Modeling in Automatic Speech Recognition: Beyond Hidden Markov Models


Human Language Technology and Pattern Recognition
Lehrstuhl Informatik 6
Fakultät für Mathematik, Informatik und Naturwissenschaften
RWTH Aachen University
Introduction

Outline

Introduction

Statistical Sequence Classification

ASR Architectures: State-of-the-Art in Transition

ASR Modeling Approaches

Conclusions

References
Current situation in Automatic Speech Recognition (ASR):

- Decade brought \( \geq 50\% \) relative improvements in WER by introducing artificial neural networks to all levels of modeling.
- Traditional state-of-the-art challenged by novel “end-to-end” ASR architectures.
- Enabling factor: generic machine learning tools, developed for diverse and complex tasks.

ASR very challenging task - advantages from a method evaluation viewpoint:

- Provides clear performance objective.
- Strong state-of-the-art performance to compete against for new approaches.
- Various and diverse well-covered benchmarks.

Topics of interest:

- performance vs. system complexity
- variable length sequence alignment: beyond HMM
- primary training data and secondary knowledge sources
- reusability of inferred knowledge
Outline

Introduction

Statistical Sequence Classification

ASR Architectures: State-of-the-Art in Transition

ASR Modeling Approaches

Conclusions

References
**Statistical Sequence Classification**

**Sequence Classification**

**Tasks:**
- automatic speech recognition
- text image recognition
- machine translation

**Most general case:**
- input sequence: \( X := x_1 \ldots x_t \ldots x_T \)
- output sequence (of unknown length \( N \)): \( W := w_1 \ldots w_n \ldots w_N \)
- true distribution \( pr(W|X) \)
  (can be extremely complex!)
Statistical Sequence Classification

Statistical Approach Revisited

- **Performance measure**: judges quality of system output
- **Probabilistic models**: capture dependencies
  - elementary observations: Gaussian mixture, log-linear, SVM, NN, ...
  - strings: $n$-gram Markov chain, HMM, CRF, RNN, LSTM, attention/transformer, ...
- **Training criterion**: learns free parameters of models
  - linked to performance criterion?
  - complex optimization (efficiency!)
- **Bayes decision rule**: generates output word sequence
  - combinatorial problem (efficient algorithms: dynamic programming, beam search, $A^*$, ...)

Speech Recognition = Modeling + Statistics + Efficient Algorithms
Statistical Sequence Classification

**Sequence Decision Rule**

- **performance measure or loss function** $L[\tilde{w}_1^N, w_1^N]$ (e.g. edit distance for word/phoneme/character error computation) between true output sequence $\tilde{w}_1^N$ and hypothesized output sequence $w_1^N$.

- **Bayes decision rule** minimizes expected loss:

$$x_1^T \rightarrow r_L(x_1^T) := \arg \min_{w_1^N} \left\{ \sum_{\tilde{w}_1^N} \Pr(\tilde{w}_1^N|x_1^T) \cdot L[\tilde{w}_1^N, w_1^N] \right\}$$

- **Standard decision rule** uses sequence-level **zero-one loss**: minimizes sentence error

$$x_1^T \rightarrow r_{0-1}(x_1^T) := \arg \max_{w_1^N} \left\{ \Pr(w_1^N|x_1^T) \right\}$$

Since [Bahl & Jelinek 1983], this simplified Bayes decision rule is widely used for speech recognition, handwriting recognition, machine translation, ...

- Works well, as often both decision rules **coincide**.

This can be proven under certain conditions [Schlüter & Nussbaum 2012], e.g.:

$$L[w_1^N, \tilde{w}_1^N] \text{ is a metric, and } \max_{w_1^N} \Pr(w_1^N|x_1^T) \geq 0.5 \Rightarrow r_L(x_1^T) = r_{0-1}(x_1^T)$$
Statistical Sequence Classification

Statistical Approach: Integrated Decisions **End-to-End**

- **Audio Signal**
  - Acoustic Feature Extraction
  - Feature Encoding
  - Segmentation and Classification
    - Subword Hypotheses
  - Word Boundary Detection and Lexical Access
    - Word Hypotheses
    - Syntactic, Semantic, and Pragmatic Analysis
      - Sentence Hypotheses
      - Search, Integrating All Knowledge Sources
      - Recognized Word Sequence
  - Search Approach
- **Knowledge Sources**
  - Signal Analysis Model
  - Subword Models (Phoneme, Character, etc.)
  - Pronunciation Lexicon
  - Language Model
  - Search Approach
Statistical Sequence Classification

