Robust TTS duration modelling using DNNs

Natural Speech Technology

Edinburgh – Cambridge – Sheffield

Gustav Eje Henter
Srikanth Ronanki
Oliver Watts
Mirjam Wester
Zhizheng Wu
Simon King

Synopsis

1. Statistical parametric speech synthesis is sensitive to bad data and bad assumptions
2. Techniques from robust statistics can reduce this sensitivity
3. Robust techniques are able to synthesise improved durations from found audiobook data

Overview

1. Background
2. Making TTS robust
   - 2.1 MDN generation
   - 2.2 β-estimation
3. An experiment
   - 3.1 Setup
   - 3.2 Results
4. Conclusion

Why duration modelling?

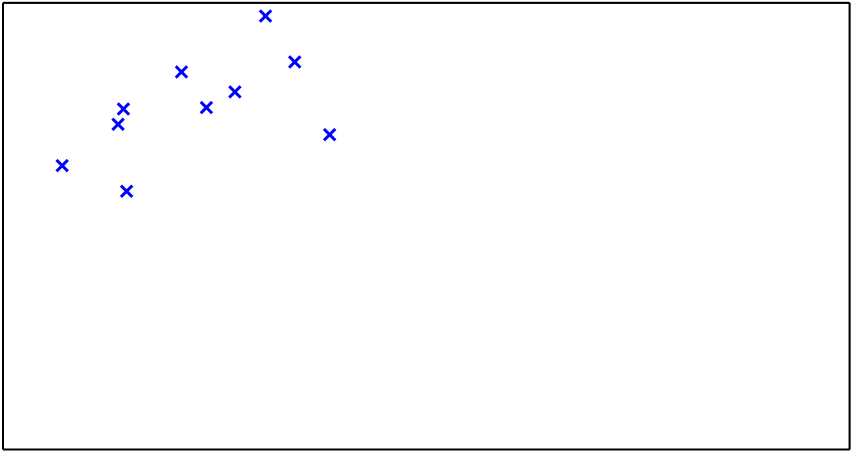
- Duration is a major component in natural speech prosody
- Current duration models are weak and unconvincing
- Throw data and computation at the problem
  - Speech data is all around us; let's use it!
  - Feed into a DNN

What problems are we addressing?

- A model is only as good as the data it is trained on
  - Errors in transcription, phonetisation, alignment, etc.
  - More of an issue in large, found datasets
- Real duration distributions are skewed and non-Gaussian
  - This does not match the models traditionally used

