps context independent phone sequences into context dependent phone sequences we can compose the WFST representation of the context independent phone sequence from Figure 2.5(a) with a WFST that has context independent phones as input symbols and context dependent phones as output symbols, as shown in Figure 2.5(b). The output is a new WFST that has the context independent phone sequence on the input side and its context dependent representation on the output side (Figure 2.5(c)).

- **Determinization** is the operation that, broadly speaking, removes redundant outgoing arcs from a node. More specifically, a deterministic WFST contains at most one transition leaving any state with a given input symbol. For example, Figure 2.6(a) and Figure 2.6(b) illustrates a non-deterministic WFST and its equivalent deterministic WFST, respectively. Two WFSTs are equivalent when both of them assign the same output sequence and weight to each input sequence. However, the



(a) WFST representation of a context independent phone sequence

(b) WFST that maps context independent phones to context dependent phones

(c) WFST with context independent phone sequence on the input side and its context dependent representation on the output side

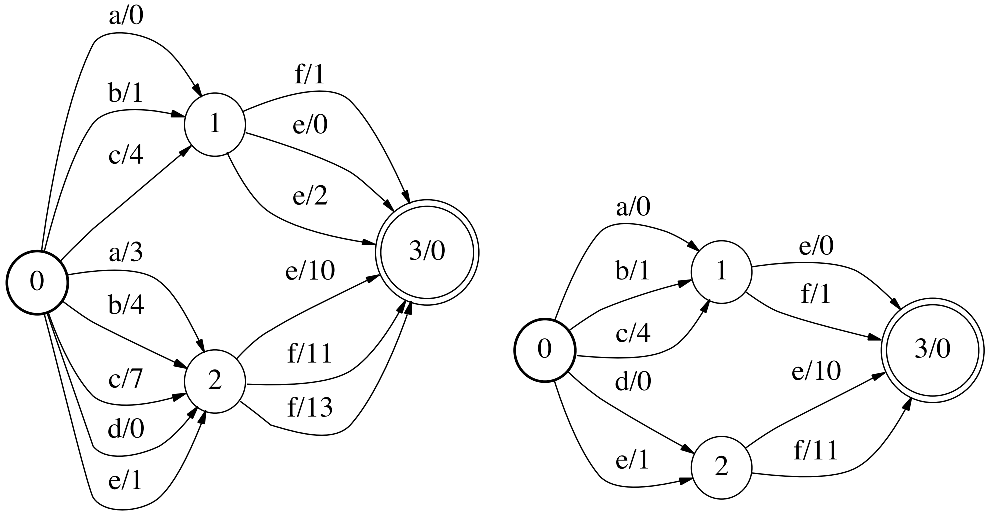
Figure 2.5: Example of composition operation with WFSTs (Example from Mohri and colleagues [122])

weights on the arcs along the path may be different in both WFSTs as long as the sums are the same over the paths.

- **Minimization** is the operation that reduces memory space requirements by representing a deterministic WFST with an equivalent WFST containing the minimum number of states and arcs. Figure 2.7 is an example of a deterministic WFST (Figure 2.7(a)) and its equivalent minimized WFST (Figure 2.7(b)). This can be viewed as a combination of prefix tree and suffix tree compression [119, 117].

2.3 Weighted finite state transducers in speech recognition

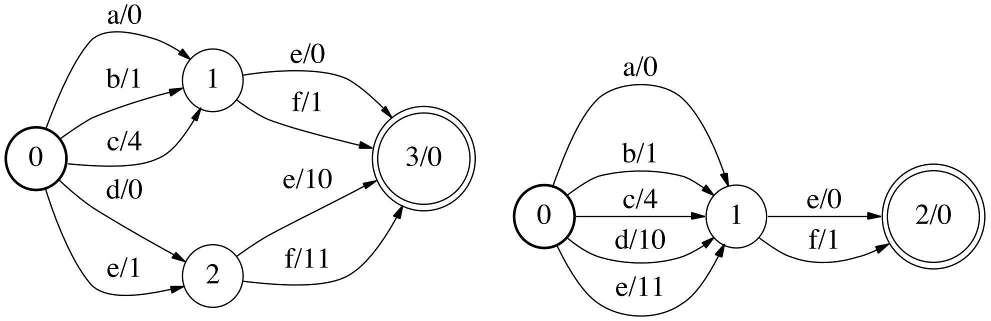
Weighted finite state transducers (WFSTs) have been widely adopted as efficient representations of a general speech recognition model and associated search space [121, 136, 118,



(a) Non-deterministic WFST

(b) Equivalent deterministic WFST

Figure 2.6: Example of determinization operation with WFSTs (Example from Mohri and colleagues [122])

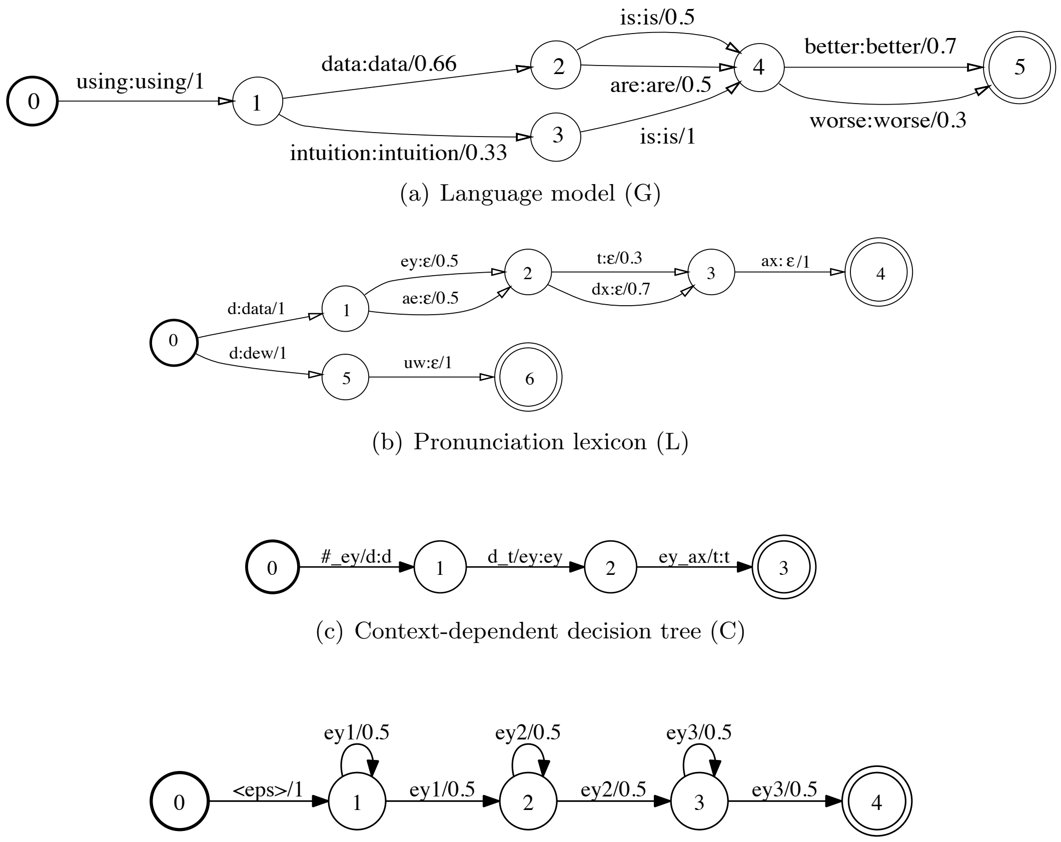


(a) Deterministic WFST

(b) Equivalent minimized WFST

Figure 2.7: Example of minimization operation with WFSTs (Example from Mohri and colleagues [122])

120, 126, 125, 124]. The WFST model for speech recognition is assembled or composed from separately estimated components that can also be represented as WFSTs. These components are typically the language model $\mathcal{G}$ representing the distribution of the word sequences, the pronunciation lexicon $\mathcal{L}$ that maps words into a sequence of phones, the decision trees for mapping phone sequences to clustered allophone sequences $\mathcal{C}$, and the topology of hidden Markov models (HMMs) $\mathcal{H}$ [122]. The graphical representation of each component is shown in Figure 2.8



(a) Language model (G)

(b) Pronunciation lexicon (L)

(c) Context-dependent decision tree (C)

(d) HMM topology (H)

Figure 2.8: Components of the speech recognizers in WFST format (Example from Mohri and colleagues [122])

After the composition of the WFST factors or components, the resulting WFST $\mathcal{T}$ is determinized and minimized to eliminate redundant intermediate paths to optimize the search space and the efficiency of the recognition.

$$\mathcal{T} = \mathbf{Min}(\mathbf{Det}(\mathcal{H} \circ \mathcal{C} \circ \mathcal{L} \circ \mathcal{G})) \tag{2.10}$$

The weights on a path through the WFST representing the decoding graph provide a cost, often negative log likelihood, incurred in choosing that mapping among all other possibilities. This cost includes the contribution from all the components modeling acoustics, pronunciation and language.

During the decoding of the speech utterance, the WFST-based speech recognizer scores the acoustic frames of the utterance with the HMM states, applies the states with the scores on the input side, chooses a path with the least cost and finally, reads off the word sequence on the output side of the chosen path. This decoding is very efficient since the redundant paths are reduced or eliminated in the precompiled search graph [69]. The decoder may generate the single best candidate or a distribution of word sequences, weighted by their probabilistic scores, in the form of a lattice or WFST, as depicted in Figure 2.4(a).

2.4 Large vocabulary speech recognition

Current state-of-the-art large vocabulary speech recognition systems are considerably complex. Both the training of the acoustic model and the decoding of an utterance are performed in multiple stages. Typically, the models are trained on a few hundred hours of speech in stages where each stage bootstraps a more powerful model than the previous stage using the improved alignments of spoken words to the waveform generated in successive stages. In this thesis, we adopt the training and decoding procedure described in previous papers [168, 167], implemented with the toolkit from IBM.

2.4.1 Training

Briefly, the acoustic model training procedure consists of the following stages.

Context-independent models. In this step we define the HMM topology for the phonemes by a three-state left-to-right topology and build a coarse context-independent phone model by dividing the observation sequence in each utterance uniformly between all the HMM states corresponding to the words in the transcript. From this alignment, a Gaussian observation probability is estimated for each HMM state across all utterances in the training data. This is then iteratively improved using Expectation-Maximization (EM) algorithm [28].

Context-dependent models. The context-independent models are employed to improve the alignment of the HMM states of the transcript of each utterance to its waveform. From these alignments, sufficient statistics are computed to estimate a linear discriminant analysis transform [47]. The phone labels are mapped to their corresponding context-dependent labels and sufficient statistics are computed for each context-dependent HMM state. As mentioned previously, a decision tree is learned from this data to cluster the states of these context-dependent phones to allophone states. The Gaussian observation models are then refined using the EM algorithm. During EM, additionally, a semi-tied covariance is estimated to alleviate the problems arising from using diagonal covariances in the Gaussian components instead of full covariances [36].

Vocal tract length normalized system. Since acoustic features are computed from the spectrum and the spectrum is affected by the vocal tract length of the speaker, the performance of the recognizer can be improved by warping the frequencies to map all the speakers to a normalized scale. The univariate warp factor is estimated by maximizing the likelihood of observing the warped acoustic features in the acoustic model. The best warp is computed by searching over a range of warps [32, 94].

Speaker adapted models. Variations that are related to the speaker and not the content are generally not useful in a speech recognition model. These variations are further removed from the model by transforming the speaker-specific space to a normalized space. The normalization may be performed in the feature space (FMLLR) or in the model space (MLLR) typically by estimating linear transforms that maximize the likelihood [38, 142]. In the case of MLLR, the Gaussian mixture models are clustered using a regression tree so that the models in a cluster can share the transformation matrix. Thus, the infrequent models may also benefit from the transformations.

Discriminative models. The models in the previous stages are estimated to maximize the likelihood of the training data. However, they are not necessarily optimal for recognition since we do not know the true form of the underlying models and we cannot estimate the parameters that achieve the global maximum. Discriminative models such as maximum mutual information (MMI) or its variant boosted MMI (BMMI) attempt to minimize a cost function that serves as a surrogate for word error rate. Word error rate is unfortunately not differentiable to obtain gradients for parameter optimization. Instead, MMI and BMMI maximize the conditional likelihood and have consistently improved the performance over maximum joint likelihood models [141].

2.4.2 Decoding

Mirroring training, the decoding is performed in several stages using the models from different stages of acoustic training. The acoustic models are combined with the language models and represented in a compact search graph as a WFST, as mentioned in Section 2.3. In certain cases, when the language model has tens or hundreds of millions of parameters (n-grams), the search graph becomes unwieldy. In such case, the search graph is factored into two factors, where the first one generates the lattices with a relatively simple language model and the second graph re-scores them with a more powerful language model using an on-the-fly composition operation. The decoders can generate 1-best candidates or a lattice in the form of a WFST [167].

Chapter 3

Joint acoustic, language and duration discriminative modeling for speech recognition

3.1 Discriminative modeling in speech recognition

Typically, the composite weighted finite state transducer (WFST) representing the decoding graph and obtained after composing the components of the speech recognizer is utilized directly in decoding speech without any attempt to optimize its overall weights.

Recall, the decision trees for mapping phone sequences to clustered allophone sequences $\mathcal{C}$ and the pronunciation lexicon $\mathcal{L}$ are deterministic mappings. The language models $\mathcal{G}$ and, until recently, the acoustic models are estimated to maximize the likelihood of the data. One problem with maximum likelihood (ML)-based training is that we do not directly minimize the objective function of the speech recognition, the recognition error rate.

Discriminative training aims to learn the parameters of the model by optimizing cost functions related to the recognition task, instead of maximizing the likelihood of the training data. It has lead to improvements in speech recognition, and therefore, it is now common practice to discriminatively reestimate parameters for the acoustic and language models. The reestimation is usually performed independently for each model, without any interaction between the parameters of the acoustic and lexical features.

Discriminative training, ideally, should optimize the objective function of the task or the word error rate directly. Because the word error rate is not differentiable and cannot be used to compute gradients with respect to the model parameters, the discriminative training of the acoustic models is often done with approximate surrogate functions that are differentiable, such as the maximum mutual information estimation (MMIE) or conditional maximum likelihood estimation (CMLE) [8, 184], minimum classification error estimation [66, 65], and minimum phone error or minimum word error estimation (MPE or MWE) [143, 139]. These estimation methods do not take into account the language model used in the decoding. Instead, language models are trained separately either using maximum likelihood or a discriminative criterion [156].

Previous work has shown the benefits of considering the interdependency between the acoustic and linguistic features by learning a model that directly estimates the most likely word sequence given an input speech utterance [85, 53, 24] or by discriminatively optimizing the weights from the composite WFST [103, 86, 81, 80].

Following this line, in this chapter I explain our approach to train discriminative models that jointly estimate parameters from the acoustic and language model. The proposed approach can be considered an extension of the discriminative language model (DLM) [156], which has the interesting property of estimating the parameters to minimize the word error rate directly. We extend this framework to not only learn the parameters of word transitions (n-grams) but also of acoustic state transitions in the HMMs, as well as the duration of the states. The proposed model is evaluated on GALE Arabic transcription task.

3.1.1 Discriminative language modeling for speech recognition

Many natural language processing (NLP) tasks can be approached with a source-channel paradigm. The noisy channel generates a collection of candidates for a given input, and the source model modifies the probability of these candidates by constraining the confusability or noise of the channel. In many applications the language model represents the source model, for example, in speech recognition. The noisy channel is the acoustic model that generates word sequence candidates for an input speech utterance, and the language model penalizes the word sequences that are unlikely in a given language. The confusability of the noisy channel is measured with an evaluation metric that depends on the task. In the case of speech recognition this metric is the recognition error rate, commonly the word error rate (WER).

As generative n-gram language models are trained with large amounts of text data independently from the channel model, the AM in this case, their parameters are typically reestimated via discriminative training. For that, speech utterances are decoded to get a set of candidate output sequences, and the parameters from the discriminative language models are learned with this set of candidates and the reference text by optimizing the classification or separability between the correct and incorrect candidates.

Linear models like the perceptron algorithm [26, 156], conditional log-linear models [156, 134, 164], minimum classification error models [84, 150] or minimum Bayes risk models [83] have been applied to train discriminative language models by minimizing an objective function directly related to the WER.

Global linear models

Global linear models have been successfully used for discriminative language modeling [156], and these are the models we also choose for training our joint models.

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

$$F(x) = \operatorname*{argmin}_{w \in \mathbf{GEN}(x)} \Phi(x, w)\bar{\alpha} \tag{3.1}$$

For large vocabulary speech recognition tasks, the global linear model typically contains millions of parameters and is trained over a few hundred thousand utterances, a setting for which the perceptron is a convenient option [156]. Let (x_j, r_j) be the jth training example out of M where x_j and r_j represent the acoustic input and its transcript (or the oracle—the candidate transcript with the minimum word error rate—in the lattice being re-scored), 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, the algorithm updates the parameters $\alpha_j^{(i)}$ when the best-scoring candidate $\bar{w}_j$ under the current model differs from r_j. Thus, the perceptron algorithm updates the model so that the score of the oracle candidate improves with respect to the competing candidate. Since the oracle by definition is the candidate with the minimum word error rate, the learning algorithm minimizes the word error rate.

