Sequence Discriminative Training

Steve Renals

Automatic Speech Recognition – ASR Lecture 14
2 April 2018

Recall: Maximum likelihood estimation of HMMs

- Maximum likelihood estimation (MLE) sets the parameters so as to maximize an objective function F_{MLE}:

$$F_{\text{MLE}} = \sum_{u=1}^{U} \log P_{\lambda}(\mathbf{X}_u \mid M(W_u))$$

for training utterances $\mathbf{X}_1 \ldots \mathbf{X}_U$ where W_u is the word sequence given by the transcription of the uth utterance, $M(W_u)$ is the corresponding HMM, and λ is the set of HMM parameters

Maximum mutual information estimation

- Maximum mutual information estimation (MMIE) aims to directly maximise the posterior probability (sometimes called conditional maximum likelihood). Using the same notation as before, with $P(w)$ representing the language model probability of word sequence w:

$$
\begin{aligned}
F_{\text{MMIE}} &= \sum_{u=1}^{U} \log P_\lambda(M(W_u) \mid \mathbf{X}_u) \\
&= \sum_{u=1}^{U} \log \frac{P_\lambda(\mathbf{X}_u \mid M(W_u)) P(W_u)}{\sum_{w'} P_\lambda(\mathbf{X}_u \mid M(w')) P(w')}
\end{aligned}
$$

Maximum mutual information estimation

- Maximum mutual information estimation (MMIE) aims to directly maximise the posterior probability (sometimes called conditional maximum likelihood). Using the same notation as before, with $P(w)$ representing the language model probability of word sequence w:

$$F_{\text{MMIE}} = \sum_{u=1}^{U} \log P_\lambda(M(W_u) \mid \mathbf{X}_u)$$

$$F_{\text{MLE}} = \sum_{u=1}^{U} \log \frac{P_\lambda(\mathbf{X}_u \mid M(W_u))P(W_u)}{\sum_{w'} P_\lambda(\mathbf{X}_u \mid M(w'))P(w')}$$

Maximum mutual information estimation

$$F_{\text{MMIE}} = \sum_{u=1}^{U} \log \frac{P_\lambda(\mathbf{X}_u \mid M(W_u))P(W_u)}{\sum_{w'} P_\lambda(\mathbf{X}_u \mid M(w'))P(w')}$$

Maximum mutual information estimation

$$F_{\text{MMIE}} = \sum_{u=1}^{U} \log \frac{P_\lambda(\mathbf{X}_u \mid M(W_u))P(W_u)}{\sum_{w'} P_\lambda(\mathbf{X}_u \mid M(w'))P(w')}$$

- **Numerator**: likelihood of data given correct word sequence ("clamped" to reference alignment)

Maximum mutual information estimation

$$F_{\text{MMIE}} = \sum_{u=1}^{U} \log \frac{P_\lambda(\mathbf{X}_u \mid M(W_u))P(W_u)}{\sum_{w'} P_\lambda(\mathbf{X}_u \mid M(w'))P(w')}$$

- **Numerator**: likelihood of data given correct word sequence ("clamped" to reference alignment)
- **Denominator**: total likelihood of the data given all possible word sequences – equivalent to summing over all possible word sequences estimated by the full acoustic and language models in recognition. ("free")

Maximum mutual information estimation

$$F_{\text{MMIE}} = \sum_{u=1}^{U} \log \frac{P_\lambda(\mathbf{X}_u \mid M(W_u))P(W_u)}{\sum_{w'} P_\lambda(\mathbf{X}_u \mid M(w'))P(w')}$$

- **Numerator**: likelihood of data given correct word sequence ("clamped" to reference alignment)
- **Denominator**: total likelihood of the data given all possible word sequences – equivalent to summing over all possible word sequences estimated by the full acoustic and language models in recognition. ("free")
- The objective function F_{MMIE} is optimised by making the correct word sequence likely (maximise the numerator), and all other word sequences unlikely (minimise the denominator)

Sequence training and lattices

- Computing the denominator involves summing over all possible word sequences – estimate by generating lattices, and summing over all words in the lattice
- In practice also compute numerator statistics using lattices (useful for summing multiple pronunciations)
- Generate numerator and denominator lattices for every training utterance
- Denominator lattice uses recognition setup (with a weaker language model)
- Each word in the lattice is decoded to give a phone segmentation, and forward-backward is then used to compute the state occupation probabilities
- Lattices not usually re-computed during training