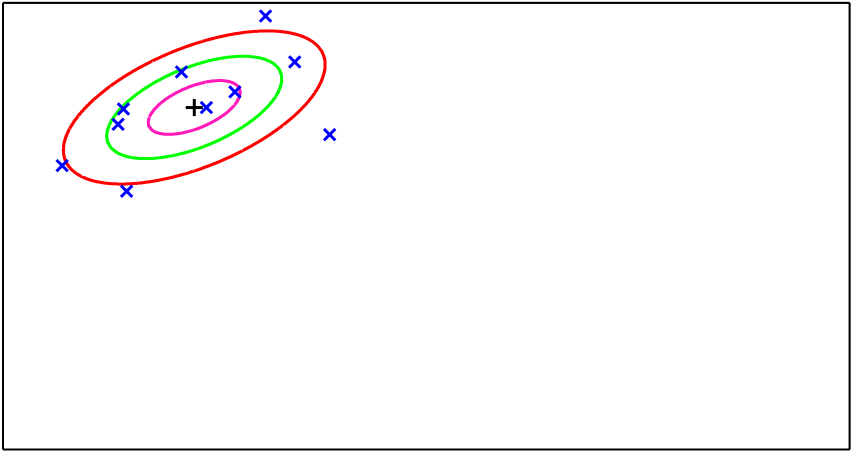
Toy example of problematic data

Generate some datapoints $\mathcal{D}$



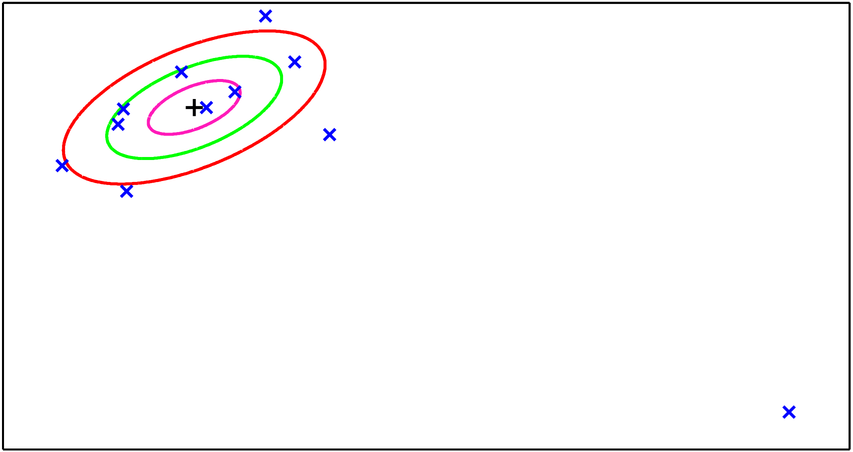
Toy example of problematic data

Fit a Gaussian using maximum likelihood



Toy example of problematic data

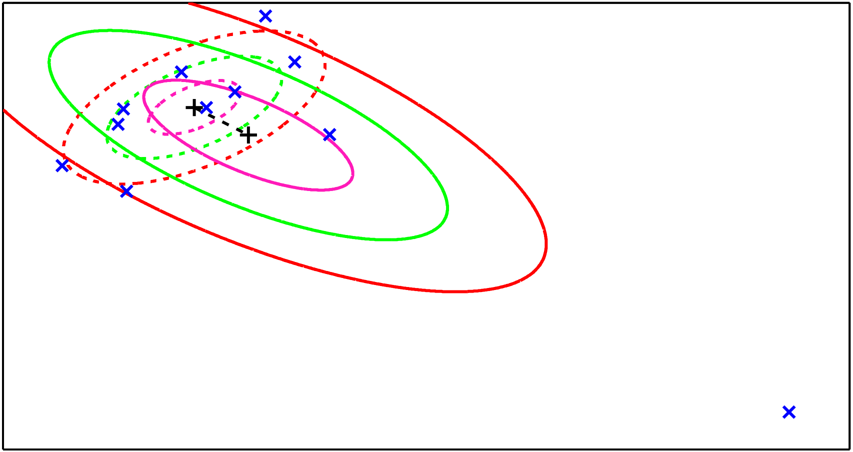
Add an unexpected datapoint



Toy example of problematic data

Natural Speech Technology

The maximum likelihood fit changes a lot!



Overview

1. Background
2. Making TTS robust
   - 2.1 MDN generation
   - 2.2 β-estimation
3. An experiment
   - 3.1 Setup
   - 3.2 Results
4. Conclusion

Robust statistics

The word "robust" can mean many things

- Here: Statistical techniques with low sensitivity to deviations from modelling assumptions
- Think: Modelling techniques that are able to *disregard poorly-fitting datapoints*
    - This assumes at least some data are good
- *Robust speech synthesis* is speech synthesis incorporating robust statistical techniques

Our work

- Phone-level: Disregarding sub-state duration vectors on a per phone basis
- Probabilistic: Probabilistic models have a natural notion of good/bad fit

Some definitions

- p is a phone instance
- $\boldsymbol{l}_p$ is a vector of (input) linguistic features
- $\boldsymbol{D}_p \in \mathbb{R}^D$ is a vector of stochastic (output) sub-state durations
- $\boldsymbol{d}_p$ is an outcome of $\boldsymbol{D}_p$
- $\mathcal{D} = \{(\boldsymbol{l}_p, \boldsymbol{d}_p)\}_p$ is a training dataset

Mixture density network

Assume phone durations are independent and follow a GMM

$$f_{\boldsymbol{D}}(\boldsymbol{d};\, \boldsymbol{\theta}) = \sum_{k=1}^{K} \omega_k \cdot f_{\mathcal{N}}(\boldsymbol{d};\, \boldsymbol{\mu}_k,\, \mathrm{diag}(\boldsymbol{\sigma}_k^2))$$

