Neural Methods in Automatic Speech Recognition

The “source-channel” model for automatic speech recognition (ASR)

- **Language Model**
  - \( P(W) \)
  - **Speaker’s Mind**

- **Acoustic Model**
  - \( P(A|W) \)
  - **Stochastic Communication Channel**

- **Decoder**
  - \( \text{arg max } P(\bullet|A) \)
  - **Listener’s Mind**

- **Listener’s Auditory Apparatus**
- **Speaker’s Vocal Apparatus**
- **Stochastic Communication Channel**
Hidden Markov models are popular as acoustic models

\[
P(A | W) = \sum_{s \in S(W)} P(A, S | W) = \sum_{s \in S(W)} P(A | S, W)P(S | W)
\]

\[
\approx \sum_{s \in S(W)} P_E(A | S)P_T(S)
\]

\[
= \sum_{s \in S(W)} P_E(a_1, a_2, \ldots, a_T | s_1, s_2, \ldots, s_T)P_T(s_1, s_2, \ldots, s_T)
\]

\[
= \sum_{s \in S(W)} \prod_{t=1}^{T} P_E(a_t | s_t)P_T(s_t | s_{t-1})
\]
Dynamic programming is popular for “decoding,” i.e. for hypothesis search

\[
\hat{W} = \arg\max_{W} P(A | W)P(W)
\]

\[
= \arg\max_{W} \sum_{S \in S(W)} P(A | S)P(S)P(W)
\]

\[
\approx \arg\max_{W} \max_{S \in S(W)} P(A | S)P(S)P(W)
\]

\[
= \arg\max_{W} \max_{S \in S(W)} \log P(A | S) + \log P(S) + \log P(W)
\]

\[
\equiv \text{Project (Bestpath (Compose (A_{\log P(A | S)} \circ L_{\log P(S)} \circ G_{\log P(W)})))}
\]
Composite HMM for “cat and hat”

Phoneme HMMs

- phonemes: k, ae, t, n, h, d
- composite HMM structure for “cat and hat”

Waveform graphs for “cat” and “hat”
Composite HMM for “cat and hat”

“Forward” Algorithm

\[
P(y|w) = \sum_{s \in \mathcal{S}(w)} P_\theta(y|s)P_t(s)
\]

\[
= \sum_{s \in \mathcal{S}(w)} \prod_{i=1}^{n} P_\theta(y_i|s_i)P_t(s_i|s_{i-1})
\]

Viterbi Algorithm

\[
\hat{s} = \arg \max_{s \in \mathcal{S}(w)} P(s|y)
\]

\[
= \arg \max_{s \in \mathcal{S}(w)} \frac{P(y, s)}{P(y)}
\]

\[
= \arg \max_{s \in \mathcal{S}(w)} \prod_{i=1}^{n} P_\theta(y_i|s_i)P_t(s_i|s_{i-1})
\]
Acoustic Modeling with Deep Neural Networks for Hybrid ASR Systems

Repurposing Algorithms Developed for HMM-based Architectures
A paper appeared in September 2011...

**Conversational Speech Transcription**
**Using Context-Dependent Deep Neural Networks**

*Frank Seide*, Gang Li, and Dong Yu

1Microsoft Research Asia, Beijing, P.R.C.
2Microsoft Research, Redmond, USA

**Phoneme Recognition: Neural Networks vs. Hidden Markov Models**

A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, K. Lang

ATR Interpreting Telephony Research Laboratories
*University of Toronto and Canadian Institute for Advanced Research
†Carnegie-Mellon University
So, a lot of progress has been made since 1988.
Phoneme HMMs

Phoneme Posterior Probabilities

Acoustic Likelihoods

\[ p(h | y_i) \]

\[ p(y_i | \phi) = \frac{p(\phi | y_i) p(y_i)}{p(\phi)} \propto \frac{p(\phi | y_i)}{p(\phi)} \]

\[ \mathcal{L}_{CE}(\boldsymbol{\theta}) = - \sum_{i=1}^{n} \log p_\theta(\hat{\phi}_i | y_i) \]
\[
\text{argmin}_\Theta = \log \sum_{\Phi} \prod_{b} p_{\theta}(\Phi|\mathbf{b}) \prod_{d} p_{\theta}(\mathbf{b}|\Phi) \\
\]
Purely sequence-trained neural networks for ASR based on lattice-free MMI

Daniel Povey\textsuperscript{1,2}, Vijayaditya Peddinti\textsuperscript{1}, Daniel Galvez\textsuperscript{3}, Pegah Ghahremani\textsuperscript{1}, Vimal Manohar\textsuperscript{1}, Xingyu Na\textsuperscript{4}, Yiming Wang\textsuperscript{1}, Sanjeev Khudanpur\textsuperscript{1,2}

\textsuperscript{1}Center for Language and Speech Processing, The Johns Hopkins University
\textsuperscript{2}HLT CoE, The Johns Hopkins University
\textsuperscript{3}Department of Computer Science, Cornell University
\textsuperscript{4}Lele Innovation and Intelligence Technology (Beijing) Co., Ltd.