Statistical Approach: Training End-To-End

- acoustic feature extraction
- segmentation and classification
- subword hypotheses
- word boundary detection and lexical access
- syntactic, semantic, and pragmatic analysis
- sentence hypotheses
- search, integrating all knowledge sources
- spoken/transcribed word sequence
- feature encoding
- additional textual data
- knowledge sources
- phoneme set
- pronunciation lexicon
- language model
- search approach
- subword models (phoneme, character, etc.)
- signal analysis model

R. Schlüter: Modeling in ASR: Beyond (Standard) HMMs
Lehrstuhl Informatik 6 — Human Language Technology and Pattern Recognition
RWTH Aachen University
Sep. 16, 2019
Outline

Introduction

Statistical Sequence Classification

ASR Architectures: State-of-the-Art in Transition

ASR Modeling Approaches

Conclusions

References
Sequence Modeling

- Problem in Bayes decision rule:
  - true posterior distribution: unknown
  - separation into language model and acoustic model
    \[ p(w_1^N | x_1^T) = \frac{p(w_1^N) \cdot p(x_1^T | w_1^N)}{p(x_1^T)} \]

- Acoustic model \( p(x_1^T | w_1^N) \): links sentence hypothesis \( w_1^N \) to observation sequence \( x_1^T \).

- Problem in ASR: speaking rate variation \( \rightarrow \) non-linear time alignment

- Hidden Markov model:
  - linear chain of states \( s = 1, \ldots, S \)
  - transitions: forward, loop and skip
  - emissions: Gaussian mixture distributions (originally)

- Acoustic model using hidden state sequences \( s_1^T \):
  \[ p(x_1^T | w_1^N) = \sum_{s_1^T} p(x_1^T, s_1^T | w_1^N) = \sum_{s_1^T} \prod_{t} \left[ p(s_t | s_{t-1}, w_1^N) \cdot p(x_t | s_t, w_1^N) \right] \]
ASR Architectures: State-of-the-Art in Transition

ASR Architecture: Statistical Approach

Speech Input

Samples $s_1...s_M$

Feature Extraction

Feature Vectors $x_1...x_T$

Global Search Process:

$$\text{maximize} \quad p(x_1...x_T | w_1...w_N)$$

over $w_1...w_N$

$$p(w_1...w_N) \cdot p(x_1...x_T | w_1...w_N)$$

Recognized Word Sequence

$$\{w_1...w_N\}_{opt}$$

Statistical Approach to Automatic Speech Recognition (ASR)
[Bahl & Jelinek 1983]
ASR Architectures: State-of-the-Art in Transition

ASR Architecture: Search

Speech Input → Samples $s_1 ... s_M$ → Feature Extraction → Feature Vectors $x_1 ... x_T$ → Global Search Process:

\[
\text{maximize } p(w_1 ... w_N) \cdot p(x_1 ... x_T | w_1 ... w_N)
\]

over $w_1 ... w_N$

Recognized Word Sequence $\{w_1 ... w_N\}_{\text{opt}}$

search approaches:

- A*/stack decoder [Jelinek & Bahl+ 1975]
- dynamic programming lexical prefix tree beam search [Ney & Haeb-Umbach+ 1992]
- WFST decoding [Mohri & Riley 1999]
- RNN LM one pass search [Hori & Kubo+ 2014]
ASR Architectures: State-of-the-Art in Transition

ASR Architecture: Neural Networks

neural acoustic modeling:
- hybrid HMM [Bourlard & Morgan 1993]
  → large vocabulary [Seide & Li+ 2011]
  → RNNs [Robinson 1994]
- deep generative modeling [McDermott 2018]
ASR Architecture: Neural Networks

neural feature transformation:
- tandem [Hermansky & Ellis+ 2000]
- bottleneck [Grézl & Karafiát+ 2007]
  earlier introduced as non-linear LDA
  [Fontaine & Ris+ 1997]
ASR Architectures: Neural Networks

Global Search Process:

\[
\text{maximize } p(w_1...w_N) \cdot p(s_1...s_M | w_1...w_N)
\]

over \(w_1...w_N\)

Recognized Word Sequence

\(\{w_1...w_N\}_{\text{opt}}\)

integrated learning of acoustic model and feature extraction

- single channel [Palaz & Collobert\textsuperscript{+} 2013]
  [Tüske & Golik\textsuperscript{+} 2014]
  [Golik & Tüske\textsuperscript{+} 2015a]

- multichannel [Sainath & Weiss\textsuperscript{+} 2015]
ASR Architectures: State-of-the-Art in Transition

**ASR Architecture: Neural Networks**

- **Speech Input**
- **Samples** $s_1 \ldots s_M$
- **Feature Extraction**
- **Feature Vectors** $x_1 \ldots x_T$
- **Global Search Process:**
  - maximize
  - $p(w_1 \ldots w_N) \cdot p(x_1 \ldots x_T | w_1 \ldots w_N)$
  - over $w_1 \ldots w_N$
- **Acoustic Model**
- **Language Model**
- **Recognized Word Sequence** $\{w_1 \ldots w_N\}_{opt}$