MMIE is sequence discriminative training

- **Sequence:** like forward-backward (MLE) training, the overall objective function is at the sequence level – maximise the posterior probability of the word sequence given the acoustics $P_\lambda(M(W_u) \mid \mathbf{X}_u)$
- **Discriminative:** unlike forward-backward (MLE) training the overall objective function for MMIE is discriminative – to maximise MMI:
  - Maximise the numerator by increasing the likelihood of data given the correct word sequence
  - Minimise the denominator by decreasing the total likelihood of the data given all possible word sequences

  This results in "pushing up" the correct word sequence, while "pulling down" the rest

MPE: Minimum phone error

- Basic idea adjust the optimization criterion so it is directly related to word error rate
- Minimum phone error (MPE) criterion

MPE: Minimum phone error

- Basic idea adjust the optimization criterion so it is directly related to word error rate
- Minimum phone error (MPE) criterion

$$F_{\text{MPE}} = \sum_{u=1}^{U} \log \frac{\sum_W P_\lambda(\mathbf{X}_u \mid M(W))P(W)A(W, W_u)}{\sum_{W'} P_\lambda(\mathbf{X}_u \mid M(W'))P(W')}$$

- $A(W, W_u)$ is the phone transcription accuracy of the sentence W given the reference W_u

MPE: Minimum phone error

- Basic idea adjust the optimization criterion so it is directly related to word error rate
- Minimum phone error (MPE) criterion

$$F_{\text{MMIE}} = \sum_{u=1}^{U} \log \frac{\sum_{W} P_\lambda(\mathbf{X}_u \mid M(W_{\underline{u}}))P(W_{\underline{u}})A(W, W_u)}{\sum_{W'} P_\lambda(\mathbf{X}_u \mid M(W'))P(W')}$$

- $A(W, W_u)$ is the phone transcription accuracy of the sentence W given the reference W_u

MPE: Minimum phone error

- Basic idea adjust the optimization criterion so it is directly related to word error rate
- Minimum phone error (MPE) criterion

$$F_{\text{MPE}} = \sum_{u=1}^{U} \log \frac{\sum_{W} P_{\lambda}(\mathbf{X}_u \mid M(W))P(W)A(W, W_u)}{\sum_{W'} P_{\lambda}(\mathbf{X}_u \mid M(W'))P(W')}$$

- $A(W, W_u)$ is the phone transcription accuracy of the sentence W given the reference W_u
- F_{MPE} is a weighted average over all possible sentences w of the raw phone accuracy
- Although MPE optimizes a phone accuracy level, it does so in the context of a word-level system: it is optimized by finding probable sentences with low phone error rates

HMM/DNN systems

- DNN-based systems are discriminative – the cross-entropy (CE) training criterion with softmax output layer "pushes up" the correct label, and "pulls down" competing labels
- CE is a frame-based criterion – we would like a sequence level training criterion for DNNs, operating at the word sequence level
- Can we train DNN systems with an MMI-type objective function?

HMM/DNN systems

- DNN-based systems are discriminative – the cross-entropy (CE) training criterion with softmax output layer "pushes up" the correct label, and "pulls down" competing labels
- CE is a frame-based criterion – we would like a sequence level training criterion for DNNs, operating at the word sequence level
- Can we train DNN systems with an MMI-type objective function? – **Yes**

Sequence training of hybrid HMM/DNN systems

- Forward- and back-propagation equations are structurally similar to forward and backward recursions in HMM training
- Initially train DNN framewise using cross-entropy (CE) error function
  - Use CE-trained model to generate alignments and lattices for sequence training
  - Use CE-trained weights to initialise weights for sequence training
- Train using back-propagation with sequence training objective function (e.g. MMI)

Sequence training results on Switchboard (Kaldi)

Results on Switchboard "Hub 5 '00" test set, trained on 300h training set, comparing maximum likelihood (ML) and discriminative (BMMI) trained GMMs with framewise cross-entropy (CE) and sequence trained (MMI) DNNs. GMM systems use speaker adaptive training (SAT).
All systems had 8859 tied triphone states.
GMMs – 200k Gaussians
DNNs – 6 hidden layers each with 2048 hidden units

	SWB	CHE	Total
GMM ML (+SAT)	21.2	36.4	28.8
GMM BMMI (+SAT)	18.6	33.0	25.8
DNN CE	14.2	25.7	20.0
DNN MMI	12.9	24.6	18.8

Veseley et al, 2013.

