NVIDIA GTC: The Race To Perfect Voice Recognition Using GPUs

Google says its speech recognition technology now has only an 8% word error rate

IBM Watson can now transcribe speech perfectly

Baidu Announces Breakthrough In Speech Recognition, Claiming To Top Google And Apple

Time to get your babble on: Microsoft opens Skype Translator Preview to all comers

Chat to your mates in Klingon

THE COMPUTERS ARE LISTENING
HOW THE NSA CONVERTS SPOKEN WORDS INTO SEARCHABLE TEXT
Deep Learning
Neural Networks
Baidu Announces Breakthrough in Speech Recognition, Claiming To Top Google And Apple
The rise of neural networks (in speech recognition)
The rise and fall and rise and fall and rise of neural networks (in speech recognition)
The Perceptron (Rosenblatt) (early 1960s)
The Perceptron (Rosenblatt) (early 1960s)
NN Winter #1
(late 1960s to mid 1980s)
Hidden Markov models
(the cybernetic underground)

GENERAL MODEL

Let the sequence $X(1), X(2), X(3), \ldots, X(T)$ be the sequence of states of a Markov process $[3]$ with transition matrix $A = (a_{i,j})$. Let $Y(1), Y(2), Y(3), \ldots, Y(T)$ be a sequence of random variables such that, for all $t$, $\Pr(Y(t) = k \mid X(t - 1) = i, X(t) = j) = b_{i,j,k}$. Use a bracket and colon notation to abbreviate sequences. Thus $X[1:T] = X(1), X(2), X(3), \ldots, X(T)$ and $Y[1:T] = Y(1), Y(2), Y(3), \ldots, Y(T)$. The assumptions of the model are that

$$
\Pr(Y(t) = y(t) \mid X[1:t]) = \begin{cases} x[1:t], Y[1:t - 1] = y[1:t - 1] \\ \Pr(Y(t) = y(t) \mid X(t - 1) = x(t), X(t) = x(t)) = b_{x(t-1),x(t),y(t)} \end{cases}
$$

and

$$
\Pr(X(t) = x(t) \mid X[1:t - 1] = x[1:t - 1]) = \begin{cases} \Pr(X(t) = x(t) \mid X(t - 1) = x(t - 1)) = a_{x(t-1),x(t)} \end{cases}
$$

Under these assumptions,

$$
\Pr(X[1:T] = x[1:T], Y[1:T] = y[1:T]) = \prod_{t=1,T} a_{x(t-1),x(t),b_{x(t-1),x(t),y(t)}}
$$

where a special extra state $x(0)$ is introduced and $a_{x(0),i}$ and $b_{x(0),i,j,k}$ are defined appropriately.
MLPs and backprop
(mid 1980s - mid 1990s)
MLPs and backprop

- Train multiple layers of hidden units – nested nonlinear functions
- Powerful feature detectors
- Posterior probability estimation
- Theorem: any function can be approximated with a single hidden layer
Neural network acoustic models
“hybrid systems”

~40 CI phone outputs

~2000 hidden units
1 hidden layer

9x39 MFCC inputs

Boulsard & Morgan, 1988–94

DARPA RM 1992

Renals, Morgan, Cohen & Franco, ICASSP 1992
Neural network acoustic models
“hybrid systems”

Broadcast news 1998
20.8% WER
(best GMM-based system, 13.5%)
Cook, Christie, Ellis, Fosler-Lussier, Gotoh,
Kingsbury, Morgan, Renals, Robinson, & Williams,
DARPA, 1999

Bourlard & Morgan, 1988–94
Neural network acoustic models (1990s)
Limitations compared with GMMs

• Computationally restricted to monophone outputs
  • context dependent systems factored over multiple NNs – limited within-word context
• Training not easily parallelisable
  • experimental turnaround slower
• systems less complex (fewer parameters)
  • RNN – <100k parameters
  • MLP – ~1M parameters
• Rapid adaptation hard
NN Winter #2
(mid 1990s - ~2010)
NN acoustic models

Benefits

• Fewer limitations on inputs
  • Correlated features
  • Multi-frame windows
• Discriminative training criteria (frame level and sequence level)
• Can be used to generate ‘higher-level’ features
  • tandem, posteriorgrams
  • bottleneck features
(Deep) neural network acoustic models (2010s)

- **WIDE**
  - Softmax output layer
  - ~2000 hidden units
  - ~6000 CD phone outputs
- **DEEP**
  - Automatically learned feature extraction
  - 3-8 hidden layers
- **ACOUSTIC INPUT**
  - Spectral? Cepstral? Derived features?