**neural language modeling:**
- feed-forward (FF) [Schwenk 2007]
- very long context FF [Tüske & Irie$^+$ 2016]
- recurrent [Mikolov & Karafiát$^+$ 2010]
- LSTM [Sundermeyer & Schlüter$^+$ 2012]
- Transformer [Liu & Saleh$^+$ 2018]
ASR Architectures: State-of-the-Art in Transition

**ASR Architecture: Novel Approaches**

- **Speech Input**
- **Feature Extraction**
- **Feature Vectors**

**Global Search Process:**

\[
p(x_1 \ldots x_T, w_1 \ldots w_N)
\]

maximize

over \( w_1 \ldots w_N \)

**Recognized Word Sequence**

\( \{w_1 \ldots w_N\}_{\text{opt}} \)

**Integrated/End2End Model**

integrated NN modeling and search:

- connectionist temporal classification (CTC) [Graves & Fernández+ 2006]
- encoder-attention-decoder approach [Bahdanau & Chorowski+ 2015] [Chan & Jaitly+ 2015]
- transformer [Zhou & Dong+ 2018]

- segmental/inverted HMM [Lu & Kong+ 2016] [Doetsch & Hegselmann+ 2016]
Outline

Introduction

Statistical Sequence Classification

ASR Architectures: State-of-the-Art in Transition

ASR Modeling Approaches

- Generative Modeling
- Time-Synchronous Discriminative Modeling
- Label-Synchronous Discriminative Modeling

Results & Analysis

Conclusions

References
Outline

Introduction

Statistical Sequence Classification

ASR Architectures: State-of-the-Art in Transition

ASR Modeling Approaches

Generative Modeling

Time-Synchronous Discriminative Modeling

Label-Synchronous Discriminative Modeling

Results & Analysis

Conclusions

References
Neural Network Acoustic Modeling within Standard HMM Approach

- Decomposition within **Bayes decision rule** [Bahl & Jelinek+ 1983]:

\[
\arg\max_{w_1^N} p(w_1^N | x_1^T) = \arg\max_{w_1^N} p(w_1^N) \cdot p(x_1^T | w_1^N)
\]

- Decomposition of first order HMM:

\[
p(x_1^T | w_1^N) = \sum_{s_1^T} \prod_{t=1}^{T} p(x_t | s_t, w_1^N) \cdot p(s_t | s_{t-1}, w_1^N)
\]

- Emission probability distribution using **Gaussian mixture**:
  - Gaussian mixture distribution:

\[
p(x_t | s_t, w_1^N) = \sum_l c_{s_t,l} \mathcal{N}(x_t | \mu_{s_t,l}, \Sigma_{sl})
\]

  - (state) posterior level:
    + Gaussian w/pooled covariance equivalent to log-linear model with linear features
    + Gaussian mixture equivalent to log-linear mixture model

- Possibilities to introduce neural network modeling while keeping the **HMM alignment** process?
Tandem [Hermansky & Ellis+ 2000]

- **Idea**: use (properly transformed) ANN outputs to augment acoustic feature set
- First ANN-approach to considerably improve LVCSR on top of Gaussian-mixture HMMs
- **Approach**:
  - train phone-classifier ANN, use its output, or the output of intermediate/hidden layers as (additional) features for Gaussian mixture HMMs,
  - variant: **bottleneck** features [Grézl & Karafiát+ 2007], earlier introduced as non-linear discriminant analysis [Fontaine & Ris+ 1997]
  - usually requires less labels for NN training, than hybrid DNN/HMM approach.
  - Typically, some post-processing is applied to the neural network output: log, decorrelation and dimension reduction with PCA, concatenation with basic acoustic feature set (e.g. MFCC).
- **Advantages**:
  - all techniques from Gaussian mixture HMM modeling can be used, in particular speaker adaptation and discriminative (sequence) training
  - cross-/multi-lingual training data exploitable [Stolcke & Grézl+ 2006, Tüske & Pinto+ 2013]
  - bootstrapping on minimal amounts of target task training data [Golik & Tüske+ 2015b]
- **Disadvantage**: training usually twofold and thus inconsistent - two models required: tandem DNN and Gaussian mixture or hybrid HMM (yet: fine-tuning end-to-end [Tüske & Golik+ 2015])
ASR Modeling Approaches

Generative Modeling

Hybrid HMM: modeling the acoustic vector $x_t$ [Bourlard & Morgan 1993]