\{dpovey,dt.galvez,asr.naxingyu\}@gmail.com,
\{vijay.p,vmanohal,pghahrel,yiming.wang,khudanpur\}@jhu.edu
Language Modeling with (Recurrent) Neural Networks

Efforts to Go Beyond $n$-gram Dependence in Language Models
Using Neural Networks to Estimate $P(w_t|h_t)$

A STUDY OF ENGLISH WORD CATEGORY PREDICTION BASED ON NEURAL NETWORKS
Masami NAKAMURA, Kiyoshiro SHIKANO
ATR Interpreting Telephony Research Laboratories
Seika-chou, Souraku-gun, Kyoto 619-02, JAPAN

Continuous space language models

Holger Schwenk

Spoken Language Processing Group, LIMSI-CNRS, BP 133, 91403 Orsay cedex, France
Received 19 December 2005; received in revised form 15 September 2006; accepted 15 September 2006
Available online 9 October 2006
A paper appeared in September 2010 ...

INTERSPEECH 2010

Recurrent neural network based language model

Tomáš Mikolov1,2, Martin Karafiát1, Lukáš Burget1, Jan “Honza” Černocký3, Sanjeev Khudanpur2

1Speech@FIT, Brno University of Technology, Czech Republic
2Department of Electrical and Computer Engineering, Johns Hopkins University, USA
{imikolov, karafiat, burget, cernocky}@fit.vutbr.cz, khudanpur@jhu.edu

Abstract

A new recurrent neural network based language model (RNN LM) with applications to speech recognition is presented. Results indicate that it is possible to obtain around 50% reduction of perplexity by using mixture of several RNN LMs, compared to a state of the art backoff language model. Speech recognition experiments show around 15% reduction of word error rate on the Wall Street Journal task when comparing models trained on the same amount of data, and around 5% on the much harder NIST RT05 task, even when the backoff model is trained on much more data than the RNN LM. We provide ample empirical evidence to suggest that connectionist language models are superior to standard n-gram techniques, except their high computational (training) complexity.

Index Terms: language modeling, recurrent neural networks, speech recognition
A Simple RNN Language Model

An RNN LSTM Language Model
Training Neural Networks without using HMM-Based Alignments

Connectionist Temporal Classification
Phoneme HMMs

Phoneme Posterior Probabilities

\[ p(\phi | y_i) \]

\[ p(y_i | \phi) = \frac{p(\phi | y_i)p(y_i)}{p(\phi)} \propto \frac{p(\phi | y_i)}{p(\phi)} \]

"cat and hat"

\[ \mathcal{L}_{CE}(\theta) = -\log \prod_{i=1}^{n} p_{\theta}(\hat{\phi}_i | y_i) = -\sum_{i=1}^{n} \log p_{\theta}(\hat{\phi}_i | y_i) \]
\[ \mathcal{L}_{\text{CE}}(\theta) = - \log \prod_{i=1}^{n} p_{\theta}(\hat{y}_i | y_i) \]

\[ \mathcal{L}_{\text{CTC}}(\theta) = - \log \sum_{t} \prod_{i=1}^{n} p_{\theta}(y_t | y_i) \]

\[ \mathcal{L}_{\text{HMM}}(\theta) = - \log \sum_{t} \prod_{i=1}^{n} p_{\theta}(y_t | y_i)p_{\theta}(\phi_{t,i} | \phi_{t,i-1}) \]
Calculating the CTC loss for “cat and hat”

\[ \mathcal{L}_{CTC}(\theta) = -\log \sum_{t} \prod_{i=1}^{n} p_{\theta}(\phi_{t_{i}} | y_{i}) \]

Calculating the gradient of the CTC loss

\[ \frac{\partial \mathcal{L}_{CTC}(\theta)}{\partial p_\theta(\phi | y_{j})} = \sum_{t} \prod_{i=1}^{n} p_{\theta}(\phi_{t_{i}} | y_{i}) \sum_{t: \phi_{t_{j}} = \phi} \frac{1}{p_\theta(\phi | y_{j})} \prod_{i=1}^{n} p_{\theta}(\phi_{t_{i}} | y_{i}) \]

\[ \frac{\partial \mathcal{L}_{HMM}(\theta)}{\partial p_\theta(\phi | y_{j})} = -\frac{1}{p_\theta(\phi | y_{j})} \sum_{t: \phi_{t_{j}} = \phi} \prod_{i=1}^{n} p_{\theta}(\phi_{t_{i}} | y_{i}) \]
Phoneme HMMs

Composite HMM for “cat and hat”

The CTC “Blend” Symbol ($\beta$)

FSA of permissible CTC strings for “cat and hat”
\[ \mathcal{L}_{CTC}(\theta) = -\log \sum_{t} \prod_{i=1}^{n} p_\theta(\phi_{t_i} | y_i) \]
Neural Speech Recognition without HMMs (aka End2End ASR)

“Purely” CTC-Based Speech Recognition Architectures
End-to-End Speech Recognition

Deep Speech: Scaling up end-to-end speech recognition

Awni Hannun; Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, Andrew Y. Ng

Baidu Research – Silicon Valley AI Lab
The CNN Architecture
The CNN+LSTM Architecture
A Bidirectional LSTM Architecture (Deep Speech)
End-to-End Speech Recognition using Neural Networks with Attention

Applying ideas from machine translation to speech recognition
Attention-Based Speech Recognition

NeurIPS 2015
An Encoder-Decoder Architecture with Attention

Decoder Network

Encoder Network

Attention Mechanism

\[ e_{i,j} = \phi(s_{i-1}), \psi(h_j) \]

\[ \alpha_{i,j} = \frac{\exp e_{i,j}}{\sum_{j'} \exp e_{i,j'}} \]

\[ a_t = \sum_{j} \alpha_{t,j} h_j \]
Summary + Q&A

- Connectionist Temporal Classification: Graves (2006)