Dahl, Yu, Deng & Acero, IEEE TASLP 2012
Hinton, Deng, Yu, Dahl, Mohamed, Jaitly, Senior, Vanhoucke, Nguyen, Sainath & Kingsbury, IEEE SP Mag 2012
Je suis un créateur de jeux.
J'ai fait des jeux en ligne depuis 10 ans.
Et à mon objectif pour la prochaine décennie est d'essayer de le rendre aussi facile de sauver le monde, dans la vraie vie, comme c'est de sauver le monde dans les jeux en ligne.
Maintenant "j" espère prévu pour cela, et elle comporte convaincre davantage de personnes, y compris vous tous passer plus de temps à jouer plus grands et de meilleurs jeux.
Maintenant, nous dépensons trois milliards d'heures par semaine, aux jeux en ligne.
Certsains d'entre vous pourriez penser, c'est beaucoup de temps à consacrer aux jeux.
Péut-être trop de temps compte tenu, combien de problèmes urgents que nous devons résoudre dans le monde réel.
Mais en fait, selon mes recherches à l'institut pour l'avenir, c'est en fait le contraire est vrai.
Trois milliards d'heures par semaine près du jeu n'est pas assez pour résoudre les problèmes les plus urgents.
Je suis en fait "l" pense que si nous voulons survivre au siècle prochain sur cette planète, nous devons augmenter ce total de façon spectaculaire.

I'm Jane McGonigal. I'm a game designer.
I've been making games online now for 10 years,
and my goal for the next decade
is to try to make it as easy
as it is to save the world in online games.
Now, I have a plan for this,
and it entails convincing more people,
including all of you, to spend more time
playing bigger and better games.
Right now we spend three billion hours a week
playing online games.
Some of you might be thinking,
"That's a lot of time to spend playing games.
Maybe too much time, considering
how many urgent problems we have to solve in the real world."
But actually, according to my research
at The Institute For The Future,
I'm a game designer
i've been making games online out for ten years
and at my goal for the next decade is to try to make it as easy to save the world in real life as it is to save the world in online games
now i hope planned for this and it entails convincing more people including all of you to spend more time playing bigger and better games
right now we spend three billion hours a week playing online games
some of you might be thinking that's a lot of time to spend playing games
maybe too much time considering how many urgent problems we have to solve in the real world
but actually according to my research at the institute for the future it's actually the opposite is true
three billion hours a week is not nearly enough gameplay to solve the world's most urgent problems
i'm in fact i believe that if we want to survive the next century on this planet we need to increase that total dramatically
i've calculated the total we need at twenty one billion hours of gameplay every week
so that's probably a bit of a counter intuitive idea so i'll just say it again what it sink in
NN research areas

WIDE
- Softmax output layer
- ~6000 CD phone outputs
- ~2000 hidden units

DEEP
- Automatically learned feature extraction
- 3-8 hidden layers

TRAINING
- optimisation, objective fn

ARCHITECTURES
- recurrent, convolutional, …

ACTIVATION FUNCTIONS
- pooling, RELU, gated units

WEIGHT SHARING
- adaptation, CNNs

ACOUSTIC INPUT
- Spectral? Cepstral? Derived features?
ASR challenges

- Fragile operation across different conditions – automatic adaptation to acoustic environment, speaker, domain of use, language
- Degraded acoustic signals – noise, reverberation, overlapping talkers, distant microphones
- Reliance on supervised approaches
- Models include relatively little speech knowledge
- Models do not factor different causes of variability
- Systems react crudely (if at all) to the context and the environment
Adaptation

Speaker adaptation

The task: given a small amount of data from a talker (10–300s) adapt the models to better match the talker

- Model-based adaptation – adapt the parameters of the model based on the target speaker
- Feature-based adaptation – adapt/normalise the acoustic features

