Neuroevolution for sound event detection in real life audio: A pilot study

Christian Kroos & Mark D. Plumbley

Centre for Vision, Speech and Signal Processing (CVSSP),
University of Surrey, UK

making sense of sounds

UNIVERSITY OF SURREY

Background

Neuroevolution methods:

- Evolution of artificial neural networks using genetic algorithms
- Evolving only weights or weights together with topology (TWEANNs, Topology and Weight Evolving Artificial Neural Networks)
- Direct or indirect encoding

Neuroevolution

- Used mostly in simulated robotics and automated computer game play
- Tricky but small problems, e.g., simulated robot in deceptive maze
- Rarely seen in classification or detection tasks
- No learning, no gradients need to be computed
- Glorified random search?

Motivation

Potential application in robotics

- Small autonomous robots
- Emulating e.g. airborne insects
- General purpose audio processing system might be available
- Many tiny classifiers/detectors needed
- (Honey bee brain: 1 mm^3 - 960,000 neurons)

Menzel, R., & Giurfa, M. (2001). Cognitive architecture of a mini-brain: the honeybee. *Trends in Cognitive Sciences*, 5(2), 62-71.

Inspiration

Ecological psychology (James J. Gibson):

- Anti-representational stance with regard to all perception
- The perceptual processing of animals (including humans) tunes in on relevant features in the environment through adaptation
- Only key features need to be tracked, no representation of the environment needed
- Perceptual system operates as a dynamical system

Research question & hypotheses

Inquiry:

- Can neuroevolution methods be used to develop parsimonious but still accurate audio event classifiers/detectors?

Hypotheses:

- Successful small (but probably deep) neural networks can be evolved with reasonable computational effort
- Performance not able to match deep learning (DNN, RNN, CNN, LSTM, BGRU, …)

How to test hypotheses?

- Performance in DCASE challenge, Task 3, Sound event detection in real life audio
- Unmodified real world recording as e.g. a robot would experience it
- Polyphonic
- And the best deep learning systems as competitors

Method

Feature extraction

Wavelet-based scattering transform

Mallat, S. (2012) “Group invariant scattering,” *Communications on Pure and Applied Mathematics,* vol. 65, no. 10, pp. 1331– 1398

- Window length of 372 ms; one channel only
- Default audio settings 520 coefficients

K-means clustering along the coefficients

- Time samples treated as input variables, coefficients as observations
- Centroids used as the required dimension-reduced data representation
- k = 17

Feature extraction

Channel differences

- From short-term FFT spectrum (window length = 372 ms)
- Averaged over all spectral coefficients

Resulting dimensionality

- 17 + 1

Downsampling

- Both feature types downsampled to 1 Hz

Foundation: NEAT

- NEAT algorithm (NeuroEvolution of Augmenting Topologies)
  - Stanley, K. O. and Miikkulainen, R. (2002) “Evolving neural networks through augmenting topologies, ”*Evolutionary Computation*, vol. 10, no. 2, pp. 99–127.
  - Stanley, K. O. and Miikkulainen, R. (2004) “Competitive coevolution through evolutionary complexification,” *Journal of Artificial Intelligence Research*, vol. 21, pp. 63–100.
- Direct encoding
- Starts with minimal network
- Grows the networks using crossover and mutations

Foundation: NEAT

- Fitness determination via loss function
- Fittest networks (typically top 80%) either
  - enter next generation unchanged (elitism) or
  - become parents of next generation (crossover) or
  - enter next generation in a mutated form

Foundation: NEAT

NEAT protects more complex networks from early elimination through 'speciation':

- Different networks (both topology and weights count) are assigned to different species
- Have to compete only within species
- Historical markers allow avoiding topology analysis

New algorithm: J-NEAT

- Original NEAT not suitable for classification and event detection with larger data sets
- Modifications
  - that adapted aspects of the original NEAT algorithm within its general paradigm
  - that extended NEAT and changed its nature
- Only a selected few of the latter described in the following

New algorithm: J-NEAT

- Network layers are determined (fast recursive algorithm)
- Activation function also subject to mutation
- Break complex classification/detection problem into smaller partial problems: **cooperative co-evolution**
  - Three populations evolve simultaneously
  - Each population → one third of the input at each sample point
  - Cooperation: Ad-hoc formed triplets of networks that together deliver classification output for each sample → joint fitness

Procedure and parameters

