Discriminative Training of GMM-based systems

Steve Renals

Automatic Speech Recognition— ASR Lecture 15
10 March 2014

Discriminative training

- Basic idea Estimate the parameters of a speech recognizer so as to make the fewest classification errors (optimize the word error rate)

Discriminative training

- Basic idea Estimate the parameters of a speech recognizer so as to make the fewest classification errors (optimize the word error rate)
- Generative model: estimate the parameters so that the model reproduces the training data with the greatest probability (maximum likelihood)

Discriminative training

- Basic idea Estimate the parameters of a speech recognizer so as to make the fewest classification errors (optimize the word error rate)
- Generative model: estimate the parameters so that the model reproduces the training data with the greatest probability (maximum likelihood)
- Generative modelling only results in minimum classification error if certain conditions are met, including
    - the model is correct (i.e. the true data source is an HMM)
    - infinite training data

  This never happens in practice

Discriminative training

- Basic idea Estimate the parameters of a speech recognizer so as to make the fewest classification errors (optimize the word error rate)
- Generative model: estimate the parameters so that the model reproduces the training data with the greatest probability (maximum likelihood)
- Generative modelling only results in minimum classification error if certain conditions are met, including
    - the model is correct (i.e. the true data source is an HMM)
    - infinite training data

  This never happens in practice
- Discriminative training:
    - Focus on learning *boundaries* between classes
    - Consider incorrect word sequences as well as correct word sequences
    - This is related to direct optimisation of the posterior probability of the words given the acoustics $P(W \mid \mathbf{X})$

Neural network acoustic models

- Neural networks are discriminatively trained at the **frame** level

Neural network acoustic models

- Neural networks are discriminatively trained at the **frame** level
- Consider a context-dependent DNN
  - Output is a softmax over HMM states
  - Training involves increasing the probability of the correct state — and hence decreasing the probabilities of the others, since probabilities sum to 1
  - Frame-level discrimination — the network learns to optimise discrimination at the frame level by choosing the best state at each time frame

Neural network acoustic models

- Neural networks are discriminatively trained at the **frame** level
- Consider a context-dependent DNN
  - Output is a softmax over HMM states
  - Training involves increasing the probability of the correct state — and hence decreasing the probabilities of the others, since probabilities sum to 1
  - Frame-level discrimination — the network learns to optimise discrimination at the frame level by choosing the best state at each time frame
- **Sequence discrimination** — train the system to select the best sequence of frames by increasing the probability of the best sequence and decreasing the probability of all competing sequences

Neural network acoustic models

- Neural networks are discriminatively trained at the **frame** level
- Consider a context-dependent DNN
  - Output is a softmax over HMM states
  - Training involves increasing the probability of the correct state — and hence decreasing the probabilities of the others, since probabilities sum to 1
  - Frame-level discrimination — the network learns to optimise discrimination at the frame level by choosing the best state at each time frame
- **Sequence discrimination** — train the system to select the best sequence of frames by increasing the probability of the best sequence and decreasing the probability of all competing sequences
- Can train both GMM and DNN based models using sequence discrimination

Neural network acoustic models

- Neural networks are discriminatively trained at the **frame** level
- Consider a context-dependent DNN
  - Output is a softmax over HMM states
  - Training involves increasing the probability of the correct state — and hence decreasing the probabilities of the others, since probabilities sum to 1
  - Frame-level discrimination — the network learns to optimise discrimination at the frame level by choosing the best state at each time frame
- **Sequence discrimination** — train the system to select the best sequence of frames by increasing the probability of the best sequence and decreasing the probability of all competing sequences
- Can train both **GMM** and DNN based models using sequence discrimination

Discriminative training criteria

- Minimum classification error (MCE)
  - Correct parameters if misrecognition occurs
  - Discriminant function is the difference between the log likelihood of the correct sentence and the average likelihood of incorrect competitors
  - Used mainly for small vocabularies
  - Uses training data inefficiently (only considers misrecognised examples)

Discriminative training criteria

- Minimum classification error (MCE)
  - Correct parameters if misrecognition occurs
  - Discriminant function is the difference between the log likelihood of the correct sentence and the average likelihood of incorrect competitors
  - Used mainly for small vocabularies
  - Uses training data inefficiently (only considers misrecognised examples)
- Maximum mutual information estimation (MMIE)
  - Maximise the mutual information between the acoustics and word sequence
  - Variant of conditional maximum likelihood

Discriminative training criteria

- Minimum classification error (MCE)
  - Correct parameters if misrecognition occurs
  - Discriminant function is the difference between the log likelihood of the correct sentence and the average likelihood of incorrect competitors
  - Used mainly for small vocabularies
  - Uses training data inefficiently (only considers misrecognised examples)
- Maximum mutual information estimation (MMIE)
  - Maximise the mutual information between the acoustics and word sequence
  - Variant of conditional maximum likelihood
- Minimum Bayes Risk (MBR)
  - Optimise the word error rate rather than a likelihood ratio
  - Use the string edit distance between competing and reference utterances
  - Minimum phone error (MPE) training

Maximum likelihood estimation (MLE)

- 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 likelihood estimation (MLE)

- 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
- This objective function can be maximised by the EM algorithm (Forward-Backward algorithm when applied to HMMs)

MLE — Updating the mean

- Update equation for the mean vector $\boldsymbol{\mu}^{jm}$ for Gaussian component m of GMM associated with state s_j is:

$$\hat{\boldsymbol{\mu}}^{jm} = \frac{\sum_{u=1}^{U} \sum_{t=1}^{T} \gamma_t^u(s_j, m)\mathbf{x}_t^u}{\sum_{u=1}^{U} \sum_{t=1}^{T} \gamma_t^u(s_j, m)}$$

where $\gamma_t^u(s_j, m)$ is the probability of the model occupying mixture component m of state j at time t given training sentence $\mathbf{X}_u$.