Compact  |  Efficient  |  Unsupervised
Model-based NN adaptation

- Adaptation of different **weight subsets** (Liao 2013)
  - 5% relative decrease in WER when all 60M weights adapted
- Adaptation cost based on **KL divergence** between SI and SA output distributions (Yu et al 2013)
  - 3% relative decrease in WER on Switchboard
- Increase compactness by **SVD factorisation** of weight matrix (Xue et al 2014)
- Automatically adapt specific parameter subsets – **output biases** (Yao et al 2012), **slope and bias** of hidden units (Siniscalchi et al 2013)
- **Speaker codes** (Bridle & Cox 1990; Abdel-Hamid & Jiang 2013) – model-based adaptation using auxiliary features
LHUC

Key idea: add a learnable **speaker-dependent amplitude** to each hidden unit

**Compact:** 1 SD parameter per hidden unit

**Unsupervised:** No speaker-adaptive training

**Efficient:** Few iterations of gradient descent on test/adapt data
LHUC experiment

TED Talks

TED Talks – IWSLT tst2011

<table>
<thead>
<tr>
<th>Model</th>
<th>WER/%</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNN</td>
<td>15.2</td>
</tr>
<tr>
<td>RELU</td>
<td>15.2</td>
</tr>
<tr>
<td>Maxout</td>
<td>14.3</td>
</tr>
<tr>
<td>Sigmoid</td>
<td>13.7</td>
</tr>
<tr>
<td>+LHUC</td>
<td>13.5</td>
</tr>
<tr>
<td>+CMLLR</td>
<td>12.8</td>
</tr>
<tr>
<td>+CMLLR+LHUC</td>
<td>12.5</td>
</tr>
<tr>
<td>+CMLLR+LHUC</td>
<td>11.9</td>
</tr>
</tbody>
</table>

6-7%
LHUC experiment
TED Talks

TED Talks - IWSLT tst2011

WER/%

DNN
+LHUC
+CMLLR
+CMLLR+LHUC

Sigmoid
ReLU
Maxout

15.2 15.2
13.7
13.5
13.9
12.8
12.5
12.9
12.7
11.9
LHUC experiment
TED Talks

TED Talks - IWSLT tst2011

WER/%

15.2 15.2
14.3
13.7 13.5 12.8
13.9 13.6 12.5
12.9 12.7 11.9

DNN +LHUC +CMLLR +CMLLR+LHUC

Sigmoid RELU Maxout

15-17%
Distant Speech Recognition

Distant Speech Recognition

... so you have your energy source your user interface who’s controlling the chip ...

hmm

rustle

click
Distant speech recognition

Combine multiple mics using beamforming

Wiener filter noise cancellation
Smoothed tdoa estimates
Delay-sum beamforming

mic array

~4000 tied state outputs
2048 hidden units
6 hidden layers
11x120 FBANK inputs
Distant speech recognition

Combine multiple mics using DNN

- 8 x 11x120 FBANK inputs
- 2048 hidden units
- 6 hidden layers
- ~4000 tied state outputs
CNN acoustic model

Statics, deltas, double-deltas for all acoustic context frames

maxpool size 3

128 convolutional filterbanks

width 9
shift 1

5 sigmoid layers

~4000 tied state outputs

2048 hidden units
CNN – Multi-channel

Combine multiple mics using multi-channel CNN

~4000 tied state outputs

2048 hidden units

5 sigmoid layers

p1

h1

h2

h3

h4

p2

maxpool

... convolutional bands

shared weights

Inputs

v1

v2

v3

v4

v5

v40
CNN – Cross-channel

Combine multiple mics using cross-channel CNN

Inputs

shared weights

cross-band maxpooling

cross-channel maxpooling

convolutional bands

shared weights

Inputs
Some results

CNN has 7–8% WER reduction over DNN (16–19% WER reduction over GMM)
Hardware/software enablers

- GPU processing
  - efficient hardware for matrix-vector operations
- Linear algebra libraries
  - efficient software for matrix-vector operations
- NN and ASR software toolkits
  - efficient construction of various NN architectures and optimisations
- Can get started on a £3500 alienware machine + open source software
- What’s needed
  - efficient run-time implementations on mobile devices
Conclusions

• NNs lead to improvement in speech recognition
  • deep structures to learn feature representations,
  • wide context-dependent output,
  • temporal context at input, correlated features
• Many open issues
  • NN architectures
  • Robustly behaving speech recognition

Thanks!