Algorithm 1 :Averaged Perceptron(x,**GEN**(x),r)

```
$\alpha_M^{(0)} \leftarrow 0$
$i \leftarrow 1$
repeat
  $\alpha_0^{(i)} \leftarrow \alpha_M^{(i-1)}$
  for $j = 1$ to $M$ do
    $\bar{w}_j \leftarrow \text{argmin}_{w_j \in \mathbf{GEN}(x_j)} \Phi(x_j, w_j)\alpha_{j-1}^{(i)}$
    if $\bar{w}_j \neq r_j$ then
      $\alpha_j^{(i)} \leftarrow \alpha_{j-1}^{(i)} + \Phi(x_j, r_j) - \Phi(x_j, \bar{w}_j)$
    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.

Discriminative language model as a weighted finite state acceptor

The discriminative language model can be represented as a weighted finite state acceptor (WFSA) and combined with the composite WFST $\mathcal{T}$ representing the decoding graph, or in a post-processing step with the lattice containing the candidate output sequences [156].

Given an acoustic input x, let $\mathcal{L}_x$ be a word lattice generated by the baseline recognizer, as shown in Figure 3.1. The lattice $\mathcal{L}_x$ is an acyclic and deterministic WFSA, representing a probability distribution P_x over all strings $w \in \Sigma^*$, i.e., all possible transcriptions of x

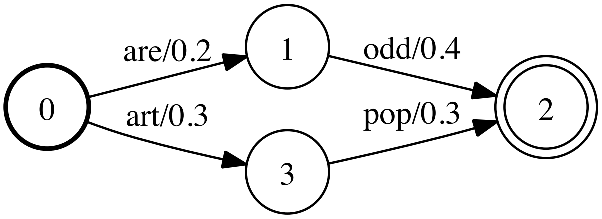


Figure 3.1: Output lattice from a speech recognizer as a weighted finite state acceptor.

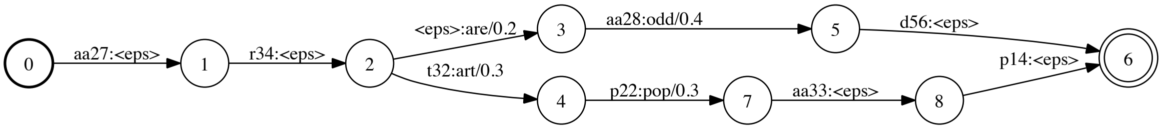


Figure 3.2: Output lattice from a speech recognizer as a weighted finite state transducer.

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 the log semi-ring, where the weights are negative log likelihood (where 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 [122] for details). The weights $w_{\mathcal{D}}[\pi]$ for all path $\pi \in \Pi_{\mathcal{D}}$ can be related to the linear form.

$$w_{\mathcal{D}}[\pi] = \sum_{j=1}^{d} \Phi_j(x, l[\pi])\alpha_j \tag{3.2}$$

Here, $l[\pi]$ is the concatenation of all the labels of the path π, $\Phi_j(x, l[\pi])$ for $j > 0$ is the count of the j'th n-gram in $l[\pi]$ (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. The log probability of the baseline recognizer is incorporated into $\mathcal{L}_x$ with a scaling factor α_0.

$$\operatorname{argmin} \Phi(x, l[\pi])\bar{\alpha} = \mathbf{BestPath}(\mathcal{L}_x \circ \mathcal{D}) \tag{3.3}$$

While the WFSA view is useful, the discriminative language model can be understood and applied without resorting to finite state operations. For re-scoring a candidate w, a feature vector $\Phi(x, w)$ consisting of n-gram subsequences from w 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 candidate w includes, for example, in the zeroth component of $\Phi(x, w)$, the log probability of w given x as evaluated by the baseline recognizer and contains the total contributions from both acoustic and language models. The feature vectors for all the candidates are evaluated using the discriminative language model (a linear model), and the best scoring candidate is chosen as the output of the rescoring pass.

Discriminative language models have been empirically demonstrated to improve speech recognition in English [156]. 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 [164, 5].

3.1.2 Joint acoustic and language modeling for speech recognition

Although the stochastic components of the speech recognizer, namely, the acoustic and language models are independently optimized via discriminative training, there has been previous work where the interaction between acoustic and language models have been explored. Printz and Olsen incorporated a notion of acoustic confusability into the language model [144]. They estimated the confusability between words by measuring the distance between their hidden Markov models using an approximate close form expression. However, the parameters of the acoustic and language model are not jointly estimated. For a small task, Kuo and Gao estimated direct models of the form $P(w|x)$ for speech recognition and estimated the parameters using maximum entropy criterion [85]. This approach was later applied to large vocabulary tasks [53, 203] where both acoustic and linguistic features are evaluated to output the best candidate for a given input. Examples of features include n-grams, word level templates, the HMM posterior probability and the dynamic time warp distance between the instance and a set of templates. Continuing this line 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 [203]. Their approach does not impose any structure. The classification of the speech utterance into a string of words is modeled directly. Therefore, it is more suited to recognize short utterances, specifically, in the Bing mobile voice search task that allows users to look for local businesses from their mobile phone. This approach was extended in the form of segmental conditional random fields (SCARF) to make it feasible to a broader set of applications [203, 205, 206] such as Broadcast News (BN) and Wall Street Journal (WSJ) tasks.

A few researchers have adopted the weighted finite state transducer framework and modified the weights on the arcs of the decoding graph after composing the acoustic and language model components [103, 86]. The key idea is to minimize the cost of the path with the minimum word error rate compared to other competing paths. Lin and Yvon defined a minimum classification error (MCE) criterion using a sigmoidal function defined over the weights of the graph [103]. After setting the slope and the threshold of the sigmoid, they refined the weights of the graph iteratively to optimize the MCE over the training data. They demonstrated the 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 [86]. As 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.

Kubo and colleagues extended the idea of optimizing the transition weights of the decoding graph by also optimizing the acoustic information in the arcs from the graph. They used the raw acoustic MFCC vectors as acoustic features and validated their approach in the MIT lecture task and the WSJ transcription task [81]. They also explored combining their approach with acoustic features from deep neural networks (DNNs), first for phone recognition task on TIMIT [79] and very recently for a large vocabulary speech recognition task [80]. They show that even in speech recognizers with stronger AMs like DNNs the joint discriminative linear models provide gains over the baseline.

Recently, Abdelhamid and Abdulla have shown that jointly optimizing the acoustic and language model parameters from the decoding graph by MCE-based discriminative training improves the performance under noisy conditions [2].

Gales and colleagues provide a detailed overview of the practices applied in the ASR field regarding discriminative models, criteria and features [37].

Discriminative joint acoustic and lexical model as weighted finite state transducers

Based on the observation that the lattices provided by the speech recognizer are actually weighted finite state transducers (WFSTs) with information from the input and output side of the decoding graph (see Figure 3.2), our approach consists of extracting the available acoustic and lexical information (features) for the discriminative models.

Consider the general finite state model of a speech recognizer, composed of weighted finite state components [122]. 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 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 $w \in \mathcal{W}$ (as before) and the input side represent acoustic state sequences $s \in \mathcal{S}$ associated with the path, as shown in Figure 3.2 and as identified by the backtrace of the Viterbi algorithm in the baseline recognizer. The state sequence may be augmented by prosodic features such as duration, or phonological features such as pronunciation variants from the Viterbi path.

In our joint model, we expand the feature representation, from $\Phi(x, w)$ to $\Phi(x, s, w)$, to include features defined over the input acoustic state sequence s in addition to those defined over w. The lattice $\mathcal{L}_x$ may, in general, have multiple acoustic state sequences $s \in \mathcal{S}_w$ associated with an output sequence w. Once again, given a function **GEN** which enumerates a set of candidates $\mathbf{GEN}(x)$ for an input x (e.g. N-best candidates or lattices), an acoustic state sequence s associated with w, a representation Φ mapping each $(x, s, w) \in \mathcal{X} \times \mathcal{S} \times \mathcal{W}$ to a feature vector $\Phi(x, s, w) \in \Re^d$ 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 search over all acoustic state sequences s associated with candidates w of an utterance x.

$$F(x) = \underset{s,w \in \mathbf{GEN}(x)}{\operatorname{argmin}} \; \Phi(x, s, w)\bar{\alpha} \tag{3.4}$$

The features for our model consist of n-grams extracted from not only w but also s. For each path, we concatenate the n-grams from s with n-grams from w; $\Phi(x, s, w) = [\Phi_s(x, s, w) \;\; \Phi_w(x, s, w)]$. In a large vocabulary system, the number of acoustic states (automatically clustered allophone states) is 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 only be marginally higher than those of discriminative language models.

The joint discriminative model can be represented in terms of finite state machines, factored into two WFSTs (or WFSAs). The parameters $\bar{\alpha}_w$ corresponding to the features of w can be converted to $\mathcal{D}$, the discriminative language model, as before. Similarly, the parameters $\bar{\alpha}_s$ corresponding to the acoustic component can be converted to another WFST, $\mathcal{E}$. Then, the output of the model $F(x)$,

$$\underset{s,w \in \mathbf{GEN}(x)}{\operatorname{argmin}} \; \Phi_s(x, s, w)\bar{\alpha}_s + \Phi_w(x, s, w)\bar{\alpha}_w \tag{3.5}$$

can be computed using finite state operations.

$$F(x) = \mathbf{BestPath}(\mathcal{E} \circ \mathcal{L}_{\mathbf{x}} \circ \mathcal{D}) \tag{3.6}$$

While re-scoring, 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 resulting WFSA is the output of the joint discriminative model. For the experiments $\mathcal{L}_x$ is constrained so that it contains only one acoustic state sequence s for each w. However, in the most general case, multiple acoustic state sequences may be associated with each w and the evidence from all the associated acoustic state sequences s needs to be summed probabilistically. This can be achieved through standard finite state machine (FSM) operations, specifically, minimization and determinization.

$$F(x) = \mathbf{BestPath}(\mathbf{Min}(\mathbf{Det}((\mathcal{E} \circ \mathcal{L}_{\mathbf{x}} \circ \mathcal{D})))) \tag{3.7}$$

Alternatively, instead of applying the WFSAs in a rescoring phase, they could be incorporated into the decoding graph; the WFSA $\mathcal{E}$ representing the discriminative acoustic model would be composed with the decoding graph on the input side, and the WFSA D describing the discriminative language model on the output side.

$$\mathcal{T}' = \mathbf{Min}(\mathbf{Det}(\mathcal{E} \circ \mathcal{H} \circ \mathcal{C} \circ \mathcal{L} \circ \mathcal{G} \circ \mathcal{D})) \tag{3.8}$$

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

3.2 Experimental setup

3.2.1 Task: GALE Arabic Transcription task

The proposed model was evaluated empirically on GALE (Global Autonomous Language Exploitation) Arabic Transcription task. The purpose of the DARPA GALE program was to make Arabic and Chinese speech and text accesible to monolingual English speakers, particularly in military setting, which involves speech recognition and machine translation techniques [25].

I now detail characteristics of the Arabic language and challenges that these characteristics bring for building Arabic automatic speech recognizers. Semitic languages—of which Arabic is the most widely spoken language—are characterized by absence of some vowels and other pronunciation information in their written form. The consonants carry most of the semantic information. The root of the words are a set of consonants and the words are obtained by inserting vowels, consonants that mark grammatical functions (these consonants may be used as prefixes, suffixes or infixes) and gemination in between the root consonants [159]. For example, *k-t-b* is the root for the meaning *write*, and if we fill it with different vowel patterns we get words such as *kitaab (book)*, *kutub (books)* or *kaatib (writer)* [76]. Arabic syllables must contain at least one vowel, and the allowed syllables are CV, CVV, CVC, CVVC, CVCC where C indicates a consonant and V a short or long vowel [159].

Common prefixes that are appended to the words are the definite article *al*, and conjunctions and prepositions like *fa (so, and)* and *li (for, in order to)*, respectively. Suffixes are used, for example, to define possessives; *kitaab-iy (my book)* , *kitaabu-ka (your (masc) book).* The way words are derived in Arabic – filling in consonant roots with vowel patterns, and appending affixes, for example, to mark grammatical categories – generates a huge number of word forms [76].

The formal written Arabic variety is Modern Standard Arabic (MSA). However, there is great variation in spoken form, especially in casual or everyday communications. Arabic dialects can be divided in different dimensions. Biadsy and colleagues [15] suggest one of the possible ways of grouping them; Gulf Arabic, Iraqi Arabic, Levantine Arabic, Egyptian

Arabic, and Maghrebi Arabic.

The standard Arabic alphabet contains 28 consonantal sounds, 2 of which sometimes act as vowels, depending on the context [159]. Letters may have up to four different shapes depending on their position in the word: Word-initial, medial, final, and separate. Short vowels and other pronunciation phenomena are marked with diacritics, short strokes placed either above or below of the preceding consonant [76]. The different pronunciation phenomena indicated by diacritics are the absence of a vowel, and the consonant and short vowels "doubling", gemination and nunation, respectively. Nunation is phonetically realized by adding *n* after the final vowel [92]. In total, Arabic has 8 diacritics, illustrated in Table 3.1, that are not usually written in texts.

Diacritic	Symbol name	Meaning
Short vowels	fatHa	/a/
	kasra	/i/
	Damma	/u/
Shadda	shadda	consonant doubling
Sukuun	sukuun	vowel absence
Tanween	tanween al-fatHa	/an/
	tanween al-kasr	/in/
	tanween aD-Damm	/un/

Table 3.1: Diacritics in Arabic.

Although the grapheme-to-phone conversion is mainly a one-to-one mapping, the absence of diacritics introduces a high degree of phonetic ambiguity [76].

The described characteristics make the automatic speech recognition of Arabic challenging due to:

- Script form and representation: Most of the text material that is available in Arabic is in MSA form, but the regular Arabic form used for the oral communication is one of the dialectal Arabic. Therefore, the training data for the acoustic and language models do not match. Furthermore, both the text material and the transcripts of the acoustic data do not usually contain information about diacritics. Due to the absence of short vowels and other pronunciation phenomena, the acoustic models are not closely related to the surface form realization. Previous work shows that adding diacritics information into the training data of the AMs yields better word error rates [186, 92, 168]. Similarly, including diacritics also improves the performance of the LMs [114]. When the language model is trained with non-diacritized text more linguistic contexts are assigned to each word form and therefore, there is more confusability during decoding.

- Many dialectal varieties: Arabic is usually spoken in dialectal form and as mentioned earlier there are many dialects. The dialectal variations are modeled implicitly by the acoustic model and consequently the acoustic states represent more than allophonic variations.
- Morphological complexity. Many word forms are possible due to the rich morphology of Arabic. Hence, it is more difficult to estimate the language model probabilities, and the out-of-vocabulary rate (OOV) is considerably higher than many other languages [75].

3.2.2 Data

The training data for our models consists of about 200 hours ($\approx$ 1.5M words) of Arabic Broadcast Conversations picked from the 1,000 hours of data used to train the acoustic models. The models were evaluated on the 20-fold cross-validation of the training data as well as on 2 independent test sets, namely, the 2007 GALE development set and the 2007 GALE evaluation set, distributed by NIST. The development set consists of 55 programs, totaling 2.6 hours ($\approx$ 18K words) from 10 Broadcast News (BN) and Broadcast Conversation (BC) sources, and the evaluation set contains 65 programs, totaling 2.8 hours ($\approx$ 20K words) from 11 BN and BC sources [168].

3.2.3 Baseline system

The baseline acoustic models, trained on about 1,000 hours of Arabic broadcast data, contain 45 phones including long vowels, a 3-state left-to-right HMM topology for phones, 5,000 clustered pentaphone states (its context includes word boundaries and pauses), a linear discriminant transform [47], and a semi-tied covariance transform, described in detail in [168]. The acoustic features consist of vectors with 13 perceptual linear prediction coefficients with speaker-specific vocal tract length normalization (VTLN). The HMM observations are computed by splicing 9 consecutive frames into a 117-dimensional supervector, then projecting to 40 dimensions using linear discriminant analysis (LDA) or heteroscedastic discriminant analysis (HDA) [168].

The baseline 4-gram language model with 122M n-grams, was estimated by interpolating 14 components. The vocabulary is relatively large at 737K and the associated dictionary has only single pronunciations that are modeled by the sequence of letters in the word. The short vowels are not explicitly modeled, and it is assumed that speech associated with the short vowels will be implicitly modeled by the adjacent phones [168].

To reduce the computational cost of decoding the training data, speaker adaptation and cross system adaptation were not used, and hence, the results of the baseline reported in section 3.2.7 correspond to an intermediate VTLN stage in [168], the IBM GALE system for the second project year. Discriminative models are most effective when the models used for training and testing are matched, so the proposed models are compared with the performance of the same VTLN model on the test sets.

3.2.4 Generation of competing candidates

For training the proposed models, 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 candidates were used to form the competing candidates. For each candidate, we also extracted the single best acoustic state sequence (instead of summing over all the acoustic state sequences for a given word sequence). Recent improvements in the efficiency of lattice generation methods enable us to produce deterministic

WFST lattices with unique word sequences and their associated lowest cost acoustic state sequences [165, 169, 140].

3.2.5 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 candidates 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 the candidates with lowest word error rate (WER) and are likely to overwhelm the other features [176]. 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.

3.2.6 Feature space

The features for our model can be understood easily by considering an toy example.

- Word sequence (output side): <s> are odd </s>
- Acoustic state sequence (input side): <s>,1000,1000,4546,4789,...,1000,1000,4546

The feature space for our joint model is illustrated in Table 3.5, where Φ_w and Φ_s represent the n-gram subsequences or features on words w and acoustic states s, respectively. The numbers from the acoustic state sequence represent one of the three states of an allophone. Figure 3.3 represents the sequence of acoustic states for the phone [aa] in a specific context; the acoustic state 1000 is representing the GMM that models the spectral distribution of the initial segment of the sound, the state 4546 the GMM for the middle part, and 4789 the corresponding GMM for the last part.

3.2.7 Results

The discriminative models were evaluated on 20-fold cross-validation of the training set (Xval) and on two independent test sets (Dev07 and Eval07) using just lexical features

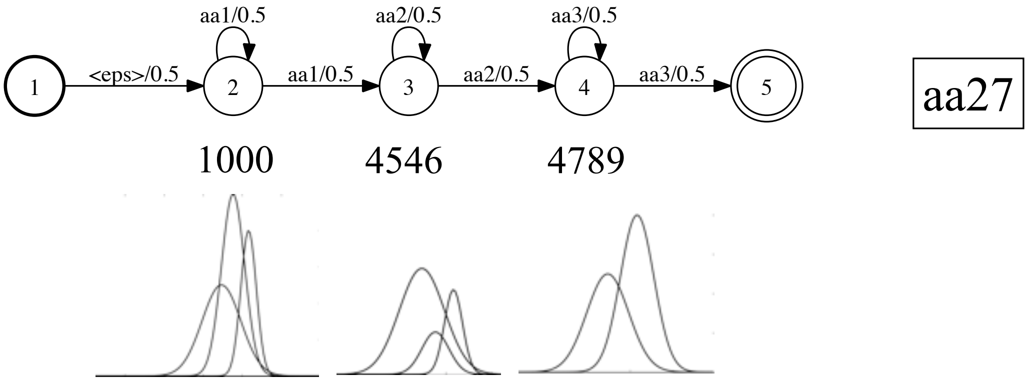


Figure 3.3: An example allophone of phoneme /aa/.

Φ_w	Φ_s
<s>,are=1	<s>,1000=1
are,tone=1	1000,4546=2
...	...

Table 3.2: Features corresponding to the word and acoustic state sequences associated with a given candidate.

Φ_{w2} (discriminative language model), and using acoustic and lexical features [Φ_{s2} Φ_{w2}]. The n-gram features are encoded so that they also include the lower order n-grams [156]. For training, 100-best unique candidates were extracted and the resulting models were evaluated on 1000-best unique candidates. Although the number of potential bigram features is large—the generative LM contains 60.8M bigrams—the averaged perceptron algorithm estimates non-zero weights for only those that are observed in the training data and are useful in discriminating the oracle from the competing candidates. The number of such active features or parameters for each model, F_U, is listed in the second column in Table 5.9. To provide an insight of how often those parameters were used in the test sets, their coverage F_C is listed for Dev07 and Eval07 sets. 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 (an example in English would be *gonna==going to*) which was available from NIST [132] 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 [132].

System	$\#F_U$	Xval	$\#F_C$	Dev07	$\#F_C$	Eval07
Baseline	-	26.6	-	21.2	-	24.0
Φ_{w2}	1.13M	26.2	24.74M	20.9	28.34M	23.8
Φ_{w2}, Φ_{s2}	1.47M	25.7	492.59M	20.6	570.91M	23.4

Table 3.3: Comparison of discriminative models with language and acoustic components, measured (WER) on 20-fold cross-validation, Dev07 and Eval07.

As reported in Table 5.9 the bigram discriminative language model has about 1M parameters and provides a consistent gain of about 0.2%-0.4% across the three sets, similar to the gains observed in morphologically-rich languages such as Turkish [5]. The gain obtained from the discriminative language component is modest because the large vocabulary size of morphologically-rich languages increases the out of vocabulary rate. Thirty-six percent and 40.5% of the features from the discriminative model are active on Dev07 and Eval07, respectively. In both sets, these features constitute 73% of all the features.

The addition of the acoustic component to the model improved the gain of about 0.6%-0.9%. The size of the discriminative model increases just by a third, but the number of total active features in the test sets is over 98%. Furthermore, acoustic features are exercised more often than lexical features, about 500M compared to 25M, contributing disproportionately to the total score of the discriminative model.

3.3 Incorporating duration features into the discriminative model

In most large vocabulary speech recognizers, elementary units of sounds, namely allophones or context dependent phones are represented by HMMs in a simple left-to-right linear topology with about 3 hidden states.

The joint probability $P(x, s)$ over observation and acoustic state sequences x and s, respectively, is represented by the context dependent HMMs as a product of two probabilities; the emission probability and the state transition probability:

$$P(x,s) = P(s_1)\prod_{t=1}^{t=T} P(x_t|s_t)\prod_{t=1}^{T-1} P(s_{t+1}|s_t) \tag{3.9}$$

The emission distribution, with Gaussian mixture models (GMMs), models the spectral variability of the speech, and the transition distribution models the time variability of the speech. In particular, the transition probabilities represent the duration of the context dependent phonemes. In a linear topology, a state with self-loop probability a_{ii} ($P(s_{t+1} = i|s_t = i)$) models the probability of observing d frames, $P(d) = (1 - a_{ii})a_{ii}^{d-1}$, which is a geometric distribution (exponential in nature) whose most likely duration is zero. This is clearly incorrect. State duration distributions are better fitted to Poisson [157] or gamma distribution [101]. The duration modeled by a linear HMM, assuming the states are independent is given by the convolution of the component distributions and results in a negative binomial distribution, which can be regarded as a discrete version of the gamma distribution. However, the resulting distribution is very limited in its capacity to represent a wide variety of distributions, since the geometric distributions only depend on a single parameter that defines the mean and variance of the distribution. The convolution of three gamma distributions also results in a gamma distribution, but with more freedom, since the gamma distribution of each state is now defined by two parameters [147]. Furthermore, even if the distribution were the right one, when transition probabilities are multiplied with the emission or observation probabilities to get the acoustic likelihood, the contribution of the transition state probability is negligible, since the dimension of the observation vectors is much larger (40 dimensional in our task) than the dimension of the transition state vectors (bidimensional).

Instead of modeling duration implicitly through the transition probabilities, a number of extensions attempt to model them explicitly, extensions that Yu reviews as variants of hidden semi-Markov models (HSMMs) [200]. He outlines HSMM variants such as explicit duration HMM [102], variable transition HMM [185] and residential time HMM [201]. Russell and Cook [158] modified the topology of the regular HMM model by representing each HMM state with a sub-HMM. They extended the constrained left-to-right topology allowing transitions that skip states.

These approaches have shown modest improvements in controlled tasks. One reason may be because duration parameters are estimated in a suboptimal way. The parameters of the model are estimated to maximize the likelihood of durations observed in the training data, and the estimation is done in isolation, without taking into consideration their interaction with other components of the ASR system. In our approach the parameters of the duration models are estimated jointly with the parameters from the language and acoustic models to minimize the word error rate. We include duration cues related to acoustic states encoded using simple discrete features into the joint discriminative model presented in Section 3.1.2.

3.3.1 Duration cue in Arabic language

In Arabic, phonemic lengthening like vowel lengthening or consonant doubling (gemination) distinguishes minimal pairs—/hal/ (he solved) vs. /ha:l/ (condition), /sid/ (pay/close) vs. /si:d/ (grandfather), /hibi/ (he crawled) vs. /hibbi/ (you love) and /ali/ (name) vs. /a:li/ (high) [116]. However, these phonemic characteristics are not specified in written form; texts usually contain just consonants and long vowels. This lack of information increases the confusability of the ASR because both the acoustic model (AM) and the language model (LM) have to handle more ambiguities.

In previous work Lamel and colleagues [92] used duration information for the GALE Arabic transcription task. Lattices were re-scored in a post-processing step by augmenting the HMM likelihood with the duration likelihood. They modeled word and phone durations with Gaussian mixture models (GMMs). Phone duration models were used as back-off models, when there was not enough data for the word duration models. They got modest gains of 0.2-0.4% on GALE 06 development data.

Although duration features can be modeled at the word, phone, or acoustic state level, when the vocabulary is large, as in morphologically-rich languages, 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. Modeling it at the phone level, on the other hand, is too coarse. Clustered allophone or acoustic states provide a reasonable compromise between these two extremes, so we choose to model their durations. Note that 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 [35, 18].

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. We therefore encode each observed duration of an acoustic state as a separate feature with its own model parameter.

Considering again the example from Section 3.2.6:

- Word sequence (output side): <s> are odd </s>
- Acoustic state sequence (input side): <s> 1000,1000,4546,4789,...,1000,1000,4546

The acoustic state sequence can be rewritten as a sequence of acoustic state labels and their durations:

<s>,$1000_2, 4546_1, 4789_1, ..., 1000_2, 4546_1$ where the subscripts 2, 1, 1, ..., 2 and 1 are representing the durations of each acoustic state in the sequence. From this sequence we extract duration features in the form of n-grams in the same way as done previously in Section 3.2.6 for the lexical and acoustic state features. The subsequence —$1000, 1000, 4546$— is represented as a duration augmented state sequence bigram—$1000_2, 4546_1$—where 4546_1 denotes a single run of length one for state 4546. The extended feature space is depicted in Table 3.4, where Φ_w, Φ_s and Φ_d represent the lexical, the acoustic state and their duration n-gram subsequences or features, respectively. In the sequence from the example, the bigram feature $1000_2, 4546_1$ is assigned a value of two since the bigram occurs twice. Thus, in the bigram case, the duration feature space spans the product of the bigram acoustic state space, augmented with every possible observed run length.