- Distribution parameters $\boldsymbol{\theta} = \{\omega_k,\, \boldsymbol{\mu}_k,\, \boldsymbol{\sigma}_k^2\}_{k=1}^{K}$ depend on $\boldsymbol{l}$ through a DNN $\boldsymbol{\theta}(\boldsymbol{l};\, \boldsymbol{W})$ with weights $\boldsymbol{W}$
- This is a *mixture density network* (MDN)
- Setting $K = 1$ yields a conventional Gaussian duration model

Estimation and generation

Natural Speech Technology

The network is typically trained using maximum likelihood

$$\widehat{\boldsymbol{W}}_{\mathrm{ML}}\left(\mathcal{D}\right) = \operatorname*{argmax}_{\boldsymbol{W}} \sum_{p \in \mathcal{D}} \ln f_{\boldsymbol{D}}(\boldsymbol{d}_p;\ \boldsymbol{\theta}(\boldsymbol{l}_p;\ \boldsymbol{W}))$$

Output durations are typically generated from the mode of the predicted distribution

$$\widehat{\boldsymbol{d}}_{\mathrm{MLPG}}\left(\boldsymbol{l}\right) = \operatorname*{argmax}_{\boldsymbol{d}} f_{\boldsymbol{D}}(\boldsymbol{d};\ \boldsymbol{\theta}(\boldsymbol{l};\ \widehat{\boldsymbol{W}}))$$

Two robust approaches

We describe two methods to create speech with robust durations:

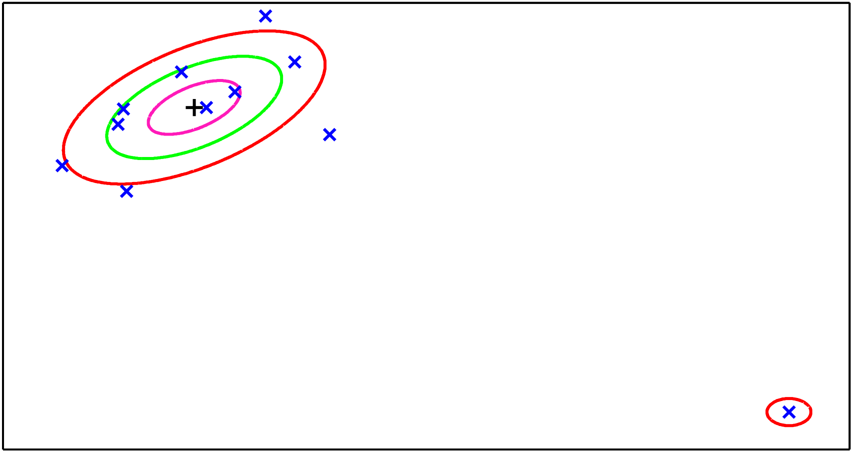
1. Generation-time robustness
   - Change model between estimation and synthesis
   - "Engineering approach"
2. Estimation-time robustness
   - Change parameter estimation technique
   - Grounded in robust statistics literature

Overview

1. Background
2. Making TTS robust
   2.1 MDN generation
   2.2 β-estimation
3. An experiment
   3.1 Setup
   3.2 Results
4. Conclusion

Fitting a mixture model

Additional components can absorb outlying datapoints



Generation-time robustness

Only generate from a single component:

$$k_{\max}(\boldsymbol{l}) = \operatorname*{argmax}_{k} \omega_k(\boldsymbol{l})$$
$$\widehat{\boldsymbol{d}}(\boldsymbol{l}) = \operatorname*{argmax}_{\boldsymbol{d}} f_{\mathcal{N}}(\boldsymbol{d};\, \boldsymbol{\mu}_{k_{\max}}(\boldsymbol{l}),\, \operatorname{diag}(\boldsymbol{\sigma}^2_{k_{\max}}(\boldsymbol{l})))$$

- Data attributed to lower-mass components is thus not used for the output
- Same as the generation principle for MDN acoustic models in Zen and Senior (2014)

Overview

1. Background
2. Making TTS robust
   2.1 MDN generation
   2.2 β-estimation
3. An experiment
   3.1 Setup
   3.2 Results
4. Conclusion

Training-time robustness

By changing the estimation principle away from MLE, we can get robustness with mathematical guarantees