- Phonetic labels (allophones, sub-phones): $(s, w_1^N) \rightarrow \phi = \phi_{s, w_1^N}$
- Typical approach: decision trees, e.g. classification and regression trees (CART):
- **Hidden Markov model (HMM)** emission probability density:

  $$p(x_t | s, w_1^N) = p(x_t | \phi_{s, w_1^N})$$

- **Idea**: rewrite the emission probability for label $\phi$ and acoustic vector $x_t$:

  $$p(x_t | \phi) = \frac{p(x_t) \cdot p(\phi | x_t)}{p(\phi)}$$

  - prior probability $p(\phi)$: estimated as relative frequencies (alternatively averaged NN posteriors)
  - for recognition purposes: term $p(x_t)$ can be dropped

- **Result**: rather than the phone label emission distribution $p(x_t | \phi)$,
model the phone label posterior probability by an NN:

  $$x_t \rightarrow p(\phi | x_t)$$

- **Justification**:
  - easier learning problem: $O(10^4)$ labels $\phi$ vs. vectors $x_t \in \mathbb{R}^{D=40}$
  - well-known result in pattern recognition (but ignored in ASR!)
Hybrid vs. Tandem and Beyond

- **Tandem**:
  - provides high-level, robust and crosslingually generalizing features.
  - known techniques from GMHMM apply (speaker adaptation, discriminative training, etc.)
- **Hybrid**:
  - single model, consistent training.
- **Discussion**:
  - Are they so much different?
  - Relation between Gaussian and log-linear modeling: with pooled covariance only linear features are used: similarity to (unnormalized) softmax layer
  - joint tandem DNN and Gaussian mixture HMM can be viewed as hybrid DNN/HMM: specific topology (combination of linear, sum-/max-pooling and softmax [Tüske & Golik\textsuperscript{+} 2015])
  - Tandem & Gaussian mixture HMM can be trained jointly [Tüske & Tahir\textsuperscript{+} 2015] → hybrid
- **Experiments**:
  - [Tüske & Sundermeyer\textsuperscript{+} 2012, Tüske & Golik\textsuperscript{+} 2015], ...: similar results for tandem & hybrid
- **Towards deep generative modeling**:
  - combine deep tandem features with deep density models [McDermott 2018]
  - what can be learnt from deep generative modeling in TTS? e.g. [van den Oord\textsuperscript{+} 2016]
    → utilize for unsupervised training [Tjandra\textsuperscript{+} 2017]
    → issue: speaker dependence/adaptation [Tjandra\textsuperscript{+} 2018]
Outline

Introduction

Statistical Sequence Classification

ASR Architectures: State-of-the-Art in Transition

ASR Modeling Approaches
  Generative Modeling
  Time-Synchronous Discriminative Modeling
  Label-Synchronous Discriminative Modeling

Results & Analysis

Conclusions

References
Discriminative Modeling: from Labels to Frames

• Basically, Bayes decision rule requires modeling of label ([sub]word, ...) **posterior probabilities**

• **Idea**: redefine label sequence on time frame level:
  
  \[ p(c_1^N|x_1^T) \leftarrow p(\overline{c}_1^T|x_1^T) \]

  with unique mapping from frame-wise to original label sequence \( G : \overline{\mathcal{V}}^* \rightarrow \mathcal{V}^* \), \( c_1^N = G(\overline{c}_1^T) \)

• **Alignment**: marginalize over label boundaries on time frame level
  
  \[
  p(c_1^N|x_1^T) = \sum_{\overline{c}_1^T} p(\overline{c}_1^T, c_1^N|x_1^T) = \sum_{\overline{c}_1^T} p(c_1^N|\overline{c}_1^T)p(\overline{c}_1^T|x_1^T) = \sum_{\overline{c}_1^T: G(\overline{c}_1^T) = c_1^N} p(\overline{c}_1^T|x_1^T)
  \]

  with deterministic frame to label mapping:
  
  \[
  p(c_1^N|\overline{c}_1^T) = \begin{cases} 
  1 & \text{iff } G(\overline{c}_1^T) = c_1^N \\
  0 & \text{otherwise}
  \end{cases}
  \]

• decompose frame-level posterior \( p(\overline{c}_1^T|x_1^T) \) into product over time frames and assume
  
  – **label independence**: connectionist temporal classification (CTC) [Graves & Fernández+ 2006]
  
Connectionist Temporal Classific. (CTC) [Graves & Fernández+ 2006]

- Mapping from frame to label level: extend label set by **blank** symbol “ε”: $\bar{\mathcal{V}} = \mathcal{V} \cup \{\epsilon\}$
  - blank symbol may be inserted at any point without any effect
  - Adjacent identical labels need to be separated by blank, e.g.:
    $$G(\epsilon s s p \epsilon e e e \epsilon e c \epsilon e h h \epsilon) = G(s p e \epsilon e c c h h \epsilon) = \text{speech}$$