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

Table 3.4: Features corresponding to the word, acoustic state and their duration sequences associated with a given candidate.

3.3.2 Results

The experimental setup and data are the same as in Section 3.2, but in this case we also incorporate duration features into our discriminative model.

System	$\#F_U$	Xval	$\#F_C$	Dev07	$\#F_C$	Eval07
Baseline	-	26.6	-	21.2	-	24.0
Φ_{w2}	1.13M	26.2	24.74M	20.9	28.34M	23.8
Φ_{w2}, Φ_{s2}	1.47M	25.7	492.59M	20.6	570.91M	23.4
$\Phi_{w2}, \Phi_{s2}, \Phi_{d2}$	3.15M	25.2	949.74M	19.6	1.10G	22.6

Table 3.5: Comparison of discriminative models with language, acoustic and duration components, measured (WER) on 20-fold cross-validation, Dev07 and Eval07.

As reported in Table 3.5 when incorporating duration features into the joint discriminative language and acoustic model the WER improves between 0.5%-1.0% at the cost of doubling the size of the model. In this case, around 76.0% and 80.0% of feature types (language, acoustic state and duration feature types) from Dev07 and Eval07 are active in the model, respectively, with a coverage of almost 98%.

3.4 Analysis of the proposed joint discriminative models

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 $[\Phi_{w2}]$ $[\Phi_{w2}\ \Phi_{s2}]$, and $[\Phi_{w2}\ \Phi_{s2}\ \Phi_{d2}]$.

3.4.1 Effect of the number of competing candidates

The size of the lattice (i.e., the number of competing candidates being considered in the training data and rescored during testing) impacts both the accuracy and the computational cost. The number of competing candidates in the training and test set can be varied independently and we explored three settings, represented in Figure 3.4, as previously done by Hollingshead when reranking n-best lists from parsers [56]. The legend in the graph shows pairs, (N_{train}, N_{test}), where the first number refers to the size of candidates in the training set and the second to that in the test set. The number of candidates in training affects computational cost disproportionately more than the number of candidates in testing, since the perceptron algorithm has to iterate over all the candidates. As a compromise between the settings (100,100) and (1K, 1K), we also investigated (100, 1K) setting.

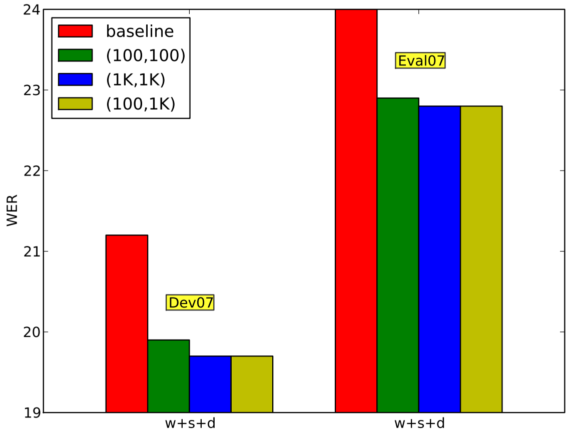


Figure 3.4: Impact of the model on Dev07 and Eval07 performance (WER) when the number of competing candidates is changed. Legend represents (N_{train}, N_{test}) as explained in the text.

As the number of candidates in training and testing is increased from 100 to 1K, 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.

3.4.2 Effect of the training data size

The amount of training data impacts the performance of discriminative models. In order to understand the trend, we measure the performance of the integrated models when they are trained with 200 hours, 100 hours and 50 hours of data.

The results, illustrated in Figure 3.5, show that the performance steadily improves as the size of the training data increases and does not appear to plateau for 200 hours, so we expect further gains with the complete 1000 hours of training data.

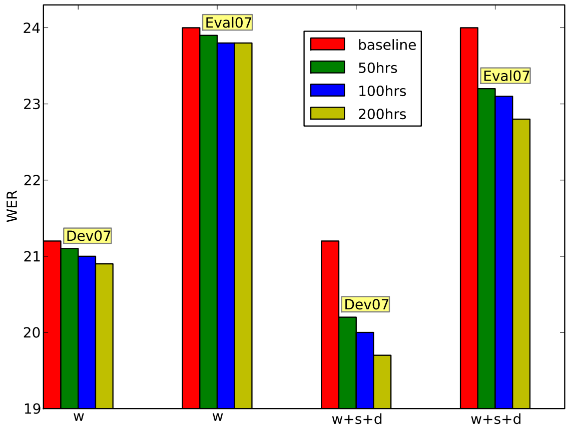


Figure 3.5: Impact of the training data size on Dev07 and Eval07 performance with Φ_{w2}, and $[\Phi_{w2}\ \Phi_{s2}\ \Phi_{d2}]$.

3.4.3 Effect of differences in genre

Recall that the discriminative models are 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 Figure 3.6, with lexical Φ_{w2} and acoustic states Φ_{s2} is higher in Broadcast News portion of Dev07 while the duration model provides similar gains across both genre.

Getting higher performance gains in BN portion may come from its speech data characteristics. The speech in this domain is more controlled, with less variability than conversational speech, and probably closer to the Modern Standard Arabic (MSA) dialect. The active feature types from the discriminative language model that are present in the BC portion from Dev07 make up 45.8% of the total feature types, while 36.5% of the feature types from the BN portion are active in the model. The vocabulary from the BC portion must be larger, and we may suffer from more sparsity issues than in the BN portion.

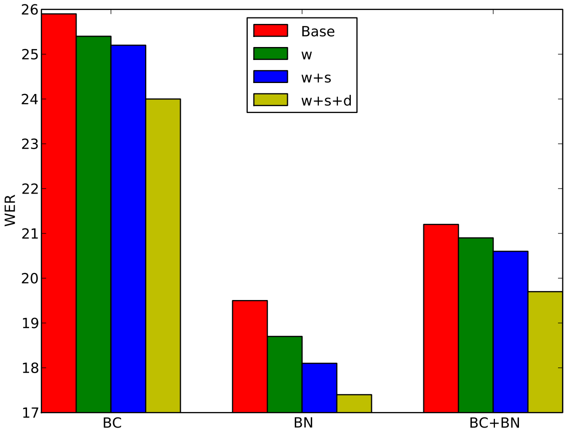


Figure 3.6: Performance on Broadcast Conversations (BC) and Broadcast News (BN) portion of Dev07 with models trained on BC.

3.4.4 Interaction between acoustic, language and duration components

Next, we investigate the interaction between acoustic, duration and language features for different orders of n-grams. The results, reported in Table 3.6, are grouped separately for different orders of lexical, acoustic, and duration n-grams. 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 corpus, whose scores are included in the likelihoods from the baseline recognizer. For each model, the interpolation weight (α_0) is 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 that the duration features explicitly encode the observed span for the acoustic states, and as a result, their bigrams are numerous. When acoustic and duration features are augmented with lexical bigrams, the performance improves consistently except for four cases out of 24 results. In this Arabic task, lexical features do not provide significant additional gains beyond those obtained for acoustic state and duration features. The most useful features

System	$\#F_U$	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

Table 3.6: Detailed performance (WER) comparison, measured on 20-fold cross-validation (Xval), Dev07 and Eval07, using models with different combinations of acoustic, duration and lexical n-grams features.

.

appear to be bigrams of durations of acoustic states. We observed the best performance when acoustic, duration and lexical features of the candidates are encoded as bigrams in the joint model. (Note that bigram discriminative models also contain unigram features.)

Effect of the interpolation weight (α_0): For all the experiments reported in this paper, we re-score 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:

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

The best performance on the cross-validation set is used as the optimal interpolation weight.

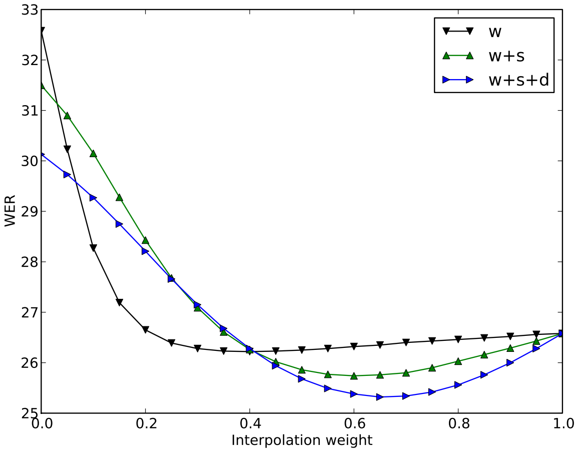


Figure 3.7: Effect of the interpolation weight α_0 on the performance of Xval for different combinations of features (Φ_{w2}, Φ_{s2} and Φ_{d2}).

As Figure 3.7 shows, the performance on the cross-validation set is robust. The performance is not sensitive to small changes in the interpolation weight and the local minima for joint models that include duration features are easy to locate.

3.5 Discussion

Regular large vocabulary speech recognition systems are built by rewriting the posterior probability $P(w|x)$ of the word sequence w given an acoustic input sequence x as a product of factors and by estimating a model for each of these factors. Then, using finite state transducer representation and operations, a compact and optimal decoding graph that combines all the components in a single WFST can be obtained. Dividing the problem into several components makes the training easier but at the cost of erroneously assuming independence between the features of the different components. In the case of speech recognition, it is assumed that acoustic and linguistic features are independent, despite the obvious interdependency that exists between the two.

Recently, a few research groups have pursued the idea of directly modeling the posterior probability $P(w|x)$ which is similar to our approach in that they jointly estimate the acoustic and language parameters [53, 203, 204, 205, 131, 206, 192, 82, 81].

Zweig and colleagues use CRF-based models which have the advantage of a convex cost function but their training is complex and computationally expensive. Our approach utilizes the structured perceptron algorithm, which does not need to compute any normalization factor and converges faster than CRFs [156].

Subsequent to our work, Kubo and colleagues [81] estimated the acoustic and linguistic parameters related to the arcs from the WFST-based decoding graph with perceptron and CRF-based training methods. Their CRF-based methods require an approximation to make the computation of the normalization factor feasible. Their acoustic features consist of an average of the frame-level vectors over the segment. This is a very crude characterization of the acoustic space. Our acoustic features in the form of acoustic states can take fewer values while modeling the acoustics using more complex GMMs. An alternative approach learns the means and variances of the GMMs along with word trigram features and optimizes an MCE criterion in a WFST-based decoding configuration [1]. They evaluated their approach on small tasks, TIMIT and RM1, and are yet to scale their approach to large vocabulary task.

Another recent related work focuses on estimating the parameters of the pronunciation model in a WFST-based framework [68]. Our approach also allows estimating the parameters from the pronunciation model jointly with other parameters from the speech recognizer as we demonstrate in the next chapter.

While our method can be applied to first-pass decoding, our experiments were performed on rescoring the output of a previous stage. Most of the observed gain in our joint model comes from the acoustic model; the contribution of the language model is about 0.2-0.4%. In the experiments from the next chapter our discriminative acoustic models also provide significant gains with more refined baseline acoustic models (with speaker adaptation and discriminatively trained GMMs).

The discriminative modeling framework from our work make it easy to incorporate other features that may not be fully exploited in general ASR systems such as duration of speech as demonstrated in Section 3.3. Previous attempts to model duration in ASR systems have only been modestly successful. We find that duration features in our joint model are effective and improve the performance significantly over the baseline system in a large vocabulary Arabic task.

Following our work, Kao and colleagues [71] included duration features into a segmental conditional random field (SCRF)-based speech recognizer to improve the discrimination between correct and incorrect word candidates. They modeled duration of the most frequent words when they were correctly and incorrectly recognized, and included that information in the constrained lattices that the speech recognizer outputs. The approach was evaluated on an English Broadcast News task yielding improvements of up to 0.3% over a state-of-the-art discriminatively trained baseline. They also estimated the parameters of duration features together with other acoustic and linguistic features. However, they modeled duration at word level while we do it at acoustic state level. Due to sparsity issues they considered only duration features of most frequent words which may be one of the reasons for their resulting modest gains. Previous papers modeling duration at the word level also reported limited gains when adding duration information [63, 35]. Arisoy and colleagues, instead, estimated duration features at the acoustic state level jointly with the discriminative language model [4] to improve the performance of an English Broadcast News task. They observed gains of about 0.3%, lower than we observed in our experiments. This may be due to the differences between English and Arabic transcription tasks. The data from our Arabic task contains BN and BC speech. The spoken Arabic is usually dialectal, and additionally some pronunciation phenomena are not written. These two aspects make the recognition of Arabic speech more ambiguous. Apart from this, we did not use discriminative acoustic models as baseline acoustic models.

Soltau and colleagues refined the Arabic unvowelized generative acoustic models by explicitly modeling the short vowels and other diacritics. The performance obtained was significantly better than the one with the unvowelized AMs, but these gains diminished with discriminative acoustic models [168]. This raises the question whether our joint model and duration features would be less effective with baseline acoustic models whose GMMs are trained discriminatively.

In the results from Soltau and colleagues [168] the vowelized systems appear to be more helpful for broadcast news data than for broadcast conversation data. In the experiments we ran to analyze the impact of the genre, the acoustic state features provide more gains for BN data even when the training set contains just BC data. However, when adding duration features we get similar gains for both genres, demonstrating the robust contribution of the duration features in the noisier BC data.

3.6 Summary

In this chapter I proposed a joint discriminative model to estimate parameters from the acoustic and language components as well as parameters from duration model. I described how this model generalizes the discriminative language model from Roark and colleagues [156]. 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 ($\mathcal{T}$) for speech recognition $\mathcal{E} \circ \mathcal{T} \circ \mathcal{D}$, and thus can be used in first pass recognition as well as for rescoring. Empirical results demonstrated that the proposed joint model improves performance by 1.4%-1.6% absolute on a GALE Arabic transcription task. The aim of the work from this chapter was to learn the weights on 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 VTLN baseline system.

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 hours to 100 hours and 200 hours 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 showed that most gains from the joint model can be obtained by training with only 100 competing candidates while testing with

1000 competing candidates. This is good news since the cost of training using an iterative procedure is more than that of testing this 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 a transition matrix of HMMs, estimated by maximum likelihood, has failed to provide any gains in large vocabulary recognition. The probabilities from the transition matrix are dwarfed by those from the high-dimensional observation components of the HMMs.

Chapter 4

Discriminative pronunciation modeling for dialectal speech recognition

4.1 Introduction

Speech recognition technology is increasingly ubiquitous in everyday life, but its adoption depends on the user's level of satisfaction with the performance of the system. Differences between individual speakers pose one of the main challenges in speech recognition [11]. Speech varies with age, gender, ethnicity, geography (in the form of regional dialects), and socioeconomic status [89, 90]. In particular, the performance of the recognition is very sensitive to the dialect or accent of the speakers, especially when the given dialect or accent is not covered in the training data [191, 202]. As it is not feasible to develop new speech recognition systems for every accent and dialect, the approach commonly adopted is to adapt an existing recognizer trained with general data from the main language, for example, standard American English (SAE), into the specific accent or dialect such as non-native English or African American Vernacular English (AAVE). Previous work has focused on applying the generative acoustic model, the pronunciation lexicon and the polyphone decision tree adaptation, especially, for non-native or accented speech. A review of the different adaptation approaches proposed in previous work is detailed in the final report from the workshop on dialectal Chinese speech recognition [170]. In this chapter, instead, we use the discriminative approach presented in the previous chapter (Chapter 3.1) to learn the weights of a phonetic transformation model (pronunciation variation model) that captures the phonological differences between the task dialect and the canonical pronunciations. In our case, the phonetic transformations are generated from knowledge-based phonological rules. We evaluate our approach on adapting an ASR system trained with SAE speech to recognize AAVE speech. As pronunciation is also modeled with the acoustic models by representing acoustics at context dependent phonetic level, we also explore adapting pronunciation at acoustic level by training a discriminative acoustic state model using features at acoustic state level. Additionally, we jointly train parameters from the pronunciation and language model. The intuition behind this is that, first, the language model may be able to capture pronunciation mismatches not covered by the pronunciation-related features, and second, the model may just need to include the phonetic transformations that correct the errors not considered by the language model. The experiments are also run with SAE speech from the same domain of AAVE speech to tease apart the impact of dialect and domain difference.

The common techniques applied in the literature for generative acoustic model adaptation are the maximum likelihood linear regression (MLLR) [95] and maximum a posteriori (MAP) adaptation [39]. Wang and colleagues [191] investigate the speech recognition for German-accented English speakers by comparing the performance using MLLR adaptation vs. MAP adaptation. They also give results for cases when acoustic models are trained only with the accented speech, with data containing native and accented speech, and by interpolating the native and accented acoustic models. Tomokiyo and Waibel [178] examine Japanese-accented English speakers by applying MLLR and MAP adaptation separately. The combination of both adaptation techniques in the speech recognition of Shanghai-accented Mandarin is analyzed by Zheng and colleagues [202]. Vergyri and colleagues [187] investigate the effect of accent variation in English Broadcast News domain with data from various geographical regions by training independent models for each accent vs. training a single model with data from all accents and then, applying MAP adaptation. As an alternative to these common adaptation techniques, Motlicek and colleagues employ subspace Gaussian mixture models [129], while Soltau and colleagues [166] use a dialect classifier for separating Modern Standard Arabic (MSA) speech utterances from Levantine Arabic ones, and then, the acoustic models are trained with all the data containing MSA and Levantine Arabic dialect speech, but weighting the statistics computed from the utterances with the score from the binary dialect classifier. Other authors [191, 130] propose modifying the decision trees used to cluster the context-dependent phone models based on questions about the phonetic context. The effectiveness of the above methods has been limited. This may be in part due to limitations of the above adaptation techniques. They largely impact the observation model of the states, and do not model the systematic phonemic variations that occur in dialectal or accented speech.

There is extensive literature that explores ways to capture these systematic phonemic variations by adapting the pronunciation model at the lexical level [58, 44, 179, 104, 188, 59]. Pronunciation variants may be derived from knowledge-based or data-driven techniques, and filtered or weighted so as not to increase the confusability during the search [172]. Rather than adding pronunciation variants directly into the lexicon, Tomokiyo [179] and Hutchinson and Droppo [59] use them for re-scoring lattices obtained in the first pass decoding.

A popular approach to capture new pronunciation variants is to build joint-multigram models to map graphemes onto phones [16]. They are built as regular word n-gram language models (LMs), but the tokens are grapheme/ phone pairs—graphone units—instead of words. These models consider information about grapheme and phone context. McGraw and colleagues [112] use a graphone model to derive alternative pronunciations for the words from the training data. After running Expectation-Maximization (EM) for several iterations, they learn pronunciation weights, and filter out any pronunciation variant with a probability lower than a certain threshold. Lu and colleagues [106] use a similar method. However, in their experiments, the graphone model is learned with a subset of the expert pronunciation lexicon instead of the whole set, and the EM algorithm is run over the lattices rather than over the N-best alternative pronunciation sequences.

Alternatively, a phonological rule model can be represented with the joint multigram model. In this case the model maps the spoken (surface) phone sequence onto a canonical phone sequence. A unigram model with tokens representing surface and canonical phone pairs is used by Karanasou and colleagues [72]. The weights of the model are learned discriminatively, using conditional random fields (CRFs) and the perceptron algorithm to minimize phone edit distance. They report improvements in phone error rate, but not in word error rate. In the work we present in this chapter we also learn a pronunciation variation model that maps surface phones into canonical phones with the perceptron algorithm, but with several differences. First, our surface phone sequences are obtained after decoding with the canonical pronunciation lexicon, while in their case the surface phone sequences come from a phone decoder. Second, we use unigram and bigrams of phone transformations as opposed to just unigram phone transformations. Finally, instead of optimizing the phone error rate, we learn the parameters by optimizing the word error rate.

Vinyals and colleagues [188] also use a discriminative criterion for pronunciation learning. They first employ a minimum classification error approach to select the pronunciation alternatives to include into the pronunciation model. Then, the principle of maximum entropy is used to learn weights for pronunciation candidates. This approach is limited by the amount of variants they are able to take into account; many test words simply do not appear frequently in the training data.

Under a weighted finite state transducer (WFST) framework, Jyothi and colleagues [68] discriminatively weight the arcs from the lexicon to optimize the recognition on Switchboard conversational speech data [42]. They report results when they learn parameters of the arcs only from the pronunciation lexicon WFST, and also when the parameters from the pronunciation lexicon are jointly estimated with the parameters related to the arcs from the acoustic HMM and decision tree WFSTs. The features to train the discriminative models are extracted at the level of individual acoustic frames to take phone duration into account. The authors find significant improvements in the recognition of isolated words, but not recognizing continuous speech.

Our goal is not only to discriminatively weight existing canonical pronunciations at phonetic level, but also to provide an approach to learn new pronunciation variants that are closer to the way words are pronounced in the task dialect. Additionally, we show that our framework also allows us to learn parameters related to pronunciation features at acoustic level.

4.2 African American Vernacular English

African American Vernacular English (AAVE) is a dialect of English spoken by many African Americans, especially those living in predominantly black communities when speaking in a casual register [153]. AAVE is distinct from standard American English (SAE) in grammar, pronunciation, and vocabulary. Table 4.1, based on Rickford (1999), lists a number of "variable rules" of AAVE. These relate canonical SAE pronunciations to common realizations in AAVE. These rules apply stochastically rather than categorically, and the probability of application is sensitive to phonological context as well as speaker-specific social factors such as socioeconomic status [88, 194]. Some of these variable rules are also found in other English dialects.

Phonological rule	Example	Phonological rule	Example
th → t	*thin*	[+cons] → ∅ / [+cons] __ |	*hand*
dh → d	*this*	s p → p s / __ |	*grasp*
th → f	*bath*	s t → t s / __ |	*pianist*
dh → v	*with*	s k → k s / __ |	*ask*
oy → ao	*boy*	ax → ∅ / | __	*about*
ay → aa	*right*	ih → ae / __ ng	*drink*
v → b / __ [+nasal]	*movement*	eh → ih / __ [+nasal] {|, [+cons]}	*pin*
l → ax / [+vowel] __	*bell*	l → ∅ / [+vowel] __ [+labial]	*help*
z → d / __ [+nasal]	*isn't*	z → d / [+vowel] __ [+vowel]	*reason*
r → ax / [+vowel] __	*four*	v → b / [+vowel] __ [+vowel]	*having*
r → ∅ / [+vowel] __	*there*	d → t / [+vowel] __ |	*god*
ng → n / ih __ |	*walking*	b → p / [+vowel] __ |	*cab*
r → ∅ / | th __	*throw*	t → k / s __ r	*destroy*
ts → ih z / s __ |	*breasts*	g → k / [+vowel] __ |	*log*
ts → ∅ / __ |	*ghosts*		

Table 4.1: Variable rules mapping from SAE pronunciations to possible pronunciations in AAVE.

4.2.1 Discriminative pronunciation model

The performance of an ASR system often degrades considerably when it is employed to recognize certain dialects that are not well represented in the training data. We assume that pronunciation variation is the primary cause of this degradation. The pronunciation lexicon is traditionally a deterministic mapping from words into their canonical phonemic representation. There may be a few pronunciation variants for some words (e.g., *and*), but these are unweighted. The lexicon is not learned or adapted for each task, so pronunciation variation is largely captured by the AMs [67]. However, it is not often feasible to train new AMs for other dialects, so baseline AMs, trained on publicly available corpora, must be adapted for recognition of those other dialects. Furthermore, when dealing with speech from a dialect other than the standard, the pronunciation at the lexical level differs more from the canonical pronunciation of the lexicon. Hence, the lexical-level pronunciation must be adapted too.

The adaptation may be performed by adding new pronunciation variants into the lexicon, or by learning a pronunciation variation model that maps surface phones onto canonical phones. The pronunciation variation model may be built from knowledge-based rules or rules learned from the data. In our work we use knowledge-based phonological rules, depicted in Table 4.1, to generate the features for the discriminative pronunciation variation model. Specifically, we discriminatively learn the weights for n-grams of phone transformations of the form $p_s : p_c$ that match any rule from Table 4.1, where p_s is the surface phone and p_c the canonical phone. The goal is to introduce pronunciation variability with the phone transformation pairs to model dialectal speech without increasing the confusability or ambiguity during decoding.

We apply the idea of the global linear models explained in Section 3.1.1 for the estimation of the discriminative models. The goal of the decoding is to estimate the word sequence w for a given input speech utterance x. For that, the speech utterances x are decoded to get the ASR outputs y. A function **GEN** enumerates a set of candidates $\mathbf{GEN}(x)$ for the input x (for example, N-best candidates or lattices). In our case, each candidate y contains not only the word sequence w, but also the sequence with phonetic information, where each token is a phone transformation $p_s : p_c$. In the proposed discriminative pronunciation model, the representation Φ maps each (x, y) to a feature vector $\Phi_p(x, y) \in \Re^d$ and a parameter vector $\bar{\alpha}_p \in \Re^d$ (features and parameters related to the pronunciation model are denoted with the subscript p), and the output of a linear model $F(x)$ is computed as below.

$$F(x) = \underset{y \,\in\, \mathbf{GEN}(x)}{\operatorname{argmin}} \; \Phi_p(x,y)\bar{\alpha}_p \tag{4.1}$$

The learned pronunciation variation model could easily be represented as a WFST $\mathcal{P}$ and included into the WFST-based speech recognizer $\mathcal{T}$ as follows:

$$\mathcal{T}' = \mathbf{Min}(\mathbf{Det}(\mathcal{H} \circ \mathcal{C} \circ \mathcal{P} \circ \mathcal{L} \circ \mathcal{G})) \tag{4.2}$$

Alternatively, the WFST $\mathcal{P}$ could also be applied in a post-processing step to re-score the paths from the ASR lattices L_x. For our experiments, the pronunciation variation model is applied in the re-scoring of N-best lists extracted from the speech recognizer output.

4.2.2 Discriminative pronunciation model at the level of acoustic states