- Even with $K = 1$, standard output generation, and no garbage model

β-estimation

In this work, we consider the estimation principle

$$\widehat{\boldsymbol{W}}_{\mathrm{M}\beta}(\mathcal{D}) = \underset{\boldsymbol{W}}{\operatorname{argmax}} \sum_{p \in \mathcal{D}} \Big(\big(f_{\boldsymbol{D}}(\boldsymbol{d}_p;\, \boldsymbol{\theta}(\boldsymbol{l}_p;\, \boldsymbol{W}))\big)^{\beta} - \frac{\beta}{1+\beta} \int \big(f_{\boldsymbol{D}}(\boldsymbol{x};\, \boldsymbol{\theta}(\boldsymbol{l}_p;\, \boldsymbol{W}))\big)^{1+\beta} \mathrm{d}\boldsymbol{x} \Big)$$

introduced by Basu et al. (1998), based on minimising the so-called density power divergence or β-divergence

- For lack of a better term, we will call this *β-estimation*

Statistical properties

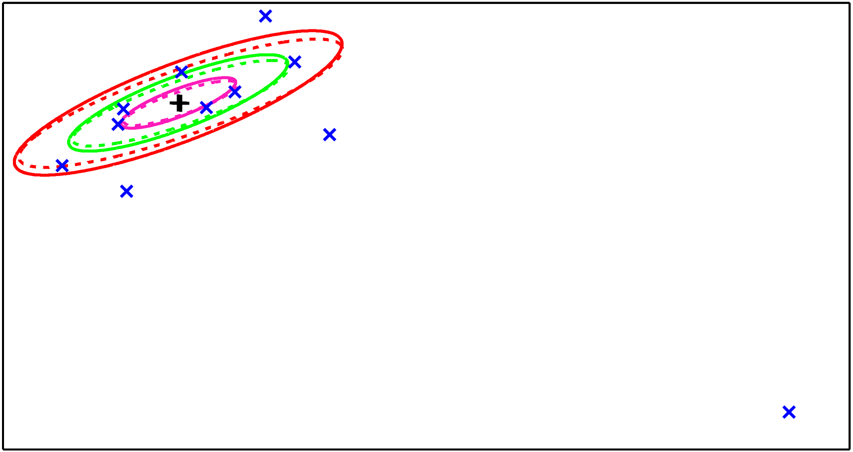
One can show that β-estimation is:

1. Consistent (if the data is clean)
2. Robust
3. Not (maximally) efficient
    - Since observations are discarded, more data is required to reach a certain estimation accuracy
    - The expected amount of data discarded can be used to set β

MLE is recovered in the limit $\beta \to 0$

β-estimation example

Gaussian distribution fit using $\beta = 1$



Overview

1. Background
2. Making TTS robust
   2.1 MDN generation
   2.2 β-estimation
3. An experiment
   3.1 Setup
   3.2 Results
4. Conclusion

Overview

1. Background
2. Making TTS robust
   2.1 MDN generation
   2.2 β-estimation
3. An experiment
   3.1 Setup
   3.2 Results
4. Conclusion

Setup in brief

Natural Speech Technology

- **Data:** Vol. 3 of Jane Austen's "Emma" from LibriVox as found TTS data ($\approx$ 3 hours)
- **Features:**
  - 592 binary + 9 continuous input features based on Festvox
    - Pauses inserted based on natural speech
  - 86×3 normalised output features (STRAIGHT)
- **DNN design:** 6 tanh layers with MDN output
- **Implementation:** Deep MDN code from Zhizheng Wu (Theano)

Reference systems

VOC Vocoded held-out natural speech (top line)

Same acoustic DNN, but different duration models:

- FRC Synthesised speech with oracle durations (forced-aligned to VOC)
- BOT Mean monophone duration (bottom line)
- MSE MMSE DNN (baseline)
- MLE1 Single-component, deep MDN maximising likelihood

Robust systems

MLE3 Three-component ($K = 3$), deep MDN maximising likelihood; only the maximum-weight component is used for synthesis

B75 Single-component, deep MDN optimising β-divergence, set to include approximately 75% of datapoints ($\beta = 0.358$)