- Assume **statistical independence** of label sequence
  $$p(\bar{c}_1^T | x_1^T) = \prod_{t=1}^{T} p_t(\bar{c}_t | x_1^T)$$

- Related to **hybrid HMM**:
  - two-states per label, 2\textsuperscript{nd} state globally shared for all labels
  - w/o division by state prior
- During training, sum over alignments can be computed with forward-backward algorithm, like the expectation step in the EM algorithm for HMM training.
CTC: Search/Decoding
• w/o language model:
  – leads to independent **frame-by-frame decisions**: trivial
  – with extremely large training set even possible on word level [Soltau & Liao+ 2016]
  – result of statistical independence assumption on label level

\[
\begin{align*}
\text{argmax } \bar{p}(\bar{c}_T^T | x_1^T) &= \text{argmax } \prod_{t=1}^{T} p_t(\bar{c}_t | x_1^T) \\
&= (\text{argmax } p_{t=1}(\bar{c} | x_1^T), \ldots, \text{argmax } p_{t=T}(\bar{c} | x_1^T))
\end{align*}
\]

– equivalent to frame-level **word error loss**-based Bayes decision rule [Wessel & Schlüter+ 2001]:

\[
\begin{align*}
\text{argmin } \sum_{\bar{c}_1^T} \bar{p}(\bar{c}_1^T | x_1^T) \cdot C(\bar{c}_1^T, \bar{c}_1^T) &= \text{argmin } \sum_{\bar{c}_1^T} \bar{p}(\bar{c}_1^T | x_1^T) \cdot \sum_{t=1}^{T} (1 - \delta_{\bar{c}_t, \bar{c}_t}) \\
&= \text{argmax } \sum_{\bar{c}_1^T} \bar{p}_t(\bar{c}_t | x_1^T) \\
&= (\text{argmax } p_{t=1}(\bar{c} | x_1^T), \ldots, \text{argmax } p_{t=T}(\bar{c} | x_1^T))
\end{align*}
\]

• with language model: CTC used within **hybrid HMM** approach [Miao & Gowayyed+ 2015]
  \(\rightarrow\) standard decoding/search approach (here using WFST)
Recurrent Neural Aligner

- Recurrent neural aligner (RNA) [Sak & Shannon+ 2017]:
  - similar to CTC, but
  - avoids label independence assumption:

\[
p(\bar{c}_1^T | x_1^T) = \prod_{t=1}^{T} p_t(\bar{c}_t | \bar{c}_{t-1}^t, x_1^T)
\]

RNN-Transducer

- RNN-transducer [Graves 2012]:
  - similar to RNA, but does only forward to next frame, if blank label is hypothesized

Search/decoding:
- pursues tree of all label sequences
- fixed-size beam pruning

[Diagram showing transition probabilities in an RNN-Transducer model]
Outline

Introduction

Statistical Sequence Classification

ASR Architectures: State-of-the-Art in Transition

ASR Modeling Approaches
  Generative Modeling
  Time-Synchronous Discriminative Modeling
  Label-Synchronous Discriminative Modeling

Results & Analysis

Conclusions

References
Directly Hypothesizing Label by Label

- Decompose label sequence posterior probability on a label-by-label level:

\[ p(c_1^N|x_1^T) = \prod_{n=1}^{N} p(c_n|c_1^{n-1}, x_1^T) \]

- modeling of **unlimited label context**: can be done by RNN structures (cf. RNN LMs)
- However: how do position-wise label posteriors access/align to corresponding input intervals?
  - encoder/decoder attention and transformer: **attention** in time
  - segmental/inverse HMM: explicit **label boundary** modeling
  - 2D LSTM approach: temporal averaging/**not at all**

- **Advantage**: integrated model, fully exploits interaction between input and label sequence
- **Disadvantage**: training domain integration - domain transfer?
(Non-Latent) Attention-based Encoder/Decoder
[Bahdanau & Chorowski\textsuperscript{+} 2015, Chan & Jaitly\textsuperscript{+} 2015]

- **decoder input** for $n$-th label determined by attention process depending on label context:

$$p(c_n^N | x_1^T) = \prod_{n=1}^{N} p(c_n | c_{n-1}^{n-1}, x_1^T)$$

$$= \prod_{n=1}^{N} p(c_n | c_{n-1}^{n-1}, \xi(c_{n-1}^{n-1}, x_1^T))$$

- **soft attention**: weighted average over encoder output of entire utterance:

$$\xi(c_{n-1}^{n-1}, x_1^T) = \sum_{t=1}^{T} \alpha_t(c_{n-1}^{n-1}, x_1^T) \cdot x_t$$

- observations/problems:
  - attention determined by context, does not consider current label
    - attention intervals are not revised after label hypothesization: no recombination
  - left-right asymmetry [Mimura & Sakai\textsuperscript{+} 2018]
  - competitive performance reported with sufficiently large training sets
Attention Visualization

- attention weights: peaky, incomplete coverage of encoder output (depending on downsampling) ⇒ encoder needs to **temporally compress** information
- informal experiments: attention trained on top of fixed hybrid encoder does not perform
- strong **interaction** of attention and encoder.