As an alternative to explicitly modeling the pronunciation variation at symbolic level we can implicitly model at the acoustic state level [161, 49, 70]. In this case the parameters $\bar{\alpha}_s$ to estimate with our discriminative models are the weights of the features $\Phi_s(x,y)$ representing acoustic state n-grams. They are the same kind of features as the features used in the experiments from Chapter 3. The output of the linear model $F(x)$ is computed as:

$$F(x) = \underset{y \,\in\, \mathbf{GEN}(x)}{\operatorname{argmin}} \; \Phi_s(x,y)\bar{\alpha}_s \tag{4.3}$$

If we wanted to incorporate the discriminative model into the WFST-based decoding graph, these parameters could be compiled in a WFST $\mathcal{E}$ and composed with the WFST-based decoding graph. Alternatively, it could be applied in a post-processing stage as mentioned in Section 3.1.2. In our experiments the discriminative models are applied to re-score the N-best lists extracted from the lattices $\mathcal{L}_x$.

4.3 Experimental results

4.3.1 Corpus

We use data from the StoryCorps project[1] for all the experiments, a subset of the data set used by Chen and colleagues for their dialect recognition experiments [20]. This data consists of conversations between two or more speakers of the same dialect, either AAVE or SAE. Speaker dialect is self-reported. We learn independent discriminative models for each dialect. For AAVE, the training and test set contain 56 speakers and 11 speakers, respectively. The training and test sets for SAE include 46 speakers and 14 speakers, respectively. For both dialects, the training set consists of approximately 13 hours of speech, and the test set approximately 3 hours. Both the training and test sets have equal numbers of male and female speakers.

4.3.2 Baseline system

The baseline ASR system is a recognizer designed for English Broadcast News modeled with the IBM toolkit following Soltau and colleagues [74]. The acoustic model consists of 4000 clustered allophone or acoustic states defined over a pentaphone context and a set of 44 phones, with states represented by Gaussian mixture models with a total of 150K mixture components. The observation vectors consist of PLP features, stacked from 10 neighboring frames and projected into a 50-dimension space using linear discriminant analysis [47]. The acoustic models were trained using 430 hours of transcribed English Broadcast News speech [45, 34]. The language model was estimated using several corpora of conversational telephone speech; it has a 48k-word vocabulary and contains approximately 16M bigrams, 16M trigrams, and 11M 4-grams. On average, the pronunciation lexicon has 1.1 pronunciations per word. Decoding is performed in several stages using successively refined acoustic models, including a context-dependent model, a vocal-tract normalized model, a speaker-adapted maximum likelihood linear regression model, and a feature and model-based discriminative model explained in more detail in Section 2.4.

[1] `www.npr.org/series/4516989/storycorps`

4.3.3 Generation of competing candidates

For training the discriminative pronunciation models—the AAVE and SAE models—each utterance x is decoded with the baseline recognizer, generating lattices L_x containing word sequences, acoustic state sequences, their time indices, and associated log probabilities. The 100-best unique candidates are then extracted from each lattice. For each candidate y, apart from the word sequences, we also extract the single best acoustic state sequence s with the associated durations, and the canonical phone sequence p_c. The canonical phone sequence is obtained by forced-aligning the speech utterance with the word sequence candidate using the baseline (out-of-dialect) ASR system with the canonical pronunciation lexicon.

Additionally, we also generate a second phone sequence for each candidate using the knowledge-based phonological rules from Table 4.1. The second phone sequence contains surface phones and is generated by forced-aligning the speech utterance with the word sequence candidate using an extended pronunciation lexicon. The extended pronunciation lexicon contains canonical pronunciations and new additional pronunciations generated after applying the knowledge-based phonological rules to the canonical pronunciations. Table 4.2 shows the extended lexicon generated for a candidate with the word sequence "*and then*". In this example, a new pronunciation is generated for the word *and* after applying the rule [+cons] → $\emptyset$. The word *then* gets assigned two new pronunciations from the rule dh → d and dh → v.

and(01)	ae	n	d
and(02)	ae	n	
then(01)	dh	eh	n
then(02)	d	eh	n
then(03)	v	eh	n

Table 4.2: Extended lexicon for candidate "*and then*".

After forced-aligning the word sequence candidate with the extended lexicon the phone sequence associated with the lowest acoustic cost (highest likelihood) may be different from the canonical phone sequence. We call this new phone sequence the surface phone sequence p_s. Then, the Levenshtein alignment [100] is computed for the canonical phone p_c and surface phone sequence p_s. The alignment produces a sequence whose tokens are pairs of phones of the form $p_{i,s} : p_{i,c}$, where i represents the position in the sequence.

4.3.4 Parameter estimation

The parameters of the models are iteratively estimated with perceptron algorithm, using the oracle candidates as the reference. The parameter estimation is iterated until we see no improvement on the held-out data for five iterations. Maximum likelihood scores, L_{ML}, are interpolated with scores from the discriminative models, L_{DM}, using an interpolation weight α_0 determined by 20-fold cross validation.

$$L = \alpha_0 * L_{ML} + (1 - \alpha_0) * L_{DM} \tag{4.4}$$

4.3.5 Features Space

For each candidate in the N-best list we have the following sequences:

- Word sequence: <s> and then </s>
- Canonical phone sequence: <s> | ae n d | dh eh n| </s>
- Surface phone sequence <s> | ae n | d eh n | </s>
- Joint phone sequence: <s> | ae:ae n:n -:d | d:dh eh:eh n:n | </s>
- Acoustic state sequence: <s> 3219, 3219, 3219,3179,2900... </s>

From these sequences we extract features in the form of n-grams to train the pronunciation variation model (phone transformation n-grams), the discriminative acoustic state model (n-grams of acoustic states) and the joint pronunciation model (n-grams of pronunciation cues and word n-grams). Table 4.3 gathers some of the n-grams extracted from the candidate example.

$\Phi_{p_s:p_c}$	Φ_s	Φ_w
ae:ae,n:n=1	3219,3179=1	and,then=1
n:n,-:d=1	3179,2900=1	then, </s>
...	...	...

Table 4.3: Features corresponding to the phone transformations, acoustic state sequences and words associated with the candidate example.

4.3.6 Results

Discriminative pronunciation model

We compare our models on StoryCorps corpus, using both the AAVE and SAE portions described in Section 4.3.1 and report word error rate on 20-fold cross-validation (Xval) and test set. As explained earlier, the phonetic pronunciation-related features are unigrams and bigrams of phone transformations; $\Phi_{p_s:p_c}$. The experiments on SAE and AAVE are performed independently to understand the effectiveness of the model on standard vs. AAVE dialectal speech. Statistical significance is computed with respect to the baseline system using the matched-pairs test [41] provided by NIST SCTK toolkit [132] and results non-significant at $p < 0.001$ are marked with †.

A large number of previous studies were performed with maximum likelihood (ML) acoustic models. So, we report results with both our best generative and discriminative models, specifically the speaker-adapted models (FSA) and boosted MMI models (BMMI), respectively. The results from the FSA and BMMI models are reported in Table 4.4 and Table 4.5, respectively.

The performance of the baseline speech recognizer on AAVE data is over 10% worse than on SAE data. This is expected since the SAE data is closer to the baseline acoustic models (English Broadcast News). For reference, the oracle accuracy for the 100-best candidates is reported which specifies the lowest achievable WER for all the re-scoring experiments from the tables. There is considerable room for improvement in both SAE and AAVE dialects. When the baseline AMs are discriminatively trained (BMMI models) the WER improves on both dialects with respect to the results with generative acoustic models (FSA models). Although the discriminative training of the baseline acoustic models

	AAVE					SAE				
	$\#F_U$	α_0	Xval	$\#F_C$	test	$\#F_U$	α_0	Xval	$\#F_C$	test
Oracle	-	-	30.3	-	32.1	-	-	17.4	-	19.6
Baseline	-	-	40.4	-	42.8	-	-	25.8	-	29.3
AM rescoring	-	-	40.2	-	42.3	-	-	26.1	-	29.7†
Φ_{p_c}	1.42K	0.40	39.1	15.98M	41.6	1.43K	0.35	25.0	14.92M	28.5
$\Phi_{p_s:p_c}$	1.53K	0.35	39.1	21.13M	41.3	1.47K	0.35	25.5	19.58M	28.8

Table 4.4: Performance (WER) comparison of discriminative pronunciation model on SAE and AAVE portions of the corpus using features from canonical phone sequence (Φ_{p_c}) and sequence with phone transformations ($\Phi_{p_s:p_c}$) with **FSA models**. F_u and F_c denote the number of unique features in the model and the number of times they were employed in the test sets, respectively. Results non-significant at $p < 0.001$ are marked with †.

	AAVE					SAE				
	$\#F_U$	α_0	Xval	$\#F_C$	test	$\#F_U$	α_0	Xval	$\#F_C$	test
Oracle	-	-	26.7	-	28.1	-	-	14.8	-	16.6
Baseline	-	-	36.1	-	38.2	-	-	22.3	-	25.2
ML rescoring	-	-	35.9	-	37.7	-	-	22.4†	-	25.3†
Φ_{p_c}	1.0K	0.35	35.1	20.68M	37.0	0.93K	0.45	21.8	18.45M	25.0†
$\Phi_{p_s:p_c}$	1.48K	0.35	35.1	20.67M	36.9	1.40K	0.45	21.9	18.33M	25.1†

Table 4.5: Performance (WER) comparison of discriminative pronunciation model on SAE and AAVE portions of the corpus using features from canonical phone sequence (Φ_{p_c}) and sequence with phone transformations ($\Phi_{p_s:p_c}$) with **BMMI models**. F_u and F_c denote the number of unique features in the model and the number of times they were employed in the test sets, respectively. Results non-significant at $p < 0.001$ are marked with †.

is done with out-of-dialect data the recognition of AAVE data also improves, since they share common phones.

Next, we re-score the N-best candidates with the baseline acoustic models and the extended lexicon that incorporates the phonological rules from Table 4.1. The performance of this maximum likelihood re-scoring pass is denoted as “ML rescoring” in Table 4.4 and Table 4.5. The WER improves on the AAVE portion but not on the SAE portion. This may be an indicator of the utility and relevance of the phonological rules for AAVE data. In the case of the baseline FSA acoustic models (less accurate models) the ML re-scoring has more negative impact on the SAE portion than when using baseline BMMI models (more accurate models).

	AAVE					SAE				
	$\#F_U$	α_0	Xval	$\#F_C$	test	$\#F_U$	α_0	Xval	$\#F_C$	test
Oracle	-	-	30.3	-	32.1	-	-	17.4	-	19.6
Baseline	-	-	40.4	-	42.8	-	-	25.8	-	29.3
$\Phi_{p_s:p_c}$	1.53K	0.35	39.1	21.13M	41.3	1.47K	0.35	25.5	19.58M	28.8
Φ_{p_c}	1.42K	0.40	39.1	15.98M	41.6	1.43K	0.35	25.0	14.92M	28.5
Φ_s	22.33K	0.65	38.8	47.38M	40.9	21.98K	0.85	25.1	44.15M	28.5

Table 4.6: Performance (WER) comparison of discriminative pronunciation model on SAE and AAVE portions of the corpus using features from the acoustic state sequence (Φ_s) with **FSA models**. F_u and F_c denote the number of unique features in the model and the number of times they were employed in the test sets, respectively.

Instead of ML re-scoring, we use canonical phones Φ_c as features in a discriminative model. The performance of this model is marginally better than the ML rescoring for both SAE and AAVE. Increasing the complexity of the model, we investigate the utility of features that encode the transformed surface form and canonical form $\Phi_{p_s:p_c}$. The additional complexity does not appear to be particularly useful in our case. Adding new pronunciation variants does not bring further gains. We may need a larger set of phone transformations to see further improvements.

Discriminative pronunciation model at the level of acoustic states

Previous work has modeled pronunciation both explicitly at phonetic level and implicitly in the acoustic models [161, 49]. They found that modeling pronunciation with the acoustic models can provide similar or even better performance than adding new pronunciation variants at phonetic level, without the need to worry about increasing the decoding confusability. Instead of using phonetic features, we learn discriminative models with features from the acoustic models to determine if they are also able to capture AAVE-specific pronunciations. The discriminative acoustic state model is represented using unigram and bigrams of acoustic state as features: Φ_s, as in Chapter 3 (Section 3.2.7). We do not reestimate parameters from the Gaussian mixture models, just transition weights between acoustic states. Results are detailed in Table 4.6 and Table 4.7 for baseline generative speaker adapted (FSA) and discriminative (BMMI) acoustic models, respectively.

Discriminative pronunciation models with features from the acoustic models behave

	AAVE					SAE				
	$\#F_U$	α_0	Xval	$\#F_C$	test	$\#F_U$	α_0	Xval	$\#F_C$	test
Oracle	-	-	26.7	-	28.1	-	-	14.8	-	16.6
Baseline	-	-	36.1	-	38.2	-	-	22.3	-	25.2
$\Phi_{p_s:p_c}$	1.48K	0.35	35.1	20.67M	36.9	1.40K	0.45	21.9	18.33M	25.1†
Φ_{p_c}	1.0K	0.35	35.1	20.68M	37.0	0.93K	0.45	21.8	18.45M	25.0†
Φ_s	22.03K	0.60	34.6	46.34M	36.7	21.98K	0.75	21.6	41.56M	24.6

Table 4.7: Performance (WER) comparison of discriminative pronunciation model on SAE and AAVE portions of the corpus using features from the acoustic state sequence (Φ_s) with **BMMI models**. F_u and F_c denote the number of unique features in the model and the number of times they were employed in the test sets, respectively. Results non-significant at $p < 0.001$ are marked with †.

slightly better than the models with phonetic features. They provide gains of 1.5% or more on the AAVE dialect, but no more than 0.8% on the SAE dialect. However, the feature space of discriminative models with acoustic features is much larger than the feature space of the models with phonetic features, about 15 times larger. The gains for the baseline discriminative (BMMI) acoustic models are lower for both pronunciation models (with acoustic states and phones), but not significantly. Training models that combine acoustic and phonetic features do not provide additional gains. It may be that their contributions are not additive or that the large feature space of the acoustic features hides the contribution of the phonetic features.

Joint discriminative pronunciation and language model

As the proposed estimation framework allows us to easily learn parameters from different ASR components jointly, we next analyze the interactions between the language and pronunciation models. The language model is represented using unigram and bigrams of words as features: Φ_w, as in Chapter 3 (Section 3.2.7).

As results from Table 4.8 and Table 4.9 show, discriminative language models (DLMs) are more useful for the AAVE data than for the SAE data. They provide gains over 1% for the AAVE test set, but lower than 0.7% for the SAE test set.

Including language features along with acoustic state features provides consistent additional gains in both dialects, although these gains are not always statistically significant.

	AAVE					SAE				
	$\#F_U$	α_0	Xval	$\#F_C$	test	$\#F_U$	α_0	Xval	$\#F_C$	test
Baseline	-	-	40.4	-	42.8	-	-	25.8	-	29.3
Φ_w	34.04K	0.25	39.2	4.02M	41.7	46.80K	0.25	25.1	3.79M	28.7
Φ_s	22.33K	0.65	38.8	47.38M	40.9	21.98K	0.85	25.1	44.15M	28.5
Φ_w, Φ_s	90.22k	0.65	38.7	51.73M	40.7	63.3K	0.65	24.7	47.87M	28.2
$\Phi_{p_s:p_c}$	1.53K	0.35	39.1	21.13M	41.3	1.47K	0.35	25.5	19.58M	28.8
$\Phi_w, \Phi_{p_s:p_c}$	47.86K	0.45	38.7	25.28M	41.0	53.53K	0.50	25.1	23.41M	28.4
Φ_{p_c}	1.42K	0.40	39.1	15.98M	41.6	1.43K	0.35	25.0	14.92M	28.5
Φ_w, Φ_{p_c}	54.65K	0.45	38.8	25.49M	41.1	67.93K	0.45	24.9	23.51M	28.4

Table 4.8: Comparison of performance (WER) of joint discriminative models on SAE and AAVE portions with features from different levels—words (Φ_w), canonical phones (Φ_{p_c}), surface and canonical phone transformations ($\Phi_{p_s:p_c}$) and acoustic states with **FSA models**. F_U and F_C denote the number of unique features in the model and the number of times they were employed in the test sets, respectively.

	AAVE					SAE				
	$\#F_U$	α_0	Xval	$\#F_C$	test	$\#F_U$	α_0	Xval	$\#F_C$	test
Baseline	-	-	36.1	-	38.2	-	-	22.3	-	25.2
Φ_w	61.03K	0.25	34.9	4.06M	36.7	48.48K	0.25	21.8	6.94M	24.7
Φ_s	22.03K	0.60	34.6	46.34M	36.7	21.98K	0.75	21.6	41.56M	24.6
Φ_w, Φ_s	78.35K	0.65	34.5	50.38M	36.3	84.45K	0.80	21.5	45.21M	24.4
$\Phi_{p_s:p_c}$	1.48K	0.35	35.1	20.67M	36.9	1.40K	0.45	21.9	18.33M	25.1†
$\Phi_w, \Phi_{p_s:p_c}$	67.12K	0.45	34.6	24.75M	36.1	59.93K	0.55	21.7	21.91M	24.5
Φ_{p_c}	1.0K	0.35	35.1	20.68M	37.0	0.93K	0.45	21.8	18.45M	25.0†
Φ_w, Φ_{p_c}	76.75K	0.40	34.6	24.67M	36.2	60.52K	0.45	21.5	22.04M	24.4

Table 4.9: Comparison of performance (WER) of joint discriminative models on SAE and AAVE portions with features from different levels—words (Φ_w), canonical phones (Φ_{p_c}), surface and canonical phone transformations ($\Phi_{p_s:p_c}$) and acoustic states with **BMMI models**. F_U and F_C denote the number of unique features in the model and the number of times they were employed in the test sets, respectively. Results non-significant at $p < 0.001$ are marked with †.

Including phonetic features into the discriminative language model, either with, Φ_{p_c} or $\Phi_{p_s:p_c}$ provides further consistent gains, especially for the AAVE data. Combining the discriminative language model with pronunciation features at the level of acoustic states or at the phonetic level provides comparable results for the AAVE data, but twice as many features are activated when using the acoustic states as when using the phonetic features.

A discriminative model that jointly estimates the parameters of the language model and the pronunciation model improves the performance of the AAVE recognizer by about 2.1% WER of which about 0.5% can be attributed to pronunciation models. On the other hand, improvements on the SAE data are lower and mainly attributed to the discriminative language model, since the canonical pronunciations are well-matched to the SAE and the training data used in the baseline (out-of-dialect) recognizer.

Even if the data sets are quite small and the observed gains are quite limited we can state that the discriminative models seem to be more effective for the AAVE dialect than for the SAE dialect. The lower gains on SAE data are expected since the models are closer to the baseline speech recognizer. Furthermore, the phonological rules make sense for the AAVE dialect, but not for the SAE dialect, as expected. However, the phone transformations we use to generate the features and capture phonological characteristics of AAVE do not provide benefits over using only features extracted from the canonical phone sequences. We may see benefits from using phone transformations if we use a larger set of rules. Modeling pronunciation at the acoustic state or phonetic level provides similar gains, but the features learned at the phonetic level are more intuitive and provide an easier analysis about which pronunciation variations are relevant to the given dialect. However, if our main goal is just to adapt the speech recognizer to the given dialect using acoustic state features is more straightforward. We do not need to compute the forced-alignments between the speech utterances and the word sequences.

Incorporating duration features

As in our experiments in the large vocabulary Arabic speech recognition task, described in Section 3.3, we incorporate duration information of acoustic states into our discriminative models, and explore the interaction between duration related features and the rest of the features. The baseline speech recognizer has acoustic models discriminatively trained with English Broadcast News data (BMMI models). Like the other features, the duration features, Φ_d, are also bigrams and unigrams. They are extracted and represented as explained in Section 3.3.1.

	AAVE					SAE				
	$\#F_U$	α_0	Xval	$\#F_C$	test	$\#F_U$	α_0	Xval	$\#F_C$	test
Baseline	-	-	36.1	-	38.2	-	-	22.3	-	25.2
Φ_d	224.0K	0.65	34.9	41.02M	36.8	181.37K	0.65	21.7	35.97M	24.7
Φ_w	61.03K	0.25	34.9	4.06M	36.7	48.48K	0.25	21.8	6.94M	24.7
Φ_w, Φ_d	288.80K	0.65	34.6	45.18M	36.5	223.57K	0.65	21.5	39.34M	24.5
Φ_s	22.03K	0.60	34.6	46.34M	36.7	21.98K	0.75	21.6	41.56M	24.6
Φ_s, Φ_d	277.18K	0.75	34.5	88.03M	36.3	221.61K	0.80	21.6	78.02M	24.2
Φ_w, Φ_s	78.35K	0.65	34.5	50.38M	36.3	84.45K	0.80	21.5	45.21M	24.4
Φ_w, Φ_s, Φ_d	340.96K	0.80	34.3	92.05M	36.2	225.30K	0.70	21.5	80.26M	24.4
$\Phi_{p_s:p_c}$	1.48K	0.35	35.1	20.67M	36.9	1.40K	0.45	21.9	18.33M	25.1
$\Phi_{p_s:p_c}, \Phi_d$	159.13K	0.60	34.6	59.70M	36.8	194.80K	0.70	21.7	54.47M	24.5
$\Phi_w, \Phi_{p_s:p_c}$	67.12K	0.45	34.6	24.75M	36.1	59.93K	0.55	21.7	21.91M	24.5
$\Phi_w, \Phi_{p_s:p_c}, \Phi_d$	289.16K	0.65	34.3	65.82M	36.4	202.02K	0.70	21.5	56.91M	24.5

Table 4.10: Comparison of performance (WER) of joint discriminative models on SAE and AAVE portions with duration features (Φ_d) combined with features from different levels—words (Φ_w), canonical phones (Φ_{p_c}), surface and canonical phone transformations ($\Phi_{p_s:p_c}$) and acoustic states with **BMMI models**. F_U and F_C denote the number of unique features in the model and the number of times they were employed in the test sets, respectively.

Discriminative duration models (Φ_d) provide gains comparable to those achieved using discriminative language models (Φ_w) for both dialects, but the feature space is almost four times larger. The joint models trained with both feature sets give consistent gains with respect to the independent discriminative duration or language models. The gains observed with AAVE data are approximately twice of what we observe with SAE data. Combining duration features with pronunciation features does not bring any significant improvement. Overall, duration features are useful for correcting errors from the baseline speech recognizer, especially when the pronunciation of the target speech (AAVE speech) significantly differs from the canonical pronunciation. However, the gains that duration features provide are not additive.

Discriminative pronunciation modeling with adapted acoustic models

Because it is common practice to use MLLR adaptation to adapt pronunciation of accented and dialectal speech [178, 191, 202], we next investigate whether our discriminative models

MLLR	N/A	w/ SAE	w/ AAVE
baseline	38.2	36.8	33.3
Φ_w	36.7	35.8	32.1
Φ_s	36.7	35.7	32.5
Φ_w, Φ_s	36.3	35.6	32.1
AM rescoring	37.7	36.5†	33.0†
$\Phi_{p_s:p_c}$	36.9	35.8	32.8†
$\Phi_w, \Phi_{p_s:p_c}$	36.1	35.6	32.0

Table 4.11: WER (%) on AAVE test set for joint discriminative models when the baseline BMMI acoustic models are adapted to AAVE (w/ AAVE) and SAE (w/ SAE) training sets using MLLR technique. Results non-significant at $p < 0.001$ are marked with †.

bring any additional gains when the underlying acoustic models are already adapted using standard techniques like MLLR. There are two cases to be considered: Adapting the baseline BMMI models (1) to the SAE training data and (2) to the AAVE training data. In the first case, we map the acoustic models from SAE Broadcast News domain to SAE StoryCorps domain. In the second case, the models are not only transformed from one domain to the other one, but also from SAE to AAVE. We compare these two cases with the baseline case (baseline BMMI models) and report results on AAVE test data in Table 4.11. As in the previous experiments we use a matched-pairs test [41] provided by NIST SCTK toolkit [132] for the statistical significance test. All the results are statistically significant with respect to the baseline (first row in Table 4.11) with $p < 0.001$ except for the cases marked with †.

When the baseline BMMI models are adapted to SAE data, the WER decreases by more than 1%. Both discriminative language and pronunciation models (with acoustic or phonetic features) provide an additional 1% improvement, slightly less than what we get with the non-adapted baseline AMs. The gains from the two models are not additive.

Interestingly, even after the BMMI models are adapted to AAVE data, we still find gains from pronunciation models. This confirms our earlier assertion that standard acoustic model adaptation (MLLR) is not sufficient to capture the phonetic transformations observed in AAVE dialect. Not surprisingly, the gains are less than in the unadapted case (first column from the table). In the adapted case, we found that the gains from the language model and the pronunciation models are not additive. This may be due to the overuse of the limited amount of AAVE training data for both MLLR and the training of our discriminative models.

4.4 Discussion

Dealing with dialectal or accented speech is one of the important challenges in speech recognition. Pronunciation mismatches between the speech recognizer and the speech from the target domain is one of the main causes of the performance degradation. Pronunciation lexicons of general speech recognizers usually contain more than one pronunciation for only a small number of words. The weak modeling of the pronunciation in the lexicon is overcome by the acoustic models that represent context dependent phones and are trained with many hours of speech data. This configuration may be acceptable for standard speaking styles, but when we need to recognize, for example, dialectal speech other than the standard, trying to capture the pronunciation variations that do not match the canonical pronunciations with the acoustic models might not be feasible, especially because we do not usually have large amounts of data for every specific dialect.

There are different approaches that people have proposed to refine the general pronunciation model from the speech recognizer, and also to adapt it to dialectal or accented speech. The technique we presented to adapt the pronunciation of an out-of-dialect speech recognizer to dialectal speech is related to recent publications [131, 204, 68]. Zweig, Nguyen and Heigold [131, 204] present a framework for discriminatively learning parameters not only from the pronunciation model but also from other components of the speech recognizer. They use flat direct models and segmental conditional random fields to decode a speech utterance into a word sequence. Apart from features from the language model they also consider pronunciation related features extracted from a decoded phonetic sequence that contains surface phones. They use more complex features than the ones we use in our work, but eventually we could also include more complex features in our models. Jyothi and colleagues [68] employ their approach to learn the parameters of the pronunciation model in terms of WFSTs. Specifically, they train the arcs from the pronunciation lexicon WFST that contains in average 1.7 pronunciations per word, in isolation, and also jointly with the weights for the arcs of the acoustic HMM and context-dependent decision tree WFSTs. As in our approach, they do not reestimate parameters of the GMMs from the acoustic models. They report gains for an isolated speech recognition task with Switchboard data, but not for the continuous case.

Both cited approaches [131, 204, 68] are tested on in-domain data, both the training data for the speech recognizer and the evaluation set contain data from the same domain. Their goal is to refine the pronunciation model that is poorly modeled in regular speech recognizers, so in that sense their work differs from our work.

The features we used for the pronunciation model are simple unigram and bigram features of phone transformations, but we may see some benefits from, for example, using triphone transformations, that is, considering phone transformations to the left and to the right of the current token. Furthermore, we generated the transformations by applying knowledge-based phonological rules that only consider intra-word phonological variations. An extension would be to also consider cross-word phonological rules, and using data-driven phone transformations instead of or in addition to phone transformations generated from knowledge-based rules.

4.5 Summary

In this chapter we applied discriminative linear models to adapt the pronunciation of an out-of-dialect speech recognizer into dialectal speech. We ran the experiments in parallel for two dialectal sets; AAVE and SAE dialects. Our hypothesis was that the style of the SAE dialect should be closer to the baseline speech recognizer. Therefore, the discriminative models learned with features representing phonological characteristics particular to AAVE dialect should provide more gains for the AAVE data set. The results obtained empirically seem to follow this hypothesis. As a preliminary step we re-scored the ASR N-best lists with the baseline acoustic models and the extended lexicon that incorporates the phonological rules. The re-scoring improved the WER on the AAVE set, but not on