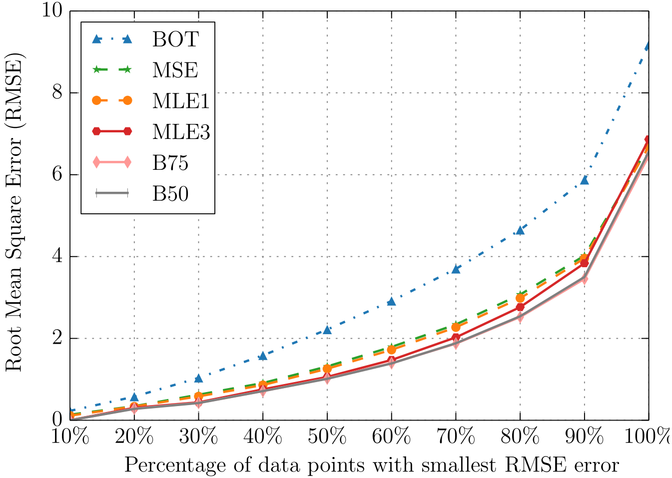
B50 Single-component, deep MDN optimising β-divergence, set to include approximately 50% of datapoints ($\beta = 0.663$)

Overview

1. Background
2. Making TTS robust
   2.1 MDN generation
   2.2 β-estimation
3. An experiment
   3.1 Setup
   3.2 Results
4. Conclusion

Outlier rejection

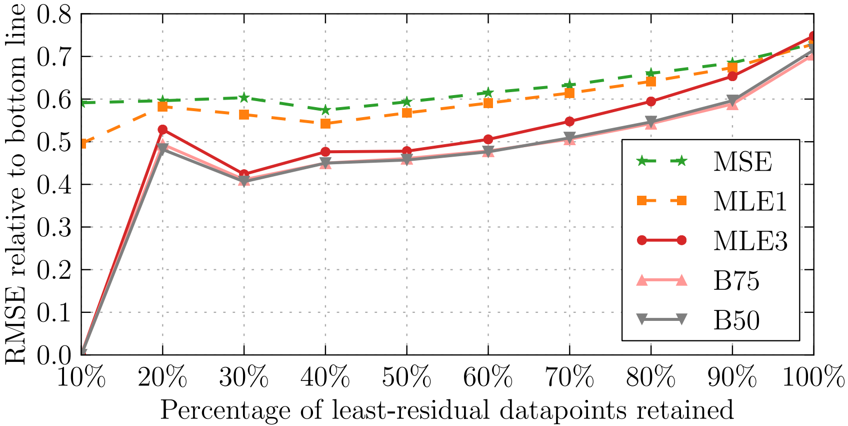
RMSE with respect to FRC on test-data subsets:



Outlier rejection

Natural Speech Technology

Relative RMSE on test-data subsets (with BOT at 1.0):

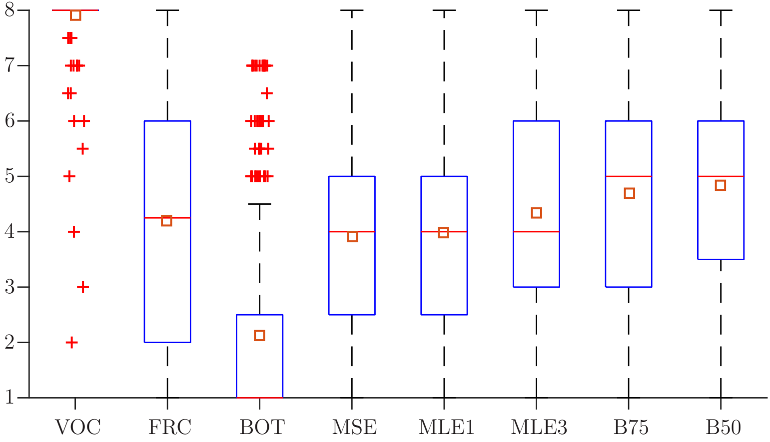


Listening test

- 21 held-out sentences (2–8 seconds long) used
- MUSHRA/preference test hybrid
  - Stimuli presented in parallel (unlabelled, random order)
  - No designated reference stimulus
  - Instructed to rank the different stimuli by preference
- 21 listeners
  - Each ranked 18 sentences in a balanced design
  - Remaining sentences used for training and GUI tutorial