- **transformer**: replaces RNN decoder by self-attention, cascades attention [Zhou & Dong+ 2018] → attention issues w.r.t. alignment apply similarly
Segmental/Inverted HMM, Posterior Attention

[Lu & Kong\textsuperscript{+} 2016, Doetsch & Hegselmann\textsuperscript{+} 2016]

- Idea: label sequence posterior with latent alignment and Markov assumptions:

\[
p(c_1^N|x_T^1) = \sum_{t_1^N} p(c_1^N, t_1^N|x_T^1) = \sum_{t_1^N} \prod_{n=1}^{N} p(c_n, t_n|c_1^{n-1}, t_1^{n-1}, x_T^1)
\]

\[
= \sum_{t_1^N} \prod_{n=1}^{N} p(c_n, t_n|c_1^{n-1}, t_{n-1}, x_T^1) \quad \text{1st-order Markov joint model (A)}
\]

\[
= \sum_{t_1^N} \prod_{n=1}^{N} p(c_n|c_1^{n-1}, t_{n-1}, x_T^1) \cdot p(t_n|c_1^{n-1}, c_n, t_{n-1}, x_T^1) \quad \text{target label-dependent (B)}
\]

alignment distribution

\[
= \sum_{t_1^N} \prod_{n=1}^{N} p(c_n|c_1^{n-1}, t_{n-1}, t_n, x_T^1) \cdot p(t_n|c_1^{n-1}, t_{n-1}, x_T^1) \quad \text{target label-independent}
\]

- marginalization of alignment efficiently performed using dynamic programming
- ongoing work: modeling of decoder model distributions for label and alignment
Inverting HMM Alignment

Exemplary topologies for standard and inverted HMM alignment:

- Standard HMM trellis.
- Inverted HMM trellis.

- In MT introduced as **neural HMM** [Wang & Zhu⁷ 2018]: results similar to attention.
- Introduced as latent generalization of attention: **posterior attention** [Shankar & Sarawagi 2019]: consistently better results reported (BLEU).
2-dim. LSTM [Bahar & Zeyer+ 2019]

- **Idea**: use 2D-LSTM for both label propagation and alignment/input coverage
- Keep label-synchronous derivation, avoid explicit temporal alignment:

\[
p(c^N_1 | x^T_1) = \prod_{n=1}^{N} p(c_n | c^{n-1}_1, x^T_1)
\]

- **Advantage**: exploits 2-dim. structure of input-output relation, completely avoids alignment.
- **Disadvantage**: as for soft attention no monotonicity or locality constraints.

2D-LSTM architecture avoiding attention.
Outline

Introduction

Statistical Sequence Classification

ASR Architectures: State-of-the-Art in Transition

ASR Modeling Approaches
  Generative Modeling
  Time-Synchronous Discriminative Modeling
  Label-Synchronous Discriminative Modeling

Results & Analysis

Conclusions

References
# Current Results for new Architectures

- Training: LibriSpeech 1000h, Switchboard 300h
- CDP: context-dependent phonemes (generalized triphone states)
- BPE: subwords based on byte-pair encoding, 1000 merges

<table>
<thead>
<tr>
<th>acoustic model approach</th>
<th>language model approach</th>
<th>Switchboard Hub5 '00 WER [%]</th>
<th>LibriSpeech test-other</th>
</tr>
</thead>
<tbody>
<tr>
<td>inv. HMM</td>
<td>CDP</td>
<td>4-gram</td>
<td>13.0</td>
</tr>
<tr>
<td>2D-LSTM</td>
<td>BPE</td>
<td>none</td>
<td>10.6</td>
</tr>
<tr>
<td>attention</td>
<td></td>
<td></td>
<td>9.9</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>9.9</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>9.9</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10.3</td>
</tr>
<tr>
<td>hybrid</td>
<td>CDP</td>
<td>4-gram</td>
<td>8.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8.8</td>
</tr>
</tbody>
</table>

(Librispeech results, hybrid: [Lüscher & Beck^+ 2019], attention: unpublished 2019)
(Switchboard results, hybrid: [Kitza & Schlüter^+ 2019], attention: unpublished 2019)
(Inv. HMM results: [Beck & Hannemann^+ 2018], work in progress)
(2D-LSTM results: work in progress following [Bahar & Zeyer^+ 2019])
### Performance as a Function of Training Data Amount