SAE data which empirically validates the usefulness of the knowledge-based phonological rules to capture pronunciation cues that happen on AAVE dialect. The discriminative pronunciation models with features representing phonetic transformations were also more effective for improving the performance on AAVE data than for improving the performance on SAE data. The gains observed for AAVE dialect were, in general, approximately twice than those observed for SAE dialect.

Because of the flexibility of our discriminative models we were also able to explore modeling pronunciation at acoustic state level. Discriminative pronunciation models with acoustic state features were also more beneficial for AAVE dialect. Discriminative pronunciation models with acoustic states performed slightly better than with phonetic features. It may be that they are capturing more than just the dialectal pronunciation characteristics. Unlike pronunciation models with phonetic features models relying on acoustic states provided gains on SAE data.

Jointly estimating parameters from the language model and the pronunciation model provides further gains on AAVE portion, although these gains are not significant for pronunciation models with acoustic state features.

Combining acoustic state and phonetic features did not give additional gains over just using acoustic state features (we did not include these results in the tables). The feature space of the acoustic state features is much larger than the space of the phonetic features, and they are also frequently activated. Therefore, when using both feature spaces together the acoustic state features may conceal the contribution of the phonetic features.

We also explored the impact of duration features since duration is a relevant cue in pronunciation. Including duration features in the discriminative pronunciation models did not provide significant gains. Discriminative duration models and discriminative pronunciation models performed similarly, so duration features may also be capturing pronunciation characteristics of AAVE dialect, but the feature space of discriminative duration models was much larger than the feature space of the pronunciation models.

Finally, we studied different acoustic model configurations for the baseline speech recognizer: (1) English Broadcast News speaker-adapted models (generative speaker adapted models), (2) English Broadcast News BMMI models (discriminative models), (3) English

Broadcast News BMMI models adapted by MLLR with SAE data, and (4) English Broadcast News BMMI models adapted by MLLR with AAVE data. First, using generative vs. discriminative acoustic models in the baseline (out-of-dialect) speech recognizer did not impact the behavior of our discriminative models. The estimated discriminative linear models performed similarly for both baseline AMs. Second, when adapting the channel and acoustic environment mismatch between the English Broadcast News AMs and the StoryCorps data (English Broadcast News BMMI models with MLLR adaptation using SAE data) the overall WER improved, and the performance of our discriminative models was comparable to the performance with the non-adapted English Broadcast News BMMI model. Third, when the AMs were adapted with MLLR using AAVE dialectal speech, our discriminative pronunciation models still provided some gains.

Chapter 5

Joint discriminative modeling of lexical variation and acoustic confusion for automated narrative retelling assessment

5.1 Introduction

Narrative production tasks are part of many standard neuropsychological test batteries. They are intended to collect information related to lexical diversity, cohesion, coherence, content richness and syntactic complexity [180, 3, 73, 107], useful markers in diagnosing aphasia [43], autism [105], Alzheimer's [3] and other neurological conditions. For example, Western Aphasia Battery (WAB) is administered for detecting presence of aphasia [46], narration of a wordless picture book is part of the Autism Diagnostic Observation Schedule (ADOS) [105] and retelling of previously narrated stories is part of both the Developmental Neuropsychological Assessment (NEPSY) [77] and the Wechsler Logical Memory (WLM) test [193].

Such tests must be administered and evaluated by experienced clinicians. In the case of dementia, regular assessment of performance on these tests can lead to early detection of neurodegeneration. Building tools to automate this process could potentially allow clinicians to more easily and efficiently monitor their patients at risk of developing dementia.

In this chapter, I present and evaluate systems based on word tagging models to automate the scoring of narrative retellings. We evaluate our systems on the Wechsler Logical Memory test [193], a widely used subtest of the Wechsler Memory Scale neuropsychological test used to assess memory function in adults. The approach is general enough to allow tagging of word confusion networks derived from lattices, thus allowing us to explore the utility of such representations to achieve robustness to the recognition errors from the speech recognizer. We analyze different experimental conditions, including: Several adaptation levels of the ASR models to the domain, maximum entropy classifiers versus CRFs log-linear models, different tagsets (BIO versus IO), and several degrees of supervision of the discriminative log-linear models. The Wechsler Logical Memory test has been shown to be a good indicator of mild cognitive impairment (MCI) [171, 137, 190, 133], the stage of cognitive decline that is often a precursor to dementia of the Alzheimer's type. Thus, we also analyze the effectiveness of the automatically detected salient features used to assess the performance on the WLM test to classify elderly individuals as having mild cognitive impairment.

5.2 Previous work on automated assessment of narrative retellings for clinical tasks

There are a number of recent developments in applying spoken language processing techniques to automate the evaluation of clinically elicited language data. Roark and colleagues [155] automatically extract various markers from spoken responses to a neuropsychological test to analyze the usefulness of these features in discriminating subjects with mild cognitive impairment from subjects without. They automatically derive speech features such as pause frequency and duration, and also linguistic complexity features extracted from syntactic parse trees. They found that the explored linguistic complexity markers model different patterns, and therefore they convey complementary information. The classifiers trained with features derived from these markers show a better classification performance than classifiers trained with global scores of the neuropsychological test.

Hakkani-Tür and colleagues examine the feasibility of a fully automated system for the cognitive decline assessment in elderly people [51]. They manually transcribe and score the recordings from their data sets containing retellings of a story-recall test— part of a subtest in the Wechsler Memory Scale (WMS)— for detecting degradation in memory skills related to very mild dementia of the Alzheimer's type, and narrations of a picture description test used for aphasia evaluation. For the manual scoring of the first test they extract a list of 35 story elements (what they called "semantic content units") from the source narrative. If the subject's retelling contains any phrase from which the story element could be entailed or inferred, they consider that particular story element as recalled. The automated scores are extracted by computing the unigrams and bigrams overlap between the original story and the retellings. For the picture description test, they also define a set of story elements and score the retellings based on the recalling of these elements. One of the main goals of the work is to analyze the correlation between the scores obtained with manual vs. automatic transcripts.

Prud'hommeaux and Roark [146, 145] develop automated scoring models for narrative retellings based on automatic word-alignment methods. Such alignments are learned in an unsupervised manner from a parallel corpus of manual transcripts of retellings and the original source narrative, much as in machine translation training. The accuracy of the automated scoring depends on correctly identifying which of the propositions or story events from the source narrative have been included in the retelling from a predetermined list of story events. Speakers may choose to relate to these elements using diverse words or phrases. Therefore, an automated method for identifying these elements needs to model the permissible variants and paraphrasing.

As an initial step towards automation, we replace the manual transcripts used by Prud'hommeaux and Roark with automated transcripts from an ASR system [96]. The experiments are performed using the 1-best transcript from the ASR output, and therefore, the performance of the scoring model degrades significantly in high-error rate scenarios. The word-alignment models are able to capture linguistic variations that occur in the retellings, but it does not get enough information from the 1-best transcripts to overcome the ASR errors or the acoustic confusions. The models are evaluated on retellings from the Wechsler Logical Memory test.

Hakkani-Tür and colleagues compare the task of automatically scoring the retellings with natural language summarization, where the speakers need to just utter the key concepts. Following the evaluation metric from speech summarization task they compute the precision, recall and F-measure of unigrams and bigrams in the retellings with respect to the source narrative [51]. Prud'hommeaux and Roark, instead, use techniques common to automatic machine translation task, in particular, word-alignment models with parallel corpora, to automate the assessment of the narrative retellings [146, 145].

In this chapter I present an alternative approach based on conditional log-linear techniques used in word tagging tasks for spoken language understanding such as concept and semantic tagging, or named entity recognition [48, 182, 183, 87]. In contrast to previous work, we do not constrain our analysis to 1-best hypothesis from the ASR, but use multiple hypotheses in the form of word confusion networks (WCNs), which have been found to improve robustness to ASR errors in other tasks [50, 195, 108, 182, 177].

5.3 Framing the problem as a tagging task

The spoken language understanding field includes tasks that involve speech and natural language processing techniques. The applications are usually built following a sequential approach where an automatic speech recognition (ASR) system is at the front-end. Therefore, natural processing techniques originally presented for written text have to be modified to deal with noisy text containing ASR errors and a different language style. Speech input, and in particular, spontaneous speech is characterized by the frequent occurrence of disfluencies like filled pauses, repetitions, repairs and false starts. Furthermore, as spontaneous speech is often agrammatical, the performance of systems like part of speech (POS) taggers, syntactic parsers or name entity recognition (NER) systems, originally trained with written text degrades considerably. Various mechanisms have been adopted to make the systems robust to ASR errors.

One of them, adopted in our approach, consists in considering multiple candidates in the form of lattices or word confusion networks [50, 87] explained in more detail in Section 2.1.5. Illustrating with an example, consider that a person utters *get cash* and the ASR 1-best output is *the ash.* If we take as output the lattice from Figure 5.1 apart from the 1-best output we also keep other candidates where one of them is the correct transcript.

Some popular application domains of spoken language understanding are spoken dialog

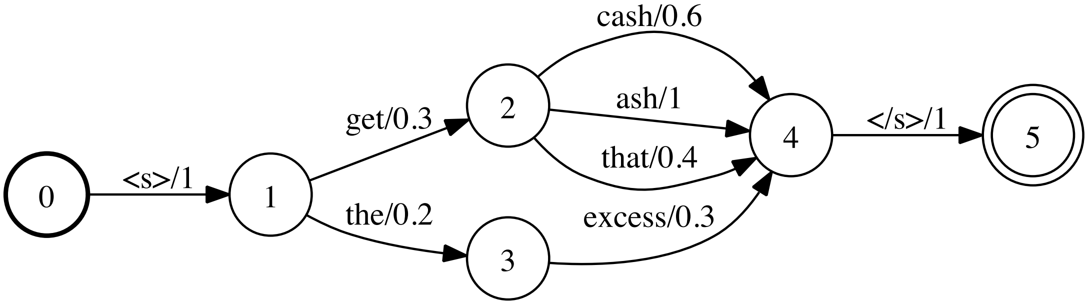


Figure 5.1: ASR output in the form of lattice.

systems, spoken information retrieval and speech translation [127], and NER is a common preliminary step in many spoken language understanding tasks. A NER task detects the segments representing entities and tags them with a specific entity such as person, location or organization. Framing the task as a tagging system, the goal is to assign a tag to each word from the sentence. The tag represents the category to which the word belongs. Different granularity levels of tagging have been proposed. We can define a tagset with just tags representing the entity (LOC, ORG, PERSON) and an additional tag for words outside any entity (O), an extended tagset with tags that also specify if the word is in the beginning (B) or inside (I) the entity—BIO models—or a more fine-grained model that also includes a tag for the last word of the entity (L) and for one word-length entities (U)—BILOU models [151]. Table 5.4 shows a tagging example with each tagset.

Tagging	*Reggie*	*Waldi*	*Blinker*	*had*	*his*	*indefinite*	*lifted*	*by*	*FIFA*
IO-tags	I-PER	I-PER	I-PER	O	O	O	O	O	I-ORG
BIO-tags	B-PER	I-PER	I-PER	O	O	O	O	O	B-ORG
BILOU-tags	B-PER	I-PER	L-PER	O	O	O	O	O	U-ORG

Table 5.1: Tagsets for NER. Adapted example sentence from Ratinov and Roth's paper [151]

By stating the NER task as a tagging problem, generative and discriminative techniques such as hidden Markov models (HMMs), maximum entropy models (ME) or conditional random fields (CRFs)—previously applied for POS tagging—can be used for estimating the NER models.

The approach we propose for the automated scoring of the narrative retellings follows the same strategy as the one used for NER, but in our case, first, the segments of interest express concepts instead of entities, and second, we do not need to consider one of the main issues of NER tasks [183]; entities often contain proper names that may be absent in the limited vocabulary of the ASR system.

In particular, I present and evaluate the use of word tagging models based on discriminative log-linear techniques (ME and CRFs) to build the automated scoring system. The evaluation of the scoring system is done with different input alternatives; with the ASR 1-best alone, with a set of transcript candidates represented as WCNs, and with manual transcripts.

5.4 Discriminative log-linear models for word tagging tasks

Log-linear models are widely used in natural language processing (NLP) tagging tasks as the discriminative alternative of hidden Markov models (HMM) [12, 138, 152, 91]. Word tagging systems assign a tag t_i from a set $\mathcal{T}$ to each word w_i of a word sequence $w_{1,N}$. The assigned tags are the ones that maximize the conditional likelihood of the tags given the input sequence: $P(t_{1,N}|w_{1,N})$.

Parameters of the discriminative log-linear model are estimated by maximizing the conditional likelihood of the training data, or what it is equivalent, by maximizing the entropy of the model given the constraints set by the training data. Replicating the explanation from the paper of Berger and colleagues [12] the idea of the maximum entropy models lies in not making any assumption apart from what we know from the data and in choosing the most uniform model that meets the contraints imposed by the data. Consider a conditional distribution $p(t|w) \in \mathcal{P}$ where $\mathcal{P}$ denotes the set of all the conditional probability distributions. Consider also a set of training sample pairs in the form of (w, t) that can be summarized in terms of its empirical distribution p̃:

$$\tilde{p}(w,t) = \frac{\sum_{w,t}(w,t)}{N} \tag{5.1}$$

where $\sum_{w,t}(w,t)$ specifies the number of times the pair (w,t) occurs in the training data and N the total number of samples.

Next, the concept of feature functions is introduced to describe facts observed in the data. A feature function is a non-negative valued function. The expected value of a feature function f with respect to the empirical distribution $\tilde{p}(w,t)$ can be expressed as:

$$E_{\tilde{p}}[f] = \sum_{w,t} \tilde{p}(w,t) f(w,t) \tag{5.2}$$

And the expected value of f with respect to the model distribution $p(w,t)$ is:

$$E_{p}[f] = \sum_{w,t} p(w,t) f(w,t) \tag{5.3}$$

As the model distribution $p(w,t)$ cannot be computed in general because of all the possible w that should be taken into account, the model distribution is rewritten as $p(w)p(t|w)$ and the probability $p(w)$ is substituted with the empirical distribution of w in the training set $\tilde{p}(w)$.

$$E_{p}[f] = \sum_{w,t} \tilde{p}(w) p(t|w) f(w,t) \tag{5.4}$$

Given a set of feature functions f_i with $i = 1, ...n$ we force our model to follow the statistics of these features in the training set by constraining the expected value $E_p[f]$ to be the same as the expected value of f in the training set $E_{\tilde{p}}[f]$:

$$E_{p}[f] = E_{\tilde{p}}[f] \tag{5.5}$$

Substituting with Equations 5.2 and 5.4:

$$\sum_{w,t} \tilde{p}(w) p(t|w) f(w,t) = \sum_{w,t} \tilde{p}(w,t) f(w,t) \tag{5.6}$$

This requirement is defined as the constraint equation. We just consider the subset $\mathcal{C}$ of conditional models $p(t|w)$ from $\mathcal{P}$ that satisfy the constraint.

$$C = p \in \mathcal{P} \mid E_{p}[f_i] = E_{\tilde{p}}[f_i] \ for \ i = 1, ..., n \tag{5.7}$$

Among all the models $p \in C$, the maximum entropy principle [60] requires the selection of the most uniform distribution. The uniformity of the conditional distribution can be mathematically expressed with the conditional entropy:

$$H(p) = -\sum_{w,t} \tilde{p}(w)p(t|w)logp(t|w) \tag{5.8}$$

Thus, based on the maximum entropy principle we select a distribution $\hat{p}$ from the subset $\mathcal{C}$ with the maximum entropy $H(p)$:

$$\hat{p} = \operatorname*{argmax}_{p \in C} H(p) \tag{5.9}$$

The distribution $\hat{p}$ with maximum entropy is unique in any constrained set C, and it can be estimated with the method of Lagrange multipliers. The Lagrangian $\Lambda(p, \alpha)$ can be defined as:

$$\Lambda(p, \alpha) = H(p) + \sum_{i} \alpha_i (E_p[f_i] - E_{\tilde{p}}[f_i]) \tag{5.10}$$

Holding α fixed, the unconstrained maximum of the Lagrangian $\Lambda(p, \alpha)$ is computed over all $p \in \mathcal{P}$. The obtained p and Lagrangian value $\Lambda(p, \alpha)$ are denoted as p_α and $\Psi(\alpha)$, respectively, and take the following form:

$$p_\alpha(t|w) = \frac{exp(\sum_i \alpha_i f_i(w,t))}{Z_\alpha(w)} \tag{5.11}$$

$$\Psi(\alpha) = -\sum_{x} \tilde{p}(w)logZ_\alpha(w) + \sum_{i} \alpha_i \tilde{p}(f_i) \tag{5.12}$$

where $Z_\alpha(w)$ is a normalization factor determined by the requirement that $\sum_t p_\alpha(t|w) = 1$ for all w:

$$Z_\alpha(w) = \sum_t exp(\sum_i \alpha_i f_i(w,t)) \tag{5.13}$$

Finally, we find the parameters α that maximize $\Psi(\alpha)$:

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmax}} \, \Psi(\alpha) \tag{5.14}$$

The log-likelihood of the model p with respect to the empirical distribution $\tilde{p}$ is defined as:

$$\mathcal{L}_{\tilde{p}}(p) = log \prod_{w,t} p(t|w)^{\tilde{p}(w,t)} = \sum_{w,t} \tilde{p}(w,t) logp(t|w) \tag{5.15}$$

It is straightforward to show that $\Psi(\alpha)$ is just the log-likelihood for the exponential model $p_\alpha(t|w)$:

$$\mathcal{L}_{\tilde{p}}(p_\alpha) = \sum_{w,t} \tilde{p}(w,t) logp_\alpha(t|w) \tag{5.16}$$

$$= \sum_{w,t} \tilde{p}(w,t) log(\frac{exp(\sum_{i=1}^{n} \alpha_i f_i(w,t))}{Z_\alpha(w)})$$

$$= \sum_{w,t} \tilde{p}(w,t)(-logZ_\alpha(w) + \sum_{i=1}^{n} \alpha_i f_i(w,t))$$

$$= -logZ_\alpha + \sum_{i=1}^{n} \alpha_i E_{\tilde{p}}[f_i]$$

$$= \Psi(\alpha)$$

Therefore, the model $\hat{p} \in C$ with maximum entropy is the model in the parametric family $p_\alpha(t|w)$ that maximizes the likelihood of the training data.

The model parameters α estimated by maximizing the conditional likelihood make the correct tag t more likely than the rest of the possible tags $t' \neq t$ for a given input w, as opposed to the case where the parameters α are estimated by maximizing the joint likelihood, where the estimated parameters make the pair (t, w) as likely as possible. Apart from their discriminative nature, conditional log-linear models also provide more flexibility than, for example, HMMs for defining the feature vectors. They do not make any independence assumptions about the features, and their framework allows the inclusion of features from multiple knowledge sources. We evaluate the performance of two kind of discriminative log-linear models: Maximum entropy classifiers [12] and conditional random fields [91] discussed in the following sections.

When the parameters of the model are trained by maximizing the likelihood of the data, as in log-linear models, it is usually necessary to add a regularization term that penalizes large parameter values and prevents overfitting to the training data. The most popular regularizers are L_1 and L_2. The L_1 regularizer produces sparse parameter vectors with many of the parameters equal to zero by penalizing the weights for its L_1 norm, reducing in this way the dimensionality of the model. The L_2 regularizer, instead, penalizes its L_2 norm, and produces parameters close to zero, but not exactly zero. The disadvantage of L_1 with respect to L_2 is that it is not differentiable, and consequently the parameters of the log-linear models cannot be efficiently estimated with optimization algorithms such as stochastic gradient descent (SGD), recently applied for NLP tasks [181].

5.4.1 Maximum entropy classifiers

Maximum entropy models are local classifiers that estimate a class, or in our case a tag, for each position i in the input sequence h. Its corresponding graphical model is depicted in Figure 5.2, and mathematically the model is expressed as:

$$p(t_i|h_i) = \frac{exp(\Sigma_{k=1}^{d} \alpha_k \phi_k(h_i, t_i))}{Z(h_i)} \tag{5.17}$$

where h_i is the input with information relative to the word w_i, and $Z(h_i)$ a normalization factor of the form:

$$Z(h_i) = \sum_{t'=1}^{T} e^{\sum_{k=1}^{d} \alpha_k \phi_k(h_i, t'_i)} \tag{5.18}$$

The probability of the entire tag sequence is computed by:

$$p(t_{1,N}|w_{1,N}) = \prod_{i=1}^{N} p(t_i|h_i) \tag{5.19}$$

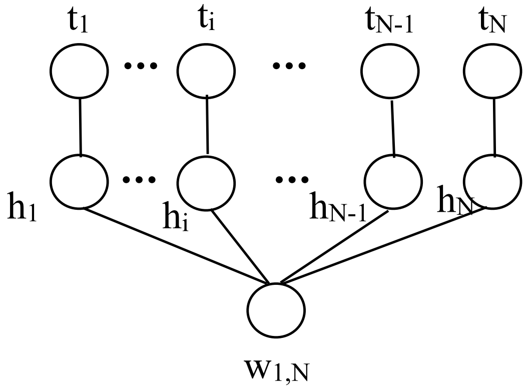


Figure 5.2: Graphical model of the maximum entropy classifier.

The parameters α_k are the parameters to learn, and $\phi_k(h, t)$ represent the feature functions, usually binary or indicator functions related to the context of the word and information from other knowledge sources with different levels of dependencies. For example, Zhang and Johnson define the following features for a NER task: (1) Previous two predictions t_{i-1} and t_{i-2} (2) current word w_i (3) w_i word type (all-capitalized, is-capitalized, all-digits, alphanumeric, etc.) (4) prefixes and suffixes of w_i (5) tokens in the window $c = (w_{i-2}, w_{i-1}, w_i, w_{i+1}, w_{i+2})$ (6) capitalization pattern in the window c (7) conjunction of c and t_{i-1} (extracted from Ratinov and Roth [151])

5.4.2 Conditional random fields

Conditional random fields (CRFs), as opposed to maximum entropy classifiers, model the conditional probability of the entire tag sequence $t_{1,N}$ given the input sequence $w_{1,N}$. It does not make any independence assumption in the estimation of the tags sequence. In a general form CRFs can consider a large variety of probabilistic relationships between the variables. For example, Sutton and colleagues apply them for a two-level dynamic CRF part of speech tagging and noun phrase chunking with better results than when modeling each task independently [175]. Skip-chain CRFs have been used to model long-distance dependencies between entities for tasks such as information extraction [174]. Linear-chain CRFs are popular for sequence labeling or tagging tasks, our task of interest. Their graphical structure is illustrated by Figure 5.3.

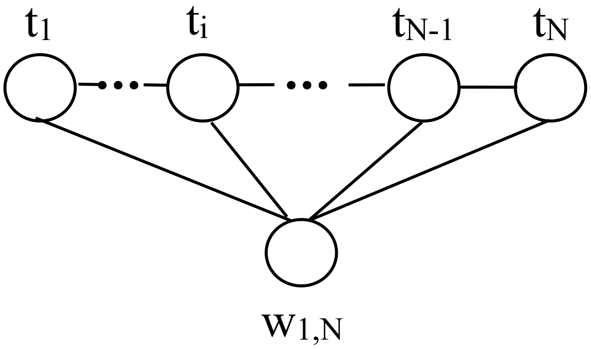


Figure 5.3: Graphical model of the linear-chain CRF.

The conditional probability of the tag sequence $t_{1,N}$ given the input sequence $w_{1,N}$ can be expressed as:

$$P(t_{1,N} \mid w_{1,N}) = \frac{\exp\left(\sum_{k=1}^{d} \alpha_k \phi_k(t,w)\right)}{Z(t)} \tag{5.20}$$

The parameters to estimate by maximizing the conditional probability $P(t_{1,N}|w_{1,N})$ (specifically, the logarithm of the conditional probability is maximized) are the parameters α_k. The term $Z(t)$ is a global normalization constant over the entire sequence —rather than local for each position as is the case for maximum entropy classifiers— represented as:

$$Z(t) = \sum_t exp\left(\sum_{k=1}^{d} \alpha_k \phi_k(t,w)\right) \tag{5.21}$$

The features $\phi_k(t, w)$ are Markov order 1 features and capture dependencies between a pair of adjacent states t_{i-1}, t_i, the whole input sequence $w_{1,N}$ and the position in the sequence i; $\phi_k(t_{i-1}, t_i, w, i)$. Additionally, it is convenient to also include redundant features for back-off, as is common in n-gram language modeling. These back-off features are Markov order 0 features that only depend on the tag at the current position, represented with the function $\phi_k(t_i, w, i)$.

5.4.3 Discriminative log-linear models with word confusion networks

By using word confusion networks (WCNs) we can get a set of acoustically confusable word candidates for each position in the sequence. The input sequence $w_{1,N}$ is extended so that instead of having just a word sequence we have a sequence of vectors $v_{1,N}$. Each vector of dimension V, the vocabulary size of the speech recognizer, represents a word confusion bin from the WCN. The bins are mapped to vectors following a bag-of-words (BOW) representation. Each word from the vocabulary is assigned an identifier, and a confusion bin is mapped to the vector by setting to one the positions of the vector that match with the identifiers of the words from the bin. The rest of the elements from the vector are set to zero. Figure 5.4 illustrates a training example for the tagging task; $(x_{1,N}, t_{1,N})$. The term $x_{1,N}$ represents any of the alternative input sequences we can use: (1) The manual transcripts $r_{1,N}$, (2) the best scoring word sequence from the speech recognizer $w_{1,N}$ or (3) the sequence of vectors representing the confusion bins from the WCN $v_{1,N}$. Each input position is paired with a tag from the tag sequence.

5.4.4 Feature templates

Given tagged or labeled examples of the form $(w_{1,N}, t_{1,N})$ the features extracted for each example are Markov order 0 (backoff features) and Markov order 1 on the tags $t_{1,N}$ for the CRFs, and just Markov order 0 features for the maximum entropy classifiers. Regarding the features that include variables from the input sequence $w_{1,N}$, we explore two settings, (1) just context-independent features, and (2) extending the set with context-dependent features that apart from the current position they also capture one position to the left and to the right. All the feature templates are illustrated in Table 5.2.

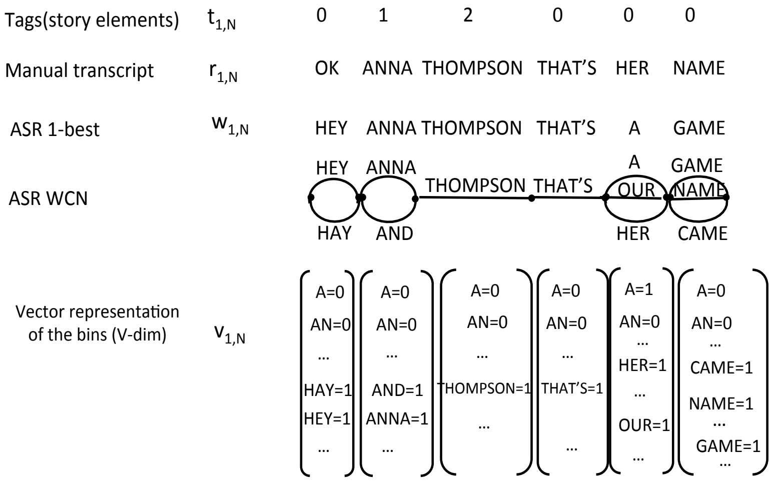

Tags(story elements) $t_{1,N}$
0 1 2 0 0 0
Manual transcript $r_{1,N}$
OK ANNA THOMPSON THAT'S HER NAME
ASR 1-best $w_{1,N}$
HEY ANNA THOMPSON THAT'S A GAME
ASR WCN
HEY ANNA THOMPSON THAT'S A GAME
OUR NAME
HAY AND HER CAME
Vector representation of the bins (V-dim) $v_{1,N}$
A=0 AN=0 … HAY=1 HEY=1 …
A=0 AN=0 … AND=1 ANNA=1 …
A=0 AN=0 … THOMPSON=1 …
A=0 AN=0 … THAT'S=1 …
A=1 AN=0 … HER=1 … OUR=1 …
A=0 AN=0 … CAME=1 NAME=1 … GAME=1 …

Figure 5.4: Input and output sequence example for log-linear models.

	Markov order 0 (ME)	Markov order 1 (CRF)
Context independent (CI)	t_i $t_i w_i$	$t_{i-1} t_i$ $t_{i-1} t_i w_i$
Context dependent (CD)	$t_i w_{i-1}$ $t_i w_{i+1}$	$t_{i-1} t_i w_{i-1}$ $t_{i-1} t_i w_{i+1}$

Table 5.2: Feature templates either using or not using neighboring tag t_{i-1} (ME vs. CRF), and using or not using neighboring words w_{i-1}, w_{i+1} (CI vs. CD).

When tagging confusion networks, each element in the input vector v_i different from zero results in a feature instance. Thus, if there are five elements (confusable words) equal to one at position i, then there will be five different w_i values being used to instantiate the features in Table 5.2. Second, following Kurata and colleagues [87], we multiply the feature counts for the context-dependent features by a weight to control their influence on the model.

5.5 Combining log-linear and word-alignment models

The word-alignment models that Prud'hommeaux and Roark [145, 146, 96] presented for the automated scoring of the retellings are generative and learned in an unsupervised way.

They are trained with expectation maximization (EM) by aligning the retelling transcripts to the source narrative without any supervised data, which makes this approach attractive. However, generative models lack the flexibility to include arbitrary features from the input due to the independence assumptions between features imposed by the model. When the retelling transcripts come from the ASR output, the word-alignment models have to capture the recognition errors. As shown in previous work [96], if the ASR output has relatively low error rate the word-alignment models are still able to score the retellings quite accurately, but in noisier conditions the models are inconsistent in providing robust automated scores. However, as the word-alignment output is still a good approximation of the correct scores, in a recent paper we propose two ways in which we can use the word-alignment output in our discriminative log-linear models [99]. In this section I describe the two ways: (1) By augmenting the feature space with the word-alignment output, and (2) by using the word-alignment output as the tags to train an unsupervised discriminative model.