Subjective results

Test results, after converting to ranks (higher is better):



Observations

- Robust duration models improve objective measures on the majority of the datapoints
  - Extreme examples are ignored, thus giving a better model of typical speech
- There are also improvements in subjective preference
  - Robust methods significantly outperform non-robust prediction methods
  - β-estimation even outperforms forced-aligned "oracle" durations

Overview

1. Background
2. Making TTS robust
   2.1 MDN generation
   2.2 β-estimation
3. An experiment
   3.1 Setup
   3.2 Results
4. Conclusion

Summary

1. Traditional synthesis methods are sensitive to errors
    - This can be incorrect data or assumptions
    - Big TTS data is likely to contain numerous errors

Summary

1. Traditional synthesis methods are sensitive to errors
   - This can be incorrect data or assumptions
   - Big TTS data is likely to contain numerous errors
2. Robust statistics can reduce the sensitivity
   - Better describes "typical speech"
   - Robust duration models preferred by listeners

The end

The end

Thank you for listening!

Bibliography

H. Zen and A. Senior, "Deep mixture density networks for acoustic modeling in statistical parametric speech synthesis," in *Proc. ICASSP*, 2014, pp. 3844–3848.

A. Basu, I. R. Harris, N. L. Hjort, and M. C. Jones, "Robust and efficient estimation by minimising a density power divergence," *Biometrika*, vol. 85, no. 3, pp. 549–559, 1998.

Example audio

Example utterance from held-out chapter:

VOC FRC BOT MSE MLE1
MLE3 B75 B50

Data

Audiobooks are a classic source of found TTS data

- Jane Austen's "Emma" from LibriVox
  - Volume 3, chapters 1–10
  - Read by Sherry Crowther (US English)
- 1739 utterances (92,025 non-silent phones)
  - 175 minutes total, 6.06 s average utterance duration
  - Train/dev/test sets: 1660/39/40 utterances

Input and output features

- 200 frames per second at 44.1 kHz
- Linguistic features
  - Based on Festvox
  - One-hot encoding of 592 categorical features $\boldsymbol{l}^{(b)}$
  - Nine continuous-valued features $\boldsymbol{l}^{(d)}$, normalised to range [0.01, 0.99]
- Acoustic features $\boldsymbol{x}$
  - STRAIGHT vocoder
    - Log-F0, 60 spectrum mel-ceps, 25 baps
  - Statics, deltas, and delta-deltas ($\approx$ 250 dimensions total)
  - Each dimension normalised to zero mean and unit variance

Synthesis steps

Natural Speech Technology

1. `ehmm` for acoustics-based pause/silence insertion
   - Oracle pausing strategy
2. text & pausing information $\rightarrow$ binary linguistic features $\boldsymbol{l}^{(b)}$
3. $\boldsymbol{l}^{(b)} \rightarrow$ DNN-predicted per-phone (rounded) Gaussian mean state durations $\boldsymbol{d}$
4. $\boldsymbol{d} \rightarrow$ duration-based linguistic features $\boldsymbol{l}^{(d)}$
5. $\boldsymbol{l}^{(b)}$ & $\boldsymbol{l}^{(d)} \rightarrow$ DNN-predicted per-frame static & dynamic feature distributions
6. MLPG with postfiltering to generate acoustic parameter trajectories

Neural network design

- 6 hidden layers
  - 256/1024 units each (duration/acoustic model)
  - tanh activation function
- MDN parameter output layer
  - Softmax outputs for weights
  - Linear outputs for means
  - Logarithmic outputs with variance flooring for diagonal covariances

Implementation

Deep MDN code courtesy of Zhizheng Wu

- Setup largely follows Zen and Senior (2014)
  - Random initialisation
  - Trained until development set likelihood peaked
- GPU implementation with Python + Theano
  - Batched stochastic gradient descent
  - β-estimation straightforward to implement
    - Trained as refinements of less robust models (e.g., MLE)
  - Log-sum-exp trick for safe GMM likelihood evaluation

What now?

Current research directions:

- LSTMs rather than DNNs
- Robust acoustic modelling
- New datasets

Journal paper in preparation