MLE — Updating the mean

- Update equation for the mean vector $\boldsymbol{\mu}^{jm}$ for Gaussian component m of GMM associated with state s_j is:

$$\hat{\boldsymbol{\mu}}^{jm} = \frac{\sum_{u=1}^{U} \sum_{t=1}^{T} \gamma_t^u(s_j, m)\mathbf{x}_t^u}{\sum_{u=1}^{U} \sum_{t=1}^{T} \gamma_t^u(s_j, m)}$$

  where $\gamma_t^u(s_j, m)$ is the probability of the model occupying mixture component m of state j at time t given training sentence $\mathbf{X}_u$.

- Some extra notation:

$$\Theta_{jm}^u(M) = \sum_{t=1}^{T} \gamma_t^u(s_j, m)\mathbf{x}_t^u \qquad \Gamma_{jm}^u(M) = \sum_{t=1}^{T} \gamma_t^u(s_j, m)$$

$$\hat{\boldsymbol{\mu}}^{jm} = \frac{\sum_{u=1}^{U} \Theta_{jm}^u(M(W_u))}{\sum_{u=1}^{U} \Gamma_{jm}^u(M(W_u))}$$

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

- **Numerator**: $P_\lambda(\mathbf{X}_u \mid M(W_u))P(W_u)$
  the likelihood of the data given the correct word sequence — similar to the MLE objective function. M_{num} is combined acoustic & language models used in the numerator

Maximum mutual information estimation

- **Numerator**: $P_\lambda(\mathbf{X}_u \mid M(W_u))P(W_u)$
  the likelihood of the data given the correct word sequence — similar to the MLE objective function. M_{num} is combined acoustic & language models used in the numerator
- **Denominator**: $\sum_{w'} P_\lambda(\mathbf{X}_u \mid M(w'_u))P(w'_u)$
  the total likelihood of the data given all possible word sequences — obtained by summing over all possible word sequences estimated by the full acoustic and language models in recognition (M_{den}):

$$P(\mathbf{X} \mid M_{\text{den}}) = \sum_{w'} P_\lambda(\mathbf{X}_u \mid M(w'_u))P(w'_u)$$

- **Numerator**: $P_\lambda(\mathbf{X}_u \mid M(W_u))P(W_u)$
  the likelihood of the data given the correct word sequence — similar to the MLE objective function. M_{num} is combined acoustic & language models used in the numerator
- **Denominator**: $\sum_{w'} P_\lambda(\mathbf{X}_u \mid M(w'_u))P(w'_u)$
  the total likelihood of the data given all possible word sequences — obtained by summing over all possible word sequences estimated by the full acoustic and language models in recognition (M_{den}):

$$P(\mathbf{X} \mid M_{\text{den}}) = \sum_{w'} P_\lambda(\mathbf{X}_u \mid M(w'_u))P(w'_u)$$

- 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)

Extended Baum-Welch (EBW)

- No EM-based optimization approach for F_{MMIE}
- Gradient-based approaches are straightforward but slow

Extended Baum-Welch (EBW)

- No EM-based optimization approach for F_{MMIE}
- Gradient-based approaches are straightforward but slow
- Approximation: Extended Baum-Welch (EBW) algorithm provides update formulae similar to forward-backward recursions used in MLE.
- Extended Baum-Welch — Updating the mean:

$$\hat{\boldsymbol{\mu}}^{jm} = \frac{\sum_{u=1}^{U} \left[\Theta_{jm}^{u}(M_{\text{num}}) - \Theta_{jm}^{u}(M_{\text{den}}) \right] + D \boldsymbol{\mu}^{jm}}{\sum_{u=1}^{U} \left[\Gamma_{jm}^{u}(M_{\text{num}}) - \Gamma_{jm}^{u}(M_{\text{den}}) \right] + D}$$

- Can interpret D as a weight between old and new estimates; in practice D estimated for each Gaussian to ensure variance updates are positive

EBW and Lattices

- Computing $\Theta^u_{jm}(M_{\text{den}})$ 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

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

Example: meeting speech recognition

WER for HMM/GMM system

System	Training criterion	WER/%
Baseline	ML	28.7
SAT	ML	27.6
SAT	MPE	24.5

Sequence training of hybrid HMM/DNN systems

- Can train HMM/NN systems using a MMI-type objective function (e.g. Bridle and Dodd, 1991)
- Forward- and back-propagation equations are structurally similar to forward and backward recursions in HMM training
- Was not used in practice, for another 20 years...
- Now used for DNN systems (e.g. Vesely et al, 2013)
- The tricky parts are in the optimisation and in the use of lattices to compute the denominator term...

Summary

- Discriminative methods optimize a criterion other than maximum likelihood (eg more directly related to the error rate)
- But, we still want to optimize all parameters according to a consistent criterion
  - MMI — directly optimise the posterior probability of the word sequence given the data
  - MPE — scale the posterior word sequence probability by an estimate of the phone error rate
- Discriminative training has a number of technical issues relating to smoothing the parameter updates

Reading

- Sec 27.3.1 of: S Young (2008). HMMs and Related Speech Recognition Technologies, in *Springer Handbook of Speech Processing*, J Benesty, MM Sondhi and Y Huang (eds), chapter 27, 539–557.
- NN sequence training:
  - Bridle & Dodd (1991), An Alphanet approach to optimising input transformations for continuous speech recognition, Proc IEEE ICASSP
  - Vesely et al (2013), Sequence-discriminative training of deep neural networks, Proc Interspeech