5.5.1 Word-alignment derived features

Log-linear models provide a suitable framework to include arbitrary features that can overlap and do not require an independence assumption. Consequently, adding features that depend on the word-alignment output $z_{1,N}$ into the feature set is straightforward. It is also common practice to include the posterior probability from the generative ASR system into the feature set when training discriminative language models for ASR systems.

The features derived from the word-alignment output $z_{1,N}$ are created with the feature templates from Table 5.3 and incorporated into the feature set to learn the discriminative models.

5.5.2 Unsupervised discriminative tagger

The main drawback of discriminative models is the need of hand-labeled data. To train our tagging model, for example, we need (x, t) sample pairs where x is the retelling and t the manual tag sequence. Tagging the retellings is quite laborious due to the variety of paraphrases subjects use to refer to a given story element. As an alternative, we

	Markov order 0 (ME)	Markov order 1 (CRF)
Context independent (CI)	$t_i z_i$	$t_{i-1} t_i z_i$
Context dependent (CD)	$t_i z_{i-1}$ $t_i z_{i+1}$	$t_{i-1} t_i z_{i-1}$ $t_{i-1} t_i z_{i+1}$

Table 5.3: Feature templates for the word-alignment output either using or not using neighboring tag t_{i-1} (ME vs. CRF), and using or not using neighboring word-alignment output z_{i-1}, z_{i+1} (CI vs. CD).

replace manual tag sequences with the word-alignment outputs $z_{1,N}$ to have an unsupervised discriminative modeling scenario. The tag sequences will be noisy, since they are automatically generated from a stochastic model (word-alignment model) whose performance depends on the accuracy of the speech recognizer. The automatically generated tag sequences convey the noise coming from the speech recognizer and word-alignment model.

5.6 Experimental setup

5.6.1 Task

The collection of narrative retellings used in our experiments were elicited using the Wechsler Logical Memory (WLM) test [193], the same test used in previous work [146, 145, 96]. In the WLM test, the subject listens to the examiner read a brief narrative and then retells the narrative twice; immediately upon hearing it and after about 20 minutes of unrelated activities. The source narrative that the examiner reads (top paragraph) and an example retelling (paragraph at the bottom) are depicted in Figure 5.5. The story elements are separated by slashes in the source narrative, and the elements recalled by the subject are underlined in the example retelling. The examiner grades the subject's response by counting how many story elements the subject recalled from a predefined set of 25 elements. The example retelling received a score of 12, with one point for each of the recalled story elements: *Anna, Boston, employed, as a cook, and robbed of, she had four, small children, reported, station, touched by the woman's story, took up a collection* and *for her.*

Anna/ Thompson/ of South/ Boston/ employed/ as a cook/ in a school/ cafeteria/ reported/ at the police/ station/ that she had been held up/ on State Street/ the night before/ and robbed of/ fifty-six dollars/. She had four/ small children/ the rent was due/ and they hadn't eaten/ for two days/.

Ann Taylor worked in Boston as a cook. And she was robbed of sixty-seven dollars. Is that right? And she had four children and reported at the some kind of station. The fellow was sympathetic and made a collection for her so that she can feed the children

Figure 5.5: Excerpt of narrative read by the clinician (at the top) and an example retelling (at the bottom).

The clinical evaluation guidelines specify what lexical substitutions, if any, are allowed for each element. Some elements, *Thompson*, *school*, *cafeteria* and *four*, must be recalled verbatim. In other cases, subjects are given credit for variants, such as *Annie* for *Anna*, or paraphrasing of concepts such as *sympathetic* for *touched by the woman's story*.

5.6.2 Tagsets

For this task we investigate two of the three different tagsets explained in Section 5.3 for tagging the story elements. Table 5.4 shows how a retelling is tagged with each tagset. The simplest tagset (IO) simply identifies words that are in a story element; there are only 26 tags. The tag O is assigned to the words that do not belong to any story element, and tags I1 to I25 represent the 25 story elements from the source narrative.

The larger tagset (BIO) handles multi-word story elements by differentiating among positions in the story element chunk as described in Section 5.3. This tagset is only of utility for models with Markov order greater than zero, and hence are only used with CRF models.

Tagging	*anna*	*rent was due*
IO-tags	I1	I19 I19 I19
BIO-tags	B1	B19 I19 I19

Table 5.4: Two possible tagsets for labeling.

5.6.3 Data

Our models are trained on immediate and delayed retellings from 144 subjects with a mean age of 85.4, of whom 36 were clinically diagnosed with MCI (training set). We evaluate our models on a set of retellings from 70 non-overlapping subjects with a mean age of 88.5, half of whom had received a diagnosis of MCI (test set).

The recordings are sometimes made in an informal setting, such as the subject's home or a senior center. For this reason, there are often extraneous noises in the recordings such as music, footsteps, and clocks striking the hour. Although this presents a challenge for the ASR, part of the goal of our work is to demonstrate the robustness of our methods to noisy audio.

5.6.4 ASR output

Each retelling from the data set is decoded with the speech recognizer to get the best scoring word sequences and word confusion networks generated with the algorithm proposed by Mangu and colleagues [109], implemented in the IBM toolkit. We keep in the bins only those word candidates with posteriors above 0.002. We use different ASR configurations, with different levels of adaptation: (1) The baseline speech recognizer trained on out-of-domain data, without adaptation, (2) the speech recognizer with adapted acoustic models, (3) the speech recognizer with adapted language models, and (4) the speech recognizer with adapted acoustic and language models.

Baseline ASR system

The baseline ASR system used in the current work is a English Broadcast News system trained with the IBM toolkit following the recipe from Kingsbury and colleagues [74]. The acoustic models are the same as the ones used in the experiments from Chapter 4 and detailed in Section 4.3.2. The language model is defined over an 84K vocabulary and consists of about 1.8M bigrams, 1M trigrams and 331K 4-grams, estimated from standard English Broadcast News corpus. The decoding is performed in several stages using successively refined acoustic models—a context-dependent model, a vocal-tract normalized model, a speaker-adapted maximum likelihood linear regression (MLLR) model, and finally a discriminatively trained model with the boosted MMI criteria [141]. The system gives a word error rate of 13.1% on the 2004 Rich Transcription benchmark by NIST [33], which is comparable to state-of-the-art for equivalent amounts of acoustic training data. On the WLM corpus, the recognition word error rate was significantly higher at 47.2% due to a mismatch in domain and the skewed demographics (age) of the speakers.

Adapted ASR system

We improve the performance of the above English Broadcast News models by adapting to the domain of the WLM retellings. The acoustic models are adapted using standard MLLR, where linear transforms are estimated in an unsupervised manner to maximize the likelihood over the transcripts of the retellings. The transcripts are generated from the baseline system after the final stage of decoding with the discriminative model. The language models are adapted by interpolating the in-domain model (weight=0.7) with the out-of-domain model. The gains from these adaptations are reported in the Table 5.5.

System	1-best (%)	oracle WCN(%)	oracle lat(%)
Baseline	47.2	39.9	27.7
AM adaptation	38.2	30.6	21.2
LM adaptation	28.3	29.5	19.9
AM+LM adaptation	25.6	25.5	16.5

Table 5.5: Improvement in ASR WER by adapting the Broadcast News models to the domain of narrative retelling.

As expected, we find substantial gains from both acoustic model (AM) and language model (LM) adaptation. Furthermore, we find benefit in employing them simultaneously. We also include the oracle values of the WCNs and lattices for each ASR configuration.

In theory, the oracle WER for WCNs should be better than the oracle WER for lattices, since they are considering paths that are not in the lattices. However, there are a couple factors that make the oracle WER of the WCNs worse than the oracle WER of the lattices: One is the threshold applied on the WCN bins to prune word candidates with posteriors lower than the threshold, and a second factor is the alignment errors that are introduced when generating the word candidate bins of the WCNs.

The results from the table refer to the default threshold that is the one we use for our experiments, but we also computed the WERs with a lower threshold. When we lower the threshold the substitution errors are reduced, but the insertion and deletion errors are relatively unchanged. The oracle substitution error rate with WCNs is similar to the oracle substitution error rate with lattices, but the insertion and deletion error rate is significantly higher, probably due to the alignment errors. Similarly, the 1-best WER of the ASR system with adapted LMs is slightly better than the oracle WER of the WCNs, but the substitution error rate of the 1-best transcripts is higher than the oracle substitution error rate of the WCNs.

5.6.5 Automated scoring system

The log-linear models are trained with MALLET toolkit implementation for maximum entropy-based classification and sequence tagging based on linear chain CRFs [111]. The toolkit provides implementations of several classifiers, and also sequence tagging implementations for HMMs, MEMM and linear-chain CRFs.

MALLET uses L-BFGS quasi-Newton method for the estimation of the model, and by default the regularizer is L_2 with variance values of 1.0 and 10.0 for the maximum entropy classifier and linear-chain conditional random field, respectively. We use the default settings of MALLET in our experiments.

5.6.6 Results

We analyze the performance of the scoring system with three kind of input sequences: (1) ASR 1-best word sequence, (2) ASR word confusion networks and (3) the manual transcripts that will set the upper bound in the accuracy of the scoring model. The ASR 1-best and word confusion networks are extracted from several ASR configurations that differ in their adaptation level to the task data: (1) Baseline speech recognizer (out-of-domain English Broadcast News recognizer), (2) speech recognizer with domain adapted acoustic model, (3) speech recognizer with domain adapted language model, and (3) speech recognizer with domain adapted acoustic and language model. The performance of the word-alignment models, trained with manual transcripts, is also included[145, 146, 96], since its output is used with the log-linear models as an extra feature, or as the output tag sequence for the discriminative training.

Each story element is automatically tagged by the systems as either having been recalled or not, and this is compared with manual scores to derive an F-score accuracy by calculating precision and recall of the recalled story elements. We do not take into account whether the boundaries of the story elements are correctly recognized. If any of the words in the story element is correctly tagged, we consider that story element as recalled. This is a reasonable approach to take, since the manual tagging annotations are quite noisy regarding the boundaries of the recalled story elements. Also, the input sequences are prefiltered before providing them to the scoring system by eliminating function words.

Source narrative	*in*	*a*	*school*	
Manual transcript:	*at*	*the*	*public*	*schools*
Manual tags	I7	I7	I7	I7
Automatic transcript	*were*	*schools*		
Automatic tags	O	I7		
Recalled element	7			

Table 5.6: Boundary mismatch between manual and automatic tagging.

Log-linear based automated scoring

When looking at the scoring performance of the discriminative log-linear models we explore the impact of including the context information from the input and output sequence by running the following experiments: (1) Without any context information, (2) with context information from the input sequence, (3) with context information from the output sequence, and (4) with context information from the input and output sequence.

Context-independent features. Table 5.7 summarizes the accuracy of the maximum entropy classifier and conditional random fields when we do not consider any context information from the input sequence. Maximum entropy classifiers do not either consider the context of the tag sequence while conditional random fields just look at the value from the previous tag (Markov order 1).

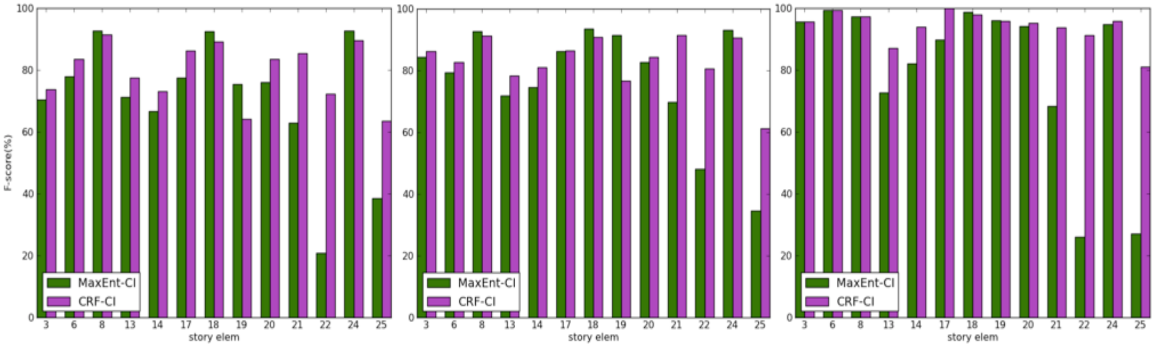
Transcripts	1-best				WCN				manual
ASR	base	AM	LM	AM+LM	base	AM	LM	AM+LM	N/A
Word-alignment	71.9	77.3	84.3	85.4	N/A				88.8
ME	76.0	81.7	84.6	85.6	78.9	83.4	84.0	84.7	86.4
CRF	80.3	87.3	89.7	91.4	83.7	88.8	88.2	90.8	94.4

Table 5.7: Story element F-score achieved by log-linear models (ME and CRF) using **context independent** features (CI) with 3 different inputs (1-best ASR, word confusion network, and manual transcripts) and different ASR models (baseline out-of-domain, AM adapted, LM adapted and AM+LM adapted).

Maximum entropy classifiers beat word-alignment models just in noisy conditions. The average F-score is similar for both models in low error rate scenarios. Conditional random fields perform significantly better than maximum entropy classifiers in all the configurations, especially in low error rate scenarios, when the transcripts are generated with the fully adapted ASR system or with manual transcripts.

Word confusion networks are most effective in noisy scenarios. As the error rate of the transcripts decreases the number of word candidates in the bins gets smaller, and therefore, the performance with word confusion networks gets closer to the performance with the best scoring word transcript candidate. With domain adapted language models the performance is a bit worse, probably because the candidates from the bin that are kept after the pruning of the candidates with posteriors lower than the threshold are not actually acoustically confusable candidates.

Figure 5.6 illustrates the F-scores for the story elements whose accuracy differs with maximum entropy classifiers and CRFs, with the unadapted speech recognizer. The graph from the left refers to the 1-best transcript (figure(a)), the graph from the middle to the WCNs (figure (b)), and the graph from the right to the manual transcripts (figure (c)). Story elements related to *cafeteria, small children, the rent was due* and *took up a collection* are better recognized with maximum entropy classifiers, although this difference vanishes with manual transcripts. For the rest of the story elements from the graphs, CRFs perform better, especially, for story elements related to *The police* and *for her*. This difference may come from the fact that the story elements *for her* contain words that could also be common in parts of the retelling that do not belong to any story element. Regarding the story element *The police* the Markov order 1 features help in disambiguating this story element with story element (10): *at the police*.



(3) of South (*South* in any context)
(6) as a cook (*cook* or some form of the word)
(8) cafeteria (*cafeteria*)
(13) on State Street (*State Street* in any context)
(14) the night before (indication that the hold-up occurred the previous night)
(17) she had four (*four* is required with an indication that the children were hers)
(18) small children (*children* or a synonym is required)

(19) the rent was due (phrase indicating that the rent was due)
(20) and they had not eaten (indication that her children or family were without food)
(21) two days (*two days* or phrase meaning about two days)
(22) The police (word or phrase signifying one or more members of the police department, in any context)
(24) took up a collection (phrase indicating that money was collected)
(25) for her (indication that the money collected was for her or her children)

Figure 5.6: Story elements whose F-score differs for ME and CRF models. (a) ASR 1-best, (b) ASR WCNs, (c) manual transcripts

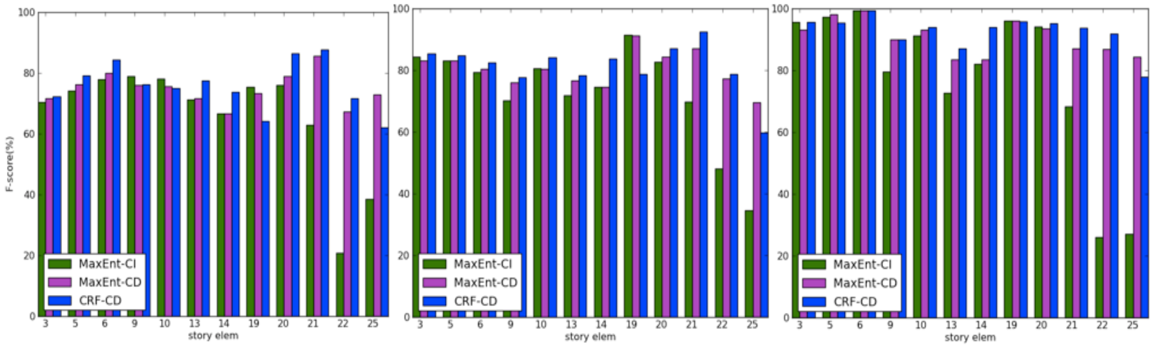
Context-dependent features and BIO tagset. Next, we include context information from the input sequence. In particular, we add to the input information of the current position the input values from the previous and next position in the sequence. We use a scaling weight of the context-dependent features equal to 0.3. We also tried a couple more values (0.5 and 1.0), but 0.3 gave the best results.

Transcripts	1-best				WCN				manual
ASR	base	AM	LM	AM+LM	base	AM	LM	AM+LM	N/A
ME	80.1	87.3	90.0	91.1	83.5	88.6	88.2	90.3	93.3
CRF-IO	80.6	88.0	89.9	91.2	84.2	89.6	88.8	90.5	94.7
CRF-BIO	81.1	87.9	90.6	91.7	84.5	89.5	88.8	90.8	94.7

Table 5.8: Story element F-score achieved by log-linear models (ME and CRF) when adding **context dependent** features (CD) and **BIO tags** for the CRF models, with 3 different inputs (1-best ASR, word confusion network, and manual transcripts) and different ASR models (baseline out-of-domain, AM adapted, LM adapted and AM+LM adapted).

When including the context information the gap between maximum entropy classifiers and conditional random fields gets significantly reduced. The average F-scores are similar for both models in all conditions. The use of word confusion networks over the best scoring word sequence is just effective in noisy scenarios. Using a more fine-grained tagset does not bring significant improvements on the average F-score obtained with the CRF models, possibly because of the limited training data.

Figure 5.7 gathers story elements whose F-score gets degraded or improved after considering the context from the input sequence. For example, the story element that refers to the word “reported” benefits from the context information, probably because the set of words used to recall this story is quite variable. This improvement is not seen with ASR 1-bests, possibly, because of the high degree of noise in the transcripts. Context information is also helpful to detect that when the word “street” goes in some specific context it belongs to a story element (story element 13). Also, the detection of story elements “and they had not eaten”, “two days”, “The police” and “for her” considerably improves with context related features. However, for the story element ”for her” taking into account the previous tag on the output sequence seems to degrade the performance.



(3) of South (*South* in any context)
(5) employed (indication that she held a job)
(6) as a cook (*cook* or some form of the word)
(9) reported (formal statement was made to someone in authority)
(10) at the police (police in any context)
(13) on State Street (*State Street* in any context)

(14) the night before (indication that the hold-up occurred the previous night)
(19) the rent was due (phrase indicating that the rent was due)
(20) and they had not eaten (indication that her children or family were without food)
(21) two days (*two days* or phrase meaning about two days)
(22) The police (word or phrase signifying one or more members of the police department, in any context)
(25) for her (indication that the money collected was for her or her children)

Figure 5.7: Story elements whose F-score differs for ME and CRF models when adding input context. (a) ASR 1-best, (b) ASR WCNs, (c) manual transcripts

Combining word-alignment and log-linear models

After studying the supervised scoring models, we use the output from the word-alignment model to explore two new scenarios: (1) Hybrid discriminative tagging and (2) unsupervised discriminative tagging explained in Section 5.5.1 and Section 5.5.2, respectively.

Context-independent features. As shown in Table 5.9, the unsupervised scenario follows the same trend as the supervised scenario; CRFs perform better than maximum entropy classifiers. However, in the unsupervised case, the higher accuracy of the maximum entropy classifiers over the word-alignment models disappears with low error rate input sequences. As in the supervised case, the word confusion networks are effective only with noisy inputs. The unsupervised models have lower accuracy than their supervised counterparts, but they are still competitive.

The hybrid discriminative models give the best F-scores among the three scenarios. In this case, using CRFs does not provide consistent gains over the maximum entropy classifiers, probably because the word-alignment output is a strong predictor of the output tag, and the Markov order 1 features do not add extra information to the discrimination. Furthermore, using word confusion networks is not helpful even with noisy inputs. The performance of the hybrid model with the transcripts from the fully adapted ASR system is just over 2% worse than with manual transcripts.

Training	Transcripts	1-best				WCN				manual
Scenario	ASR	base	AM	LM	AM+LM	base	AM	LM	AM+LM	N/A
Word-alignment		71.9	77.3	84.3	85.4	N/A				88.8
Sup.	ME-CI	76.0	81.7	84.6	85.6	78.9	83.4	84.0	84.7	86.4
	CRF-CI	80.3	87.3	89.7	91.4	83.7	88.8	88.2	90.8	94.4
Unsup.	ME-CI	72.1	79.3	82.7	84.2	77.5	81.2	83.4	83.2	84.8
	CRF-CI	79.4	85.4	86.8	88.0	81.2	85.8	86.2	87.2	90.5
Hybrid	ME-CI	88.1	89.4	89.2	89.6	87.6	89.2	88.8	89.5	91.8
	CRF-CI	87.0	90.9	91.5	92.1	87.4	91.5	90.1	92.4	94.6

Table 5.9: Story element F-score achieved by combining word-alignment model and log-linear models (ME and CRF) using **context independent** features (CI) under 3 different scenarios (supervised, unsupervised and hybrid), with 3 different inputs (1-best ASR, word confusion network, and manual transcripts) and different ASR models (baseline out-of-domain, AM adapted, LM adapted and AM+LM adapted).

Context-dependent features and BIO tagset. When including features related to the context information from the input sequence, the performance of the unsupervised maximum entropy classifiers is similar to the performance of the unsupervised CRFs. However, with noisy transcripts the CRFs show modest gains over the maximum entropy classifiers, as shown in Table 5.10.

The hybrid models do not get any significant benefit from the context dependent features. As mentioned earlier the features related to the word-alignment output are a good predictor of the output tag, and the contribution of the rest of the features gets reduced.

As in the supervised scenario the more fine-grained tagset (BIO tagset) does not provide any significant gains either in the unsupervised or hybrid scenarios.

Training	Transcripts	1-best				WCN				manual
Scenario	ASR	base	AM	LM	AM+LM	base	AM	LM	AM+LM	N/A
Sup.	ME-CD	80.1	87.3	90.0	91.1	83.5	88.6	88.2	90.3	93.3
	CRF-CD-IO	80.6	88.0	89.9	91.2	84.2	89.6	88.8	90.5	94.7
	CRF-CD-BIO	81.1	87.9	90.6	91.7	84.5	89.5	88.8	90.8	94.7
Unsup.	ME-CD	77.1	83.1	86.5	89.0	80.2	85.0	86.2	87.6	90.7
	CRF-CD-IO	79.1	85.3	87.1	88.3	81.0	85.9	86.4	87.5	90.3
	CRF-CD-BIO	79.1	85.6	87.2	88.4	81.3	85.9	86.2	87.3	90.6
Hybrid	ME-CD	88.4	90.2	90.7	91.6	88.6	90.5	90.4	91.4	93.5
	CRF-CD-IO	87.9	91.3	91.6	92.5	88.3	91.7	90.7	92.1	94.8
	CRF-BIO	87.8	91.9	91.8	93.0	88.7	92.0	90.7	92.3	94.7

Table 5.10: Story element F-score achieved by combining word-alignment model and log-linear models (ME and CRF) when adding **context dependent** features (CD) and **BIO tags** for the CRF models, under 3 different scenarios (supervised, unsupervised and hybrid), with 3 different inputs (1-best ASR, word confusion network, and manual transcripts) and different ASR models (baseline out-of-domain, AM adapted, LM adapted and AM+LM adapted).

5.6.7 MCI classification

The main goal of this chapter is to explore the best practices to accurately detect in retellings the story elements from a source narrative. However, as Prud'hommeaux and Roark empirically demonstrated that features extracted from the retellings of the WLM test are useful in distinguishing subjects with mild cognitive impairment (MCI) from neurotypical age-matched controls [145, 146, 96], we also run the MCI classification experiments with the output from our log-linear models to see how improvements in the scoring systems translate into the classification.

Each of the individuals producing retellings in our corpus underwent a battery of neuropsychological tests, and were assigned a clinical dementia rating (CDR) [128], which is a composite score derived from measures of cognitive function in six domains, including memory. Importantly, it is assigned independently of the WLM test, which was used to elicit the retellings analyzed here, allowing us to evaluate the utility of our WLM analyses in an unbiased manner. MCI is defined as a CDR of 0.5 [154], and subjects in this study have either a CDR of 0 (no impairment) or 0.5 (MCI).

Each subject produces two retellings: One immediately after hearing the original narrative, and another after a delay of 20 to 30 minutes. From each retelling, we extract Boolean features for each story element indicating whether the retelling contains that story element, plus a count feature representing the total score assigned to the retelling. In total, there are 52 features for the classification. We use the classification setup from Prud'hommeaux and Roark [145, 146, 96] where a support vector machine (SVM) is built using the LibSVM [19] extension to the Waikato environment for knowledge analysis (WEKA) data mining Java API [52]. The SVM is trained with the radial basis function kernel and default parameter settings.

From all the discriminative models analyzed in Section 5.6.6 we select only the models with the highest F-score performance to run the MCI classification. The experiments are run with the three types of input and the four ASR configurations under the supervised, unsupervised, and hybrid scenarios. Classification results, evaluated as area under the curve (AUC), are reported in Table 5.11. The values of AUC are in the range of 0 to 1, with 1 representing the highest score for classification. We also include the classification accuracy obtained with the word-alignment models. For reference (not shown in the table), the SVM classifier performed at 0.83 when features values are manually populated.

The results show that the supervised approach does not tend to provide a better MCI classification than the unsupervised approach. Overall, the best classification values are given by the hybrid approach, which performs slightly better than the other two approaches. The best AUC in the hybrid scenario (0.79, very close to the AUC=0.81 achieved with manual transcripts) is obtained with a CRF trained with WCNs from the

Training	Transcripts	1-best				WCN				manual
Scenario	ASR	base	AM	LM	AM+LM	base	AM	LM	AM+LM	N/A
Word-alignment		0.65	0.67	0.74	0.76	N/A				0.79
Sup.	ME-CD	0.65	0.73	0.76	0.77	0.70	0.73	0.77	0.77	0.81
	CRF-CD-BIO	0.69	0.76	0.77	0.76	0.73	0.76	0.77	0.78	0.82
Unsup.	ME-CD	0.65	0.72	0.75	0.76	0.70	0.75	0.75	0.76	0.80
	CRF-CD-BIO	0.74	0.75	0.78	0.78	0.71	0.74	0.77	0.76	0.81
Hybrid	ME-CD	0.72	0.76	0.77	0.78	0.74	0.76	0.77	0.77	0.82
	CRF-CD-BIO	0.72	0.76	0.78	0.78	0.76	0.77	0.78	0.79	0.81

Table 5.11: Classification performance (AUC) for the word-alignment model and the best performing log-linear models of both types (ME and CRF) under 3 different scenarios (supervised, unsupervised and hybrid) with 3 types of input and 4 types of ASR models.

fully adapted ASR system and with context dependent features.

In our experiments, using WCNs as input is more effective for the out-of-domain ASR configuration. For the adapted ASR configurations, we do not observe any significant gains by using WCNs versus 1-best inputs.

Comparing the ME model with the CRF model for detecting the story elements, we see that the average F-scores for the ME models trained on CD features are nearly as good as and sometimes slightly better than those produced using the CRF models. The CRF extracted story elements, however, tend to yield classifiers that perform slightly better, especially in the unsupervised approach with 1-best inputs.

5.7 Discussion