Lattice-Free MMI (LF-MMI) 1

- Sequence training of NN systems requires initially training a CE model to give a (very good) weight initialisation and to generate lattices for the denominator computation
- Lattice-free MMI (Povey et al, 2016) (sometimes called the 'Chain' model)
    - Avoids the need to pre-compute lattices for the denominator
    - Avoids the requirement to train using frame-based CE loss function, before sequence training
- Denominator calculation directly applies forward-backward computations to the denominator; speed-ups:
    - *phone-level* language model (typically 4-gram) (rather than word-level)
    - process training input in 1 second chunks (for GPU memory reasons)
    - Use 30ms frame rate at the output
    - Use a simpler HMM topology (hence fewer states, and a smaller output layer)

Lattice-Free MMI (LF-MMI) 2

- LF-MMI is vulnerable to overfitting:
  - L2 regularization on the network output (aims to prevent over-confident likelihood estimations)
  - Multitask training: train the network with two output layers – one trained using MMI, the other trained at the frame-level using CE. Only the MMI output layer is used for recognition, but the network learns to optimise both MMI and CE.
- LF-MMI in practice
  - Faster than conventional training – subsampling at output layer (30ms frame rate), smaller networks (fewer HMM states)
  - Similar word error rates to sequence training
  - In practice LF-MMI is more sensitive to noisy training transcripts compared with frame based CE or conventional sequence training

LF-MMI word error rates on various ASR tasks

pre ASR Data Set	Size	CE	CE →sMBR	LF-MMI	Rel. Δ
AMI-IHM	80 hrs	25.1%	23.8%	22.4%	6%
AMI-SDM	80 hrs	50.9%	48.9%	46.1%	6%
TED-LIUM*	118 hrs	12.1%	11.3%	11.2%	0%
Switchboard	300 hrs	18.2%	16.9%	15.5%	8%
LibriSpeech	960 hrs	4.97%	4.56%	4.28%	6%
Fisher + Switchboard	2100 hrs	15.4%	14.5%	13.3%	8%

TDNN acoustic models
Similar architecture across LVCSR tasks

Povey et al, 2016

Summary

- Sequence training: discriminatively optimise GMM or DNN to a sentence (sequence) level criterion rather than a frame level criterion
  - ML training of HMM/GMM – sequence-level, not discriminative
  - CE training of HMM/NN – discriminative at the frame level
  - MMI training of HMM/GMM or HMM/NN – discriminative at the sequence level
- Usually initialise sequence discriminative training
  - HMM/GMM – first train using ML, followed by MMI
  - HMM/NN – first train at frame level (CE), followed by MMI
- Sequence discriminative training is computationally costly – need to compute the "denominator lattices"
- Lattice-free MMI for HMM/NN
  - avoids the need to compute denominator lattices
  - avoids the need to first apply CE training

Reading

- HMM discriminative training: Sec 27.3.1 of: S Young (2008), "HMMs and Related Speech Recognition Technologies", in *Springer Handbook of Speech Processing*, Benesty, Sondhi and Huang (eds), chapter 27, 539–557. http://www.inf.ed.ac.uk/teaching/courses/asr/2010-11/restrict/Young.pdf
- NN sequence training: K Vesely et al (2013), "Sequence-discriminative training of deep neural networks", Interspeech-2013, http://homepages.inf.ed.ac.uk/aghoshal/pubs/is13-dnn_seq.pdf
- Lattice-free MMI: D Povey et al (2016), "Purely sequence-trained neural networks for ASR based on lattice-free MMI", Interspeech-2016. http://www.danielpovey.com/files/2016_interspeech_mmi.pdf; slides – http://www.danielpovey.com/files/2016_interspeech_mmi_presentation.pptx