GMM/HMM vs. hybrid BLSTM/HMM vs. BLSTM/attention: Comparison on LibriSpeech, dev-clean

<table>
<thead>
<tr>
<th>amount training data [h]</th>
<th>WER [%] dev-clean</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GMM AM + 4g LM</td>
<td>BLSTM AM + LSTM LM</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>13.0</td>
<td>9.2</td>
<td>&gt;100</td>
</tr>
<tr>
<td>50</td>
<td>23.0</td>
<td>10.1</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>9.7</td>
<td>5.1</td>
<td></td>
</tr>
<tr>
<td>1000*</td>
<td>7.6</td>
<td>2.2</td>
<td>2.9</td>
</tr>
</tbody>
</table>

(* Results for 1000h from [Lüscher & Beck 2019]*)
### Results on LibriSpeech Test

Results published at this Interspeech* and coming ASRU° 2019

<table>
<thead>
<tr>
<th>data augm.</th>
<th>approach</th>
<th>encoder</th>
<th>WER [%]</th>
<th>clean</th>
<th>other</th>
<th>publication</th>
</tr>
</thead>
<tbody>
<tr>
<td>no</td>
<td>CTC</td>
<td>CNN</td>
<td>6gram</td>
<td>3.3</td>
<td>9.6</td>
<td>[Li &amp; Lavrukhin+ 2019]*</td>
</tr>
<tr>
<td></td>
<td>attention</td>
<td>TDS conv</td>
<td>CNN</td>
<td>3.3</td>
<td>9.8</td>
<td>[Hannun &amp; Lee+ 2019]*</td>
</tr>
<tr>
<td>yes</td>
<td>CTC</td>
<td>CNN</td>
<td>Trafo</td>
<td>2.8</td>
<td>7.8</td>
<td>[Li &amp; Lavrukhin+ 2019]*</td>
</tr>
<tr>
<td></td>
<td>attention</td>
<td>LSTM</td>
<td>LSTM</td>
<td>2.5</td>
<td>8.0</td>
<td>[Tüske &amp; Audhkhasi+ 2019]*</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.5</td>
<td>5.8</td>
<td>[Park &amp; Chan+ 2019]*</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Trafo</td>
<td>2.8</td>
<td>7.4</td>
<td>[Zeyer &amp; Bahar+ 2019]°</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>LSTM</td>
<td>2.4</td>
<td>8.2</td>
<td>[Kim &amp; Shin+ 2019]*</td>
</tr>
<tr>
<td></td>
<td>CTC+att’n</td>
<td>Trafo</td>
<td>RNN</td>
<td>2.6</td>
<td>5.7</td>
<td>[Karita &amp; Chen+ 2019]°</td>
</tr>
<tr>
<td>no</td>
<td>hybrid</td>
<td>LSTM</td>
<td>LSTM</td>
<td>2.6</td>
<td>5.5</td>
<td>[Lüscher &amp; Beck+ 2019]*</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Trafo</td>
<td>2.3</td>
<td>5.0</td>
<td></td>
</tr>
</tbody>
</table>

---

31 of 40

R. Schlüter: Modeling in ASR: Beyond (Standard) HMMs
Lehrstuhl Informatik 6 — Human Language Technology and Pattern Recognition
RWTH Aachen University

Sep. 16, 2019
Encoder

- even hybrid model can be seen as:
  - encoder (up to last hidden layer)
  - decoder (output activation + softmax: log-linear layer)

- formally, encoder modeling similar for all cases, e.g. using **deep bidirectional LSTMs**, some variation:
  - temporal sub-sampling
  - layer sizes

- however, parameterization after training may vary strongly, e.g. attention vs. hybrid:

Attention and corresponding encoder.

Phoneme positions and hybrid DNN/HMM encoder.
Alignment

Comparison of alignment/attention for exemplary SWB utterance.

- attention strongly **localized**, variation in label length covered by attention positioning
- → alignment: **interaction** between attention and encoder!
- → encoding: necessarily differs between hybrid and inverted HMM
- depending on modeling, inverted HMM aligns similar to hybrid HMM
Vocabulary Modeling

Goal:

- discard intermediate modeling based on pronunciations
  → avoid pronunciation lexicon
- enable direct vocabulary modeling
  → how to cover words unseen during training?
  e.g. character-based, even for HMM, cf. e.g. [Kanthak & Ney 2002], or Babel project

Approach:

- decompose words into subwords
  → enables open vocabulary, provided all characters are included
    (explicitly or implicitly over subsequences)
- byte-pair encoding (BPE) [Sennrich, Haddow+ 2016]
  – originally data compression approach
  – successive agglomeration of frequent character (byte) pairs
  – short BPE units: good statistics, but acoustic realization (pronunciation) possibly ambiguous
  – long BPE units/full words: proper pronunciation, but much longer tail of infrequent units
Beyond Zipf’s Law: Byte Pair Encoding

Label rank $r$ vs. frequency $N(r)$ for different vocabularies (Switchboard task).