Motivated by the benefits of automating the administration and assessment of tasks administered as part of a neurological evaluation, there is active research on exploring the potential of spoken and natural language processing techniques to achieve this goal. Hakkani-Tür and colleagues compute the overlap of the unigrams and bigrams between the retellings (manual and automatic transcripts) and the reference narrative, and find the correlation between the F-scores of the ngrams overlap and the manual scores assigned to the retellings. As opposed to our work they do not attempt to detect each of the individual key elements of the narrative, but just compute the final score assigned to the retelling. Also, their approach does not capture the lexical variations that occur when subjects retell the narrative with their own words.

This chapter further investigated the task defined in the work published by Prud'hommeaux and Roark [146, 145]. We explored the same task, but from a different perspective. Instead of using word-alignment models similar to the ones used in machine translation systems, we framed the task as a tagging problem based on the techniques commonly used in spoken language processing systems such as name entity recognition or semantic tagging. In contrast to word-alignment models, generative and unsupervised, our tagging models are discriminative and supervised, and although there is previous work proposing speech translation decoding with word confusion networks [13, 14], our tagging models provide a more appropriate framework to take into account a set of ASR output candidates. Specifically, we explored the use of word confusion networks with maximum entropy classifiers and conditional random fields. One thing to note is that the use of conditional random fields with word confusion networks has not been applied for semantic parsing until very recently [182], a little bit later than our publication of the work from this chapter. They found that training the CRFs both with manual transcripts and the WCNs improves the performance of the model. In our experiments we found WCNs to be just useful in noisy scenarios, so as future work we should try incorporating the manual transcripts when training the discriminative models with WCNs.

With respect to the features used to train the conditional log-linear models, the presented results refer to models trained with binary feature functions. An alternative is to use real-valued features by including the posterior probability information of the recognized word candidates provided by the ASR system. We trained the models with ASR posterior probabilities, but we did not get consistent gains. Tür and colleagues included the posterior probability information only during the decoding [182].

When we used the outputs of the word-alignment models to train the discriminative models in an unsupervised way, we used the simple approach of just considering the best scoring output $z_{1,N}$. However, we could instead consider a distribution of tag sequences (multiple-label tagging), in a similar way as it is done for the input sequence when this input is automatically generated (multiple-instance tagging). There is previous work that looks into this problem, where the noisy labels could not just be generated automatically, but alternatively, they could come from the annotation of multiple labelers [64, 31]. Dredze and colleagues [31] present a modified CRF approach to take into account multiple label sequences in the training. The set of labels assigned to each input position also contains prior probabilities to take into account how likely the labels are. They analyze the impact that the quantity of the training data and the noise level on the labels has on the usefulness of the approach. Their results show that keeping multiple labels is effective when we have limited training data or when the labels are very noisy. The training data we use for our experiments is limited and, depending on the ASR configuration, the tags that the word-alignment outputs can be noisy too, so the multiple-labeling approach could be an interesting future line to explore.

5.8 Summary

This chapter examines the task of automatically scoring narrative retellings in terms of their fidelity to the original narrative content, using discriminatively trained log-linear tagging models for detecting the recalled story elements. Fully automated scoring systems must account for both lexical variations and acoustic confusions from the ASR errors. Lexical variation—due to paraphrasing on the part of the individuals retelling the narrative—can be modeled effectively using word-alignment models such as those employed in machine translation systems [96, 145]. However, when the retelling transcripts are automatically generated by an ASR system, the word-alignment models may be not capable of consistently learning the variation patterns. In this chapter we presented an alternative approach to the word-alignment models, where both lexical variation and ASR confusions are modeled using log-linear models by using features extracted from ASR word confusion networks. In addition to allowing very flexible feature definitions, they are discriminative, so they provide more accurate element F-score than unsupervised word-alignment models. We evaluated our systems with ASR outputs from ASR systems with varying levels of adaptation to the task data. We found the following trends. First, using word confusion networks improved the detection accuracy of the story elements in high error rate scenarios, that is, with the unadapted speech recognizer. Second, the performance with the fully adapted speech recognizer was close to the performance with manual transcripts. Third, CRFs were significantly better when only context independent features are extracted from the input sequence. When adding context dependent features the performance with maximum entropy classifiers and CRFs is similar. Four, we did not find additional gains from using a more complex tagset where we used different tags for words in the beginning and inside the story element.

The feature definition flexibility of the log-linear models allowed us to incorporate the unsupervised word-alignment labels into these models, resulting either in fully unsupervised approaches that performed competitively with the supervised models or in hybrid (supervised) approaches that provided the best performing systems in this study.

Finally, the automatically detected story elements were employed for the classification of patients according to having mild cognitive impairment or not. The story elements detected with the discriminative log-linear models yielded classification accuracies of 0.79 in terms of AUC (with 1 being the most accurate), 4% worse than what we obtain with manually detected story elements (AUC=0.83).

Chapter 6

Summary and future work

6.1 Summary

WFST-based speech recognition systems provide optimal representation of the decoding graph $\mathcal{T}$ and efficiency for the search. The WFST-based decoding graph is the result of composing the speech recognizer components (acoustic models, context dependent decision trees, pronunciation lexicon and language model) that are trained and optimized independently. However, a more accurate approach might be to consider the correlations between components when estimating the parameters and performing the estimation according to the target objective function. We explored the joint estimation of the parameters from the speech recognizer components by discriminatively optimizing the word error rate in Chapter 3 and Chapter 4. In particular, we applied discriminative models that belong to the global linear models, commonly used for discriminative language models [156], and trained them with the structured perceptron algorithm.

In Chapter 3 the joint discriminative models were used to estimate the parameters from the acoustic and language components as well as parameters from the duration model to improve the performance of the GALE Arabic transcription task, a NIST benchmark.

The presented joint discriminative acoustic and language models can be factored into two WFSTs which can be applied to the input ($\mathcal{E}$) and the output side ($\mathcal{D}$) of the general model ($\mathcal{T}$) for the speech recognition $\mathcal{E} \circ \mathcal{T} \circ \mathcal{D}$, and thus can be used in first pass recognition as well as for re-scoring. Empirical results demonstrated that the presented joint model improved the performance by 1.4%-1.6% absolute on a GALE Arabic transcription task. The goal of the work from this chapter was to learn the weights of the WFST-based speech recognizer and as such we just estimated the acoustic state transitions from the acoustic models. The observation probabilities were not modified.

The experimental analysis of the joint model suggests the following key points. First, acoustic state and duration n-grams are more effective than lexical n-grams for Arabic transcription task. Second, duration features, usually ignored in large vocabulary speech recognizers show robust gains across genres (Broadcast News vs. Broadcast Conversations). Third, from the experimental results it is expected that further gains may be obtained with additional training data.

In the next chapter (Chapter 4) the joint discriminative models presented in Chapter 3 were applied to learn a pronunciation variation model that captures phonological differences between African American Vernacular English (AAVE) and canonical pronunciations of American English (pronunciations usually included in the lexicon of standard speech recognizers). The discriminative pronunciation model was applied for the adaptation of an out-of-dialect speech recognizer. We ran the experiments in parallel for two dialectal sets: AAVE and standard American English (SAE) dialects. The discriminative pronunciation variation models were more effective to improve the performance on AAVE data than on SAE data, as expected. The gains observed for AAVE dialect were, in general, approximately twice than those observed for SAE dialect, since the canonical pronunciations used to train the out-of-dialect speech recognizer are well-matched to the SAE.

Leveraging the flexibility of our discriminative models, we also explored modeling pronunciation at acoustic state level, and jointly estimating parameters of pronunciation and language models. Discriminative models with acoustic state features performed slightly better than the models with phonetic features. Therefore, if the goal of the task is simply adapting the recognizer to the target dialect and not extracting any pronunciation characteristics or cues, the discriminative acoustic state models are a better option. In addition to performing slightly better than the discriminative phonetic features they are more straightforward and simple (there is no need to generate canonical and surface phone sequences). A discriminative model that jointly estimates the parameters of the language model and the pronunciation model improved the recognition performance on the AAVE data by about 2.1% WER of which about 0.5% can be attributed to pronunciation models. On the other hand, improvements on the SAE data were lower and mainly attributable to the discriminative language model.

Combining acoustic state and phonetic features did not give additional gains over just using acoustic state features. The feature space of the acoustic state features was much larger than the space of the phonetic features, and they were also frequently activated in the model training, so when using both feature spaces together the acoustic state features may conceal the contribution of the phonetic features.

Additionally, we studied the impact of duration features since it is a relevant cue in pronunciation. Including duration features into the discriminative pronunciation models did not provide significant gains. Discriminative duration models and discriminative pronunciation models performed similarly, so duration features may also be capturing pronunciation characteristics of AAVE dialect, but the feature space of discriminative duration models was much larger than the feature space of the pronunciation models.

We also studied how varying acoustic models of the out-of-dialect speech recognizer can impact the behavior of our discriminative models. We employed four acoustic model configurations for the out-of-dialect ASR system: (1) English Broadcast News generative speaker adapted models, (2) English Broadcast News BMMI discriminative models, (3) English Broadcast News BMMI models adapted with MLLR using SAE training data, and (4) English Broadcast News BMMI models adapted with MLLR using AAVE training data. First, using generative or discriminative acoustic models in the out-of-dialect speech recognizer did not impact the behavior of our discriminative models. The estimated discriminative models performed similarly for both acoustic models. Second, when just adapting the channel mismatch between the out-of-dialect speech recognizer and the StoryCorps data, by adapting the out-of-dialect BMMI models with MLLR adaptation using SAE data, the overall WER improved, and the performance of our discriminative models with the adapted speech recognizer was comparable to the performance with the unadapted recognizer. Third, when the acoustic models were adapted with MLLR using AAVE data, our discriminative pronunciation models could still provide further gain.

Finally, in Chapter 5 we examined how jointly modeling acoustic and lexical variations can improve the performance of a downstream application, a narrative retelling assessment task. In particular, we presented discriminative log-linear models that provide a suitable framework to capture acoustic confusions from the speech recognizer as well as lexical variations and paraphrasing present in the retellings. We borrowed techniques common from spoken language understanding and used them with ASR outputs in the form of word confusion networks. Using word confusion networks as opposed to just ASR 1-best candidates was beneficial in noisy scenarios, but in our experiments their effectiveness vanished when the speech recognizer was adapted to the task data. We demonstrated gains of 6.3% over a generative baseline in the F-score of detecting story elements on a narrative retellings-based clinical task, the Wechsler Logical Memory task.

In previous work Prud'hommeaux and Roark [146, 145] used word-alignment models for the same task, that is, to automatically detect the recalled story elements in the narrative retellings of the Wechsler Logical Memory test. Word-alignment models have the benefit of being unsupervised, and therefore, not requiring manual tagging of the retellings. We ran some experiments in joint work with Prud'hommeaux and Roark to explore two ways of combining word-alignment models with our discriminative log-linear models. First, we extended the feature space of our log-linear models by generating new feature templates containing the word-alignment output (what we called hybrid models). The experimental results showed that these models were robust and could provide accurate scoring. Second, the word-alignment output was used to train the discriminative log-linear models. This allowed us not to need manual tags and achieve an unsupervised discriminative approach. The performance was worse than in the supervised case, but the results were still comparable.

As previously done by Prud'hommeaux and Roark [146, 145] we employed the automatically detected story elements for the classification of subjects according to having mild cognitive impairment or not. The story elements detected with the discriminative log-linear models yielded classification accuracies of 0.79 in terms of AUC (with 1.0 being the most accurate), about 4% worse than what it was obtained with manually detected story elements (AUC=0.83).

6.2 Future directions

In the first part of the work we learned joint discriminative acoustic, language and duration models. These models significantly improved the performance of the baseline large vocabulary Arabic speech recognizer. However, the acoustic models from our speech recognizer were not as refined as the acoustic models used in current speech recognizers. The question is then whether our acoustic state and duration features are still effective with more accurate acoustic models. This is something to investigate in further experiments. The benefit of these features with state of the art acoustic models may be dependent on the domain. If our discriminative models are applied for English Broadcast News, they may show limited gains as reported in a recent publication [4]. However, it may not be the case for the GALE Arabic transcription task with Arabic Broadcast News and Broadcast Conversation data. This data is commonly unvowelized and with speech from different dialects. Discriminative acoustic models therefore need to capture more variability than, for example, in English Broadcast News. In the large vocabulary Arabic speech recognition task from our experiments, Soltau and colleagues [168] show that using discriminative acoustic models provides similar benefits as vowelizing the data, but there is the dialectal variability that needs to be captured. Discriminative acoustic models (discriminative GMMs) may not be capturing that variability completely, and our discriminative models may capture this variability more effectively than the discriminative GMMs.

Instead of optimizing the acoustic models with BMMI discriminative training independently from the optimization of the other components of the speech recognizer, we could extend our joint discriminative models to also estimate the parameters from the GMMs by applying the online algorithm for large margin training proposed by Cheng and colleagues [22, 23]. By using the reparameterization for GMMs from Sha and Saul work [163], the mixture weight, mean and covariance of the Gaussian distribution can be represented in a single matrix Φ, and therefore, during the online training only one update must be done. This update can be easily combined with the update of the current parameters (acoustic state and word transition weights) from our joint discriminative model.

With respect to the discriminative pronunciation modeling work from Chapter 4, there are several lines to work on further. Our pronunciation variation model contains unigram and bigram phone transformation features, but we could instead use triphone transformation features where the context is one phone transformation to the left and one phone transformation to the right, similar to what it is done when modeling context dependent acoustic models. Also, instead of applying the phonological rules at word level, we could also include cross-word rules and apply them at utterance level. Using data-driven phone transformations instead of or in addition to knowledge-based phonological rules is another alternative. When using data-driven phone transformations, incorporating the parameters from the language model into the discriminative model may be particularly useful. The feature set with phonetic transformations extracted from data may be large, and incorporating features from the language model may allow us to include only the transformations that correct errors not captured by the features from the language model.

We applied our discriminative models either with phonetic features or acoustic state features to capture pronunciation characteristics of speakers of a dialect, in particular, the AAVE dialect, but this approach could potentially be applied to capture and learn pronunciation characteristics of a set of speakers sharing some specific pronunciation patterns. An interesting line of research would be to apply the presented discriminative models to extract pronunciation patterns from children's speech, from speech of elderly people, or speech from individuals with motor speech disorders such as dysarthria.

Thus far, we have applied our discriminative models with baseline speech recognizers with GMM-based acoustic models. Currently, neural network-based acoustic models are being adopted. Then, the next step would be to analyze the performance of our discriminative models with neural network-based speech recognizers. Our discriminative models should still be useful, since we do not estimate parameters from the emission model of the acoustic models.

Regarding the work done on automating the assessment of the narrative retelling tasks, we could use recurrent neural networks instead of log-linear models. Lately, researchers on spoken language understanding field are exploring the gains that using neural networks-based models can bring over using conditional random fields [113, 198, 30]. Recurrent neural networks-based models have the potential to capture long-span dependencies and consider more complex features.

Additionally, we could extend the presented scoring models to apply them on other neuropsychological tests and other fields such as education. ETS[1], for example, has done extensive work on automating the evaluation of their written and spoken tests for second language learners. In general, their work has focused on deriving a single final score that represents the performance of the student on the given test. Specifically, in the automated analysis of spoken tests, they apply techniques originally used to measure similarity between text documents to compute the similarity between the spoken responses and a set of pre-scored responses or prompts related to the test. In a recent work they automatically evaluate if students include some relevant concepts that should be retold in their answers [196]. Therefore, further research we could do is to apply our scoring models to automate these sort of tests.

[1] `www.apple.com/ios/siri/`

Bibliography

[1] Abdelhamid, A. A., and Abdulla, W. H. Joint discriminative learning of acoustic and language models on decoding graphs. In *Signal and Information Processing Association Annual Summit and Conference (APSIPA)* (2013), pp. 1–5.

[2] Abdelhamid, A. A., and Abdulla, W. H. On the robustness of joint optimization on transducer-based decoding graphs. In *TENCON* (2013), pp. 362–365.

[3] Ahmed, S., and Garrard, P. Spoken discourse in alzheimer's disease: A review. *Linguistica* (2012), 1(52).

[4] Arisoy, E., Ramabhadran, B., and Kuo, H.-K. J. Feature combination approaches for discriminative language models. In *Conference of the International Speech Communication Association (Interspeech)* (2011).

[5] Arisoy, E., Saraçlar, M., Roark, B., and Shafran, I. Syntactic and sublexical features for turkish discriminative language models. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2010), pp. 5538–5541.

[6] Bacchiani, M., and Roark, B. Unsupervised language model adaptation. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2003), pp. 224–227.

[7] Bahl, L., Bakis, R., Jelinek, F., and Mercer, R. Language-model/acoustic-channel-model balance mechanism. *IBM Technical Disclosure Bulletin* (1980).

[8] Bahl, L., Brown, P., de Souza, P., and Mercer, R. Maximum mutual information estimation of hidden markov model parameters for speech recognition. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (1986), pp. 49–52.

[9] Baker, J. K. Stochastic modeling for automatic speech recognition. *Speech Recognition, D. R. Reddy, Ed. New York: Academic* (1975).

[10] Bellegarda, J. R. Statistical language model adaptation: review and perspectives. *Speech Communication 42* (2004), 93–108.

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

[12] Berger, A. L., Pietra, S. A. D., and Pietra, V. J. D. A maximum entropy approach to natural language processing. *Journal for Computational Linguistics* (1996), 22(1):39–71.

[13] Bertoldi, N., and Federico, M. A new decoder for spoken language translation based on confusion networks. In *Automatic Speech Recognition and Understanding Workshop (ASRU)* (2005).

[14] Bertoldi, N., Zens, R., Marcello, F., and Shen, W. Efficient speech translation through confusion network decoding. In *IEEE Transactions on Audio Speech and Language Processing* (2008), pp. 116(8):1696–1705.

[15] Biadsy, F., Hirschberg, J., and Habash, N. Spoken arabic dialect identification using phonotactic modeling. In *EACL Workshop on Computational Approaches to Semitic Languages* (2009).

[16] Bisani, M., and Ney, H. Investigations on joint-multigram models for grapheme-to-phoneme conversion. In *International Conference on Spoken Language Processing (ICSLP)* (2002), pp. 105–108.

[17] Breiman, J., Friedman, J., Olshen, R., and Stone, C. *Classification and Regression Trees.* Pacific Grove, CA: Wadsworth and Brooks, 1984.

[18] Campbell, W. Segment durations in a syllable frame. *Journal of Phonetics* (1991), 37–47.

[19] Chang, C.-C., and Lin, C.-J. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology 2*, 27 (2011), 1–27.

[20] Chen, M., Luo, D., and Wang, L. Automatic scoring in a task of retelling stories for language learners. In *World Multi-Conference on Systemics, Cybernetics and Informatics: WMSCI* (2011).

[21] Chen, S., and Goodman, J. An empirical study of smoothing techniques for language modeling. *Technical Report TR-10-98, Computer Science Group, Harvard University* (1998).

[22] Cheng, C.-C., Sha, F., and Saul, L. A fast online algorithm for large margin training of continuous-density hidden markov models. In *Conference of the International Speech Communication Association (Interspeech)* (2009).

[23] Cheng, C.-C., Sha, F., and Saul, L. Online learning and acoustic feature adaptation in large-margin hidden markov models. *Journal of Selected Topics in Signal Processing 4(6)* (2010), 926–942.

[24] Chien, J.-T., and Chueh, C.-H. Joint acoustic and language modeling for speech recognition. *Speech Communication 52*, 3 (2010), 223 – 235.

[25] Cohen, J. The gale project: A description and an update. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2007), p. 237.

[26] Collins, M. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Empirical Methods in Natural Language Processing (EMNLP)* (2002).

[27] Davis, S. B., and Mermelstein, P. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics Speech and Signal Processing 28(4)* (1980), 357–366.

[28] Dempster, A. P., Laird, N. M., and Rubin, D. B. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B 39(1)* (1977), 1–38.

[29] Deng, L., Hinton, G., and Kingsbury, B. New types of deep neural network learning for speech recognition and related applications: An overview. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2013).

[30] Deoras, A., and Sarikaya, R. Deep belief network based semantic taggers for spoken language understanding. In *Conference of the International Speech Communication Association (Interspeech)* (2013).

[31] Dredze, M., Talukdar, P. P., and Crammer, K. Sequence learning from data with multiple labels. In *ECML/PKDD Workshop on Learning from Multi-Label Data (MLD)* (2009).

[32] Eide, E., and Gish, H. A parametric approach to vocal tract length normalization. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (1996), pp. 346–348.

[33] Fiscus, J., Garofolo, J., Le, A., Martin, A., greg Sanders, Przybocki, M., and Pallett, D. 2004 spring nist rich transcription (rt-04s) evaluation data. `http://www.ldc.upenn.edu/Catalog/catalogEntry.jsp?catalogId=LDC2007S12`, 2007.

[34] Fiscus, J., Garofolo, J., Przybocki, M., Fisher, W., and Pallett, D. 1997 English Broadcast News Speech (HUB4). Linguistic Data Consortium: LDC98S71, 1998.

[35] Gadde, V. R. R. Modeling word duration. In *International Conference on Spoken Language Processing (ICSLP)* (2000), pp. 601–604.

[36] Gales, M. Semi-tied covariance matrices for hidden markov models. *IEEE Transactions on Speech and Audio Processing 7(3)* (1999), 272–281.

[37] Gales, M., Watanabe, S., and Fosler-Lussier, E. Structured discriminative models for speech recognition. *IEEE Signal Processing Magazine* (2012), 70–81.

[38] Gales, M. J. F. Maximum likelihood linear transformations for hmm-based speech recognition. *Computer Speech and Language 12* (1998).

[39] Gauvain, J.-L., and hui Lee, C. Map estimation of continuous density hmm: theory and applications. In *Proceedings of the DARPA Speech and Natural Language Workshop* (1992), pp. 185–190.

[40] Gillick, D., Gillick, L., and Wegmann, S. Don't multiply lightly: Quantifying problems with the acoustic model assumptions in speech recognition. In *Automatic Speech Recognition and Understanding Workshop (ASRU)* (2011), pp. 71–76.

[41] Gillick, L., and Cox, S. Some statistical issues in the comparison of speech recognition algorithms. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (1989), vol. 1, pp. 532–535.

[42] Godfrey, J. J., Holliman, E. C., and McDaniel, J. Switchboard: Telephone speech corpus for research and development. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (1992), vol. 1, pp. 517–520.

[43] Gordon, J. K. Measuring the lexical semantics of picture description in aphasia. *Aphasiology* (2008), 22(7–8):839–852.

[44] Goronzy, S., Rapp, S., and Kompe, R. Generating non-native pronunciation variants for lexicon adaptation. *Speech Communication 42(1)* (2004), 109–123.

[45] Graff, D., Garofolo, J., Fiscus, J., Fisher, W., and Pallett, D. 1996 English Broadcast News Speech (HUB4). Linguistic Data Consortium: LDC97S44, 1997.

[46] Grune, and Stratton. *The Western aphasia battery.* New York, NY, 1982.

[47] Haeb-Umbach, R., and Ney, H. Linear discriminant analysis for improved large vocabulary continuous speech recognition. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (1992), vol. 1, pp. 13–16.

[48] Hahn, S., Lehnen, P., Raymond, C., and Ney, H. A comparison of various methods for concept tagging for spoken language understanding. In *International Conference of Language Resources Evaluation (LREC)* (2008), pp. 387–389.

[49] Hain, T. Implicit modelling of pronunciation variation in automatic speech recognition. *Speech Communication 46* (2005), 171–188.

[50] Hakkani-Tür, D., Béchet, F., Riccardi, G., and Tür, G. Beyond asr 1-best: Using word confusion networks in spoken language understanding. *Computer Speech and Language* (2006), 20:495–514.

[51] Hakkani-Tür, D., Vergyri, D., and Tür, G. Speech-based automated cognitive status assessment. In *Conference of the International Speech Communication Association (Interspeech)* (2010), pp. 258–261.

[52] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. The WEKA data mining software: An update. *SIGKDD Explorations 11*, 1 (2009).

[53] Heigold, G., Zweig, G., Li, X., and Nguyen, P. A flat direct model for speech recognition. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2009), pp. 3861–3864.

[54] Hermansky, H. Perceptual linear predictive (plp) analysis for speech. *Journal of Acoustic Society of America 87(4)* (1990), 1738–1752.

[55] Hinton, G., Deng, L., Yu, D., Dahl, G. E., rahman Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., and Kingbury, B. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine, IEEE 29(6)* (2012), 82–97.

[56] Hollingshead, K. Formalizing the use and characteristics of constraints in pipeline systems. In *Ph.D thesis, Oregon Health and Science University* (2010).

[57] Huang, X., Acero, A., and Hon, H. W. *Spoken Language Processing.* Prentice Hall PTR, New Jersey, USA, 2001.

[58] Humphries, J., and Woodland, P. C. Using accent-specific pronunciation modelling for improved large vocabulary continuous speech recognition. In *Proceedings Eurospeech* (1997).

[59] Hutchinson, B., and Droppo, J. Learning non-parametric models of pronunciation. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2011), pp. 4904–4907.

[60] Jaynes, E. T. Information theory and statistical mechanics. *Physical Review* (1957), 106(4):620–630.

[61] Jelinek, F. Continuous speech recognition by statistical methods. In *Proceedings IEEE* (1976), vol. 64(4), pp. 532–557.

[62] Jelinek, F. *Statistical Methods for Speech Recognition.* MIT Press, 1997.

[63] Jennequin, N., and Gauvain, J.-L. Modeling duration via lattice rescoring. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2007), pp. 641–644.

[64] Jin, R., and Ghahramani, Z. Learning with multiple labels. In *Neural Information Processing Systems (NIPS)* (2002).

[65] Juang, B.-H., Hou, W., and Lee, C.-H. Minimum classification error rate methods for speech recognition. *IEEE Transactions on Speech and Audio Processing 5* (1997), 257–265.

[66] Juang, B.-H., and Katagiri, S. Discriminative learning for minimum error classification. *IEEE Transactions on Signal Processing 40* (1992), 3043–3054.

[67] Jurafsky, D., Ward, W., Banping, Z., Herold, K., Xiuyang, Y., and Sen, Z. What kind of pronunciation variation is hard for triphones to model? In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2001), pp. 577–580.

[68] Jyothi, P., Fosler-Lussier, E., and Livescu, K. Discriminative training of WFST factors with application to pronunciation modeling. In *Conference of the International Speech Communication Association (Interspeech)* (2013).

[69] Kanthak, S., Ney, H., Riley, M., and Mohri, M. A comparison of two lvr search optimization techniques. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2002), pp. 1309–1312.

[70] Kantor, A. Pronunciation modeling for large vocabulary speech recognition. *PhD dissertation, University of Illinois, Urbana-Champaign* (2010).

[71] Kao, J. T., Zweig, G., and Nguyen, P. Discriminative duration modeling for speech recognition with segmental conditional random fields. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2011), pp. 4476–4479.

[72] Karanasou, P., Yvon, F., Lavergne, T., and Lamel, L. Discriminative training of a phoneme confusion model for a dynamic lexicon in asr. In *Conference of the International Speech Communication Association (Interspeech)* (2013).

[73] Kemper, S., LaBarge, E., Ferraro, F., Cheung, H., Cheung, H., and Storandt, M. On the preservation of syntax in alzheimer's disease. *Archives of Neurology* (1993), 50:81–86.

[74] Kingsbury, B., Soltau, H., Saon, G., Chu, S. M., Kuo, H.-K., Mangu, L., Ravuri, S. V., Morgan, N., and Janin, A. The IBM 2009 GALE Arabic speech transcription system. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2011), pp. 4672–4675.

[75] Kirchhoff, K., Bilmes, J., Das, S., Duta, N., Egan, M., Ji, G., He, F., Henderson, J., Liu, D., Noamany, M., Schone, P., Schwartz, R., and Vergyri, D. Novel approaches to arabic speech recognition: report from the 2002 johns-hopkins summer workshop. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2003), pp. 344–347.

[76] Kirchhoff, K., Bilmes, J., Henderson, J., Schwartz, R., Noamany, M., Schone, P., Ji, G., Das, S., Egan, M., He, F., Vergyri, D., Liu, D., and Duta, N. Novel speech recognition models for arabic. In *Johns-Hopkins University Summer Research Workshop* (2002).

[77] Korkman, M., Kirk, U., and Kemp, S. *NEPSY: A developmental neuropsychological assessment.* The Psychological Corporation, San Antonio, 1998.

[78] Krishnamurthy, A., and Childers, D. Two channel speech analysis. *IEEE Transactions on Acoustics Speech and Signal Processing 34(4)* (1986), 730–343.

[79] Kubo, Y., Hori, T., and Nakamura, A. Integrating deep neural networks into structured classification approach based on weighted finite-state transducers. In *Conference of the International Speech Communication Association (Interspeech)* (2012), pp. 4841–4844.

[80] Kubo, Y., Hori, T., and Nakamura, A. Large vocabulary continuous speech recognition based on wfst structured classifiers and deep bottleneck features. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2013), pp. 7629–7633.

[81] Kubo, Y., Watanabe, S., Hori, T., and Nakamura, A. Structural classification methods based on weighted finite-state transducers for automatic speech recognition. *IEEE Transactions on Audio Speech and Language Processing 52* (2012), 2240–2251.

[82] Kubo, Y., Watanabe, S., and Nakamura, A. Decoding network optimization using minimum transition error training. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2012), pp. 4197–4200.

[83] Kuo, H.-K. J., Arisoy, E., Mangu, L., and Saon, G. Minimum bayes risk discriminative language models for arabic speech recognition. In *Automatic Speech Recognition and Understanding Workshop (ASRU)* (2011), pp. 208–213.

[84] Kuo, H.-K. J., Fosler-Lussier, E., Jiang, H., and Lee, C.-H. Discriminative training of language models for speech recognition. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2002), pp. 325–328.

[85] Kuo, H.-K. J., and Gao, Y. Maximum entropy direct models for speech recognition. *IEEE Transactions on Audio Speech and Language Processing* (2006), 873–881.

[86] Kuo, H.-K. J., Kingsbury, B., and Zweig, G. Discriminative training of decoding graphs for large vocabulary continuous speech recognition. In *IEEE International Conference Acoustic, Speech and Signal Processing (ICASSP)* (2007), pp. 45–48.

[87] Kurata, G., Itoh, N., Nishimura, M., Sethy, A., and Ramabhadran, B. Leveraging word confusion networks for named entity modeling and detection from conversational telephone speech. *Speech Communication 54*, 3 (2012), 491–502.

[88] Labov, W. *Language in the inner city: Studies in Black English Vernacular*. University of Pennsylvania Press, 1972.

[89] Labov, W. *Principles of linguistic change: Social factors*. Wiley-Blackwell, Malden, MA, 2001.

[90] Labov, W. *The social stratification of English in New York City*, 2nd ed. Cambridge University Press, Cambridge, 2006.

[91] Lafferty, J., McCallum, A., and Pereira, F. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *International Conference on Machine Learning (ICML)* (2001).

[92] Lamel, L., Messaoudi, A., and Gauvain, J.-L. Improved acoustic modeling for transcribing arabic broadcast data. In *Conference of the International Speech Communication Association (Interspeech)* (2007), pp. 2077–2080.

[93] Lecorvé, G., Dines, J., Hain, T., and Motlicek, P. Supervised and unsupervised web-based language model domain adaptation. In *Conference of the International Speech Communication Association (Interspeech)* (2012).

[94] Lee, L., and Rose, R. A frequency warping approach to speaker normalization. *IEEE Transactions on Speech and Audio Processing 6(1)* (1998), 49–60.

[95] Legetter, C. J., and Woodland, P. C. Maximum likelihood linear regression for speaker adaptation of continuous density hmm. *Computer Speech and Language* (1995), 9:171–185.

[96] Lehr, M., Prud'hommeaux, E., Shafran, I., and Roark, B. Fully automated neuropsychological assessment for detecting mild cognitive impairment. In *Conference of the International Speech Communication Association (Interspeech)* (2012).

[97] Lehr, M., and Shafran, I. Discriminatively estimated joint acoustic, duration and language model for speech recognition. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2010).

[98] Lehr, M., and Shafran, I. Learning a discriminative weighted finite-state transducer for speech recognition. *IEEE Transactions on Audio Speech and Language Processing* (2011), 1360–1367.

[99] Lehr, M., Shafran, I., Prud'hommeaux, E., and Roark, B. Discriminative joint modeling of lexical variation and acoustic confusion for automated narrative retelling assessment. In *North American Chapter of Association for Computational Linguistics (NAACL)* (2013).

[100] Levenshtein, V. I. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady 10(8)* (1966), 707–710.

[101] Levinson, S. E. Continuously variable duration hidden Markov models for automatic speech recognition. *Computer Speech and Language 1* (1986), 29–45.

[102] Levinson, S. E. Continuously variable duration hidden markov models for automatic speech recognition. *Computer Speech and Language. 1*, 1 (1986), 29–45.

[103] Lin, S.-S., and Yvon, F. Discriminative training of finite state decoding graphs. In *Conference of the International Speech Communication Association (Interspeech)* (2005), pp. 234–241.

[104] Livescu, K., and Glass, J. Lexical modeling of non-native speech for automatic speech recognition. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2000), pp. 1842–1845.

[105] Lord, C., Rutter, M., DiLavore, P., and Risi, S. *Autism Diagnostic Observation Schedule (ADOS).* Western Psychological Services, Los Angeles, 2002.

[106] Lu, L., Ghoshal, A., and Renals, S. Acoustic data-driven pronunciation lexicon for large vocabulary speech recognition. In *Automatic Speech Recognition and Understanding Workshop (ASRU)* (2013).

[107] Lyons, K., Kemper, S., LaBarge, E., Ferraro, F., Balota, D., and Storandt, M. Oral language and alzheimer's disease: A reduction in syntactic complexity. *Aging and Cognition* (1994), 1(4):271–281.

[108] Mamou, J., Carmel, D., and Hoory, R. Spoken document retrieval from call-center conversations. In *SIGIR conference on Research and development in information retrieval* (2006).

[109] Mangu, L., Brill, E., and Stolcke, A. Finding consensus in speech recognition: word error minimization and other applications of confusion networks. *Computer Speech and Language 14* (2000), 373–400.

[110] Matusov, E., Kanthak, S., and Ney, H. On the integration of speech recognition and statistical machine translation. In *Conference of the International Speech Communication Association (Interspeech)* (2005).

[111] McCallum, A. K. Mallet: A machine learning for language toolkit. `http://mallet.cs.umass.edu`, 2002.

[112] McGraw, I., Badr, I., and Glass, J. R. Learning lexicons from speech using a pronunciation mixture model. *IEEE Transactions on Audio Speech and Language Processing 21* (2013), 357–366.

[113] Mesnil, G., He, X., Deng, L., and Bengio, Y. Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding. In *Conference of the International Speech Communication Association (Interspeech)* (2013).

[114] Messaoudi, A., Gauvain, J.-L., and Lamel, L. Arabic broadcast news transcription using a one million word vocalized vocabulary. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2006).

[115] Mikolov, T. Statistical language models based on neural networks. In *Ph.D. thesis,Brno University of Technology* (2012).

[116] Mitleb, F. Some aspects of arabic duration. *Journal International Phonetic Association* (1992), 22:27–34.

[117] Mohri, M. Minimization of sequential transducers. *Lecture Notes in Computer Science 807* (1994).

[118] Mohri, M. Finite-State Transducers in Language and Speech Processing. *Computational Linguistics 23(2)* (1997).

[119] Mohri, M. Minimization algorithms for sequential transducers. *Theoretical Computer Science 234* (2000), 177–201.

[120] Mohri, M., Pereira, F., and Riley, M. Weighted automata in text and speech processing. In *ECAI Workshop* (1996).

[121] Mohri, M., Pereira, F., and Riley, M. The Design Principles of a Weighted Finite-State Transducer Library. *Theoretical Computer Science 231* (2000), 17–32.

[122] Mohri, M., Pereira, F., and Riley, M. Weighted finite-state transducers in speech recognition. *Computer Speech and Language 16* (2002), 69–88.

[123] Mohri, M., Pereira, F., and Riley, M. Speech recognition with weighted finite-state transducers. *Handbook on Speech Processing and Speech Communication, Part E: Speech recognition, Springer-Verlag, Germany* (2008).

[124] Mohri, M., and Riley, M. Integrated context-dependent networks in very large vocabulary speech recognition. In *Proceedings of the 6th European Conference on Speech Communication and Technology* (1999).

[125] Mohri, M., and Riley, M. Network Optimizations for Large Vocabulary Speech Recognition. *Speech Communication 28(1):1–12* (1999).

[126] Mohri, M., Riley, M., Hindle, D., Ljolje, A., and Pereira, F. Full expansion of context-dependent networks in large vocabulary speech recognition. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (1998).

[127] Mori, R. D., Béchet, F., Hakkani-Tür, D., McTear, M., Riccardi, G., and Tür, G. Spoken language understanding. *IEEE Signal Processing Magazine Special Issue on Spoken Language Technologies* (2008), 25(3):50–58.

[128] Morris, J. The clinical dementia rating (CDR): Current version and scoring rules. *Neurology 43* (1993), 2412–2414.

[129] Motlicek, P., Garner, P. N., Kim, N., and Cho, J. Accent adaptation using subspace gaussian mixture models. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2013), pp. 7170–7174.

[130] Nallasamy, U., Metze, F., and Schultz, T. Enhanced polyphone decision tree adaptation for accented speech recognition. In *Conference of the International Speech Communication Association (Interspeech)* (2012).

[131] Nguyen, P., Heigold, G., and Zweig, G. Speech Recognition with Flat Direct Models. *IEEE Journal of Selected Topics in Signal Processing* (2010).

[132] NIST. Speech recognition scoring toolkit. `http://www.itl.nist.gov/iad/mig//tools/`, 2007.

[133] Nordlund, A., Rolstad, S., Hellstrom, P., Sjogren, M., Hansen, S., and Wallin, A. The Goteborg MCI study: Mild cognitive impairment is a heterogeneous condition. *Journal of Neurology, Neurosurgery and Psychiatry 76*, 11 (2005), 1485–1490.

[134] Oba, T., Hori, T., and Nakamura, A. A comparative study on methods of weighted language model training for reranking lvcsr n-best hypotheses. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2010), pp. 5126–5129.

[135] Parthasarathi, S. H. K., Chang, S.-Y., Cohen, J., Morgan, N., and Wegmann, S. The blame game in meeting room asr: an analysis of feature versus model errors in noisy and mismatched conditions. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2013), pp. 6758–6762.

[136] Pereira, F., and Riley, M. *Finite State Language Processing.* Chapter Speech Recognition by Composition of Weighted Finite Automata. The MIT Press, 1997.

[137] Petersen, R., Smith, G., Waring, S., Ivnik, R., Tangalos, E., and Kokmen, E. Mild cognitive impairment: Clinical characterizations and outcomes. *Archives of Neurology 56* (1999), 303–308.

[138] Pietra, S. A. D., Pietra, V. J. D., and Lafferty, J. Inducing features of random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (1997), 19(4).

[139] Povey, D. Discriminative training for large vocabulary speech recognition. In *Ph.D dissertation, Cambridge University* (2004).

[140] Povey, D., Hannemann, M., Boulianne, G., Burget, L., Ghoshal, A., Janda, M., Karafiat, M., Kombrink, S., Motlicek, P., Qian, Y., Riedhammer, K., Vesely, K., and Vu, N. T. Generating exact lattices in the wfst frameworks. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2012), pp. 4213–4216.

[141] Povey, D., Kanevsky, D., Kingsbury, B., Ramabhadran, B., Saon, G., and Visweswariah, K. Boosted mmi for model and feature space discriminative training. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2008).

[142] Povey, D., and Saon, G. Feature and model space speaker adaptation with full covariance gaussians. In *Conference of the International Speech Communication Association (Interspeech)* (2006).

[143] Povey, D., and Woodland, P. Minimum phone error and i-smoothing for improved discriminative training. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2002), pp. 105–108.

[144] Printz, H., and Olsen, P. Theory and practice of acoustic confusability. *Computer Speech and Language 16* (2002), 131–164.

[145] Prud'hommeaux, E., and Roark, B. Alignment of spoken narratives for automated neuropsychological assessment. In *Automatic Speech Recognition and Understanding Workshop (ASRU)* (2011).

[146] Prud'hommeaux, E., and Roark, B. Graph-based alignment of narratives for automated neuropsychological assessment. In *Proceedings of the NAACL 2012 Workshop on Biomedical Natural Language Processing (BioNLP)* (2012).

[147] Pylkkönen, J., and Kurimo, M. Duration modeling techniques for continuous speech recognition. In *International Conference on Spoken Language Processing (ICSLP)* (2004).

[148] Rabiner, L. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE* (1989), p. 77(2).

[149] Rabiner, L. *Fundamentals of Speech Recognition.* Prentice Hall PTR, New Jersey, USA, 1993.

[150] Rastrow, A., Sethy, A., and Ramabhadran, B. Constrained discriminative training of n-gram language models. In *Automatic Speech Recognition and Understanding Workshop (ASRU)* (2009), pp. 311–316.

[151] Ratinov, L., and Roth, D. Design challenges and misconceptions in named entity recognition. In *Computational Natural Language Learning (CoNLL)* (2009), pp. 147–155.

[152] Ratnaparkhi, A. Maximum entropy models for natural language ambiguity resolution. *PhD thesis, University of Pennsylvania* (1998).

[153] Rickford, J. R. Phonological and grammatical features of African American Vernaculary English (AAVE). In *African American Vernacular English*, J. R. Rickford, Ed. Blackwell, Malden, MA, 1999, pp. 3–14.

[154] Ritchie, K., and Touchon, J. Mild cognitive impairment: Conceptual basis and current nosological status. *Lancet 355* (2000), 225–228.

[155] Roark, B., Mitchell, M., Hosom, J.-P., Hollingshead, K., and Kaye, J. Spoken language derived measures for detecting mild cognitive impairment. *IEEE Transactions on Audio, Speech and Language Processing 19(7)* (1980).

[156] Roark, B., Saraçlar, M., and Riley, M. Discriminative n-gram language modeling. *Computer Speech and Language 21* (2007), 373–392.

[157] Russell, M., and Moore, R. Explicit modelling of state occupancy in hidden markov models for automatic speech recognition. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (1985).

[158] Russell, M. J., and Cook, A. E. Experimental evaluation of duration modelling techniques for automatic speech recognition. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (1987), pp. 2376–2379.

[159] Ryding, K. C. *A Reference Grammar of Modern Standard Arabic.* Cambridge University Press, 2005.

[160] Sainath, T. N., rahman Mohamed, A., Kingsbury, B., and Ramabhadran, B. Deep convolutional neural networks for lvcsr. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2013).

[161] Saraçlar, M., Nock, H., and Khudanpur, S. Pronunciation modeling by sharing gaussian densities across phonetic models. *Computer Speech and Language 14* (2000), 137–160.

[162] Saraçlar, M., and Roark, B. Utterance classification with discriminative language modeling. *Speech Communication 48*, 3-4 (2006), 276–287.

[163] Sha, F., and Saul, L. Large margin hidden markov models for automatic speech recognition. In *Advances in Neural Information Processing Systems (NIPS)* (2007), pp. 1249–1256.

[164] Shafran, I., and Hall, K. Corrective models for speech recognition of inflected languages. In *Empirical Methods Natural Language Processing (EMNLP)* (2006), pp. 390–398.

[165] Shafran, I., Sproat, R., Yarmohammadi, M., and Roark, B. Efficient determinization of tagged word lattices using categorial and lexicographic semirings. In *Automatic Speech Recognition and Understanding Workshop (ASRU)* (2011).

[166] Soltau, H., Mangu, L., and Biadsy, F. From modern standard arabic to levantine asr: Leveraging gale for dialects. In *Automatic Speech Recognition and Understanding Workshop (ASRU)* (2011), pp. 266–271.

[167] Soltau, H., Saon, G., and Kingsbury, B. The ibm attila speech recognition toolkit. In *IEEE Workshop on Spoken Language Technology* (2010).

[168] Soltau, H., Saon, G., Kingsbury, B., H.-K.Kuo, Povey, D., and Emami, A. Advances in arabic speech transcription at IBM under DARPA GALE program. *IEEE Transactions on Audio Speech and Language Processing 17*, 5 (2009), 884–895.

[169] Sproat, R., Yarmohammadi, M., Shafran, I., and Roark, B. Applications of lexicographic semirings to problems in speech and language processing. *Computational Linguistics* (2014).

[170] Sproat, R., Zheng, F., Gu, L., Li, J., Zheng, Y., Su, Y., Zhou, H., Bramsen, P., Kirsch, D., Shafran, I., Tsakalidis, S., Starr, R., and Jurafsky, D. Dialectal chinese speech recognition: Final report. *Final report of CLSP Workshop* (2004).

[171] Storandt, M., and Hill, R. Very mild senile dementia of the Alzheimer's type: II Psychometric test performance. *Archives of Neurology 46* (1989), 383–386.

[172] Strik, H., and Cucchiarini, C. Modeling pronunciation variation for asr: A survey of the literature. *Speech Communication 29(2-4)* (1999), 225–246.

[173] Sundermeyer, M., ter, R. S., and Ney, H. Lstm neural networks for language modeling. In *Conference of the International Speech Communication Association (Interspeech)* (2012).

[174] Sutton, C., and McCallum, A. *An Introduction to Conditional Random Fields for Relational Learning.* Book chapter in Introduction to Statistical Relational Learning. MIT Press, 2006.

[175] Sutton, C., Rohanimanesh, K., and McCallum, A. Dynamic conditional random fields: Factorized probabilistic models for labeling and segmenting sequence data. In *International Conference on Machine Learning (ICML)* (2004).

[176] Sutton, C., Sindelar, M., and Mccallum, A. Reducing weight undertraining in structured discriminative learning. In *North American Chapter of Association of Computational Linguistics (NAACL)* (2006).

[177] Svec, J., Ircing, P., and Smidl, L. Semantic entity detection from multiple asr hypotheses within the wfst framework. In *Automatic Speech Recognition and Understanding Workshop (ASRU)* (2013).

[178] Tomokiyo, L., and Waibel, A. Adaptation methods for non-native speech. In *Proceedings of Multilinguality in Spoken Language Processing* (2001).

[179] Tomokiyo, L. M. Lexical and acoustic modeling of non-native speech in lvscr. In *Conference of the International Speech Communication Association (Interspeech)* (2000), pp. 346–349.

[180] Tsang, C. Y. H. The effects of age and visual stimuli on narrative production. In *Master of Science - MSc* (2009).

[181] Tsuruoka, Y., Tssujii, J., and Ananiadou, S. Stochastic gradient descent training for l1-regularized log-linear models with cumulative penalty. In *Association for Computational Linguistics (ACL)* (2009).

[182] Tür, G., Deoras, A., and Hakkani-Tür, D. Semantic parsing using word confusion networks with conditional random fields. In *Conference of the International Speech Communication Association (Interspeech)* (2013).

[183] Tur, G., and Mori, R. D. *Spoken Language Understanding - Systems for Extracting Semantic Information from Speech.* John Wiley and Sons, 2011.

[184] Valtchev, V., Odell, J. J., Woodland, P. C., and Young, S. J. Mmie training of large vocabulary recognition systems. *Speech Communication 22* (1997), 303–314.

[185] Vaseghi, S. V. State duration modeling in hidden markov models. *Signal Processing 41*, 1 (1995), 31–41.

[186] Vergyri, D., and Kirchhoff, K. Automatic diacritization of arabic for acoustic modeling in speech recognition. In *COLING, Computational Approaches to Arabic Script-based Languages* (2004), pp. 5542–5545.

[187] Vergyri, D., Lamel, L., and Gauvain, J.-L. Automatic speech recognition of multiple accented english data. In *Conference of the International Speech Communication Association (Interspeech)* (2010).

[188] Vinyals, O., Deng, L., Yu, D., and Acero, A. Discriminative pronunciation learning using phonetic decoder and minimum-classification-error criterion. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2009).

[189] Viterbi, A. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory 13*, 2 (1967), 260–269.

[190] Wang, Q.-S., and Zhou, J.-N. Retrieval and encoding of episodic memory in normal aging and patients with mild cognitive impairment. *Brain Research 924* (2002), 113–115.

[191] Wang, Z., Schultz, T., and Waibel, A. Comparison of acoustic model adaptation techniques on non-native speech. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2003), pp. 540–543.

[192] Watanabe, S., Hori, T., and Nakamura, A. Large vocabulary continuous speech recognition using wfst-based linear classifier for structured data. In *Conference of the International Speech Communication Association (Interspeech)* (2010), pp. 346–349.

[193] Wechsler, D. *Wechsler Memory Scale - Third Edition.* The Psychological Corporation, San Antonio, 1997.

[194] Wolfram, W. *A sociolinguistic description of Detroit Negro speech.* Center for Applied Linguistics, Arlington, VA, 1969.

[195] Xie, S., and Liu, Y. Using confusion networks for speech summarization. In *North American Chapter of Association for Computational Linguistics (NAACL)* (2010).

[196] Xiong, W., Evanini, K., Zechner, K., and Chen, L. Automated content scoring of spoken responses containing multiple parts with factual information. In *Proceedings of the ISCA SLaTE Workshop* (2013).

[197] Yaman, S., Deng, L., Yu, D., Wang, Y.-Y., and Acero, A. An integrative and discriminative technique for spoken utterance classification. *Transactions on Audio Speech and Language Processing* (2008), 16(6):1207–1214.

[198] Yao, K., Zweig, G., Hwang, M.-Y., Shi, Y., and Yu, D. Recurrent neural networks for language understanding. In *Conference of the International Speech Communication Association (Interspeech)* (2013).

[199] Young, S. J., Odell, J. J., and Woodland, P. C. Tree-based state tying for high accuracy acoustic modelling. In *Proceedings of Human Language Technology Workshop* (1994), pp. 307–312.

[200] Yu, S.-Z. Hidden semi-markov models. *Artificial Intelligence* (2010), 215–243.

[201] Yu, S.-Z., and Kobayashi, H. An efficient forward-backward algorithm for an explicit duration hidden markov model. *IEEE Signal Processing Letters* (2003), 11–14.

[202] Zheng, Y., Sproat, R., Gu, L., Shafran, I., Zhou, H., Su, Y., Jurafsky, D., Starr, R., and youn Yoon, S. Accent detection and speech recognition for shanghai-accented mandarin. In *Conference of the International Speech Communication Association (Interspeech)* (2005).

[203] Zweig, G., and Nguyen, P. A segmental crf approach to large vocabulary continuous speech recognition. In *Automatic Speech Recognition and Understanding Workshop (ASRU)* (2009), pp. 152–157.

[204] Zweig, G., and Nguyen, P. From flat direct models to segmental crf models. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2010).

[205] Zweig, G., and Nguyen, P. Scarf: A segmental conditional random field toolkit for speech recognition. In *Conference of the International Speech Communication Association (Interspeech)* (2010).

[206] Zweig, G., and Nguyen, P. Speech recognition with segmental conditional random fields: A summary of the jhu clsp summer workshop. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2011).