(Dichotomy in Zipf’s Law: cf. [Montemurro 2001].)
Label Positional Perplexity Trend

- LibriSpeech Dev clean+other perplexities per word position:

Word 4-gram LM: approx. stationary.

Word LSTM LM: clear initial trend.

- Full/recurrent word context models show **trend** over word positions.
- Supports “middle-out decoding” approach proposed at this NeurIPS in [Mehri & Sigal 2018]
- Might partly explain directional asymmetry considered in [Mimura & Sakai+ 2018].
Search/Decoding, Domain-Dependence

- Various HMM approaches and CTC with LM: search includes **alignment optimization**.
- Attention: search only on label level, attention is not globally optimized: locally determined by the label history: constitutes **intermediate decisions** to some extent.
- Label-synchronous decoding: how to perform pruning? Which hypotheses are comparable? Relation to input coverage?
- Size of search space varies with model quality and with input properties, search in end-to-end systems often is reduced to small fixed-size beams.

Separate audio and text data resources:
- clear separation in **standard decomposition** into acoustic/language model
- speech chain allows inclusion of separate textual data during training [Tjandra+ 2017]: interpret concatenation of TTS and ASR (and vice versa) as text (speech audio) autoencoders
- [Sriram & Jun+ 2018] includes **LM in decoder training** to prevent that the decoder implicitly learns LM information
Conclusions

Outline

Introduction

Statistical Sequence Classification

ASR Architectures: State-of-the-Art in Transition

ASR Modeling Approaches

Conclusions

References
Conclusions

Common characterization of end-to-end systems:
- directly convert input (audio signal) into output (word sequences)
- do not involve intermediate representations (ASR: phoneme set, pronunciation lexicon)
- can be trained from scratch end-to-end to optimize performance measure (ASR: word error rate)

Discussion:
- **Integrated decision** end-to-end based on all knowledge sources:
  - natural goal of statistical approach to ASR - caveats: beam search, search complexity?
- Existing **knowledge sources** (e.g. signal processing, phonetic, temporal segmentation, existing models like multilingual features, etc.) may be viewed as additional (possibly noisy or mismatched) “data” - using it may still help, especially if primary training data is sparse.
- Internal structuring provides intermediate representations that enable internal model analysis to some extent.
- taking **training from scratch** literally would also exclude pretraining or any hyperparameter optimization (aka repeated training and testing on held out data).
- Training hierarchically with intermediate representations and corresponding objectives provides potential modes of initialization, regularization, and analysis.
- Transition between training from scratch and using prior knowledge needed: supported by machine learning methods.
Conclusions

Current Situation

Training

• Any ASR system today is sequence discriminative trainable.
• However: pretraining/prior training with different objective might be necessary.
• Hyperparameter optimization concerns all approaches.
• Varying amounts of training data:
  – Insertion of external knowledge sources?
  – Transition from standard to novel end-to-end models?

Recognition:

• Strictly speaking, only CTC fully searchable (but...).
• Small vocabulary and short context LM: no pruning needed.
• All others not strictly optimal, incl. end-to-end:
  – Beam search, pruning: global optimum not guaranteed.
  – Exponential search tree with RNN LM and/or decoder.
  – How does an end-to-end system indicate uncertainty?
    → Calibration [Guo & Pleiss+ 2017] needed?
    – “Two-Pass End-to-End” Speech Recognition [Sainath & Pang+ 2019]
Thank you for your attention!
Outline

Introduction

Statistical Sequence Classification

ASR Architectures: State-of-the-Art in Transition

ASR Modeling Approaches

Conclusions

References
References


H. Bourlard, C. J. Wellekens: 'Links between Markov Models and Multilayer Perceptrons', in D.S. Touretzky (ed.): "Advances in Neural Information
References


M.A. Castano, E. Vidal, F. Casacuberta: Inference of stochastic regular languages through simple recurrent networks. IEE Colloquium on Grammatical Inference: Theory, Applications and Alternatives, pp. 16/1-6, Colchester, UK, April 1993.


References


---

R. Schlüter: Modeling in ASR: Beyond (Standard) HMMs
Lehrstuhl Informatik 6 — Human Language Technology and Pattern Recognition
RWTH Aachen University
Sep. 16, 2019
References


References


References


References


References


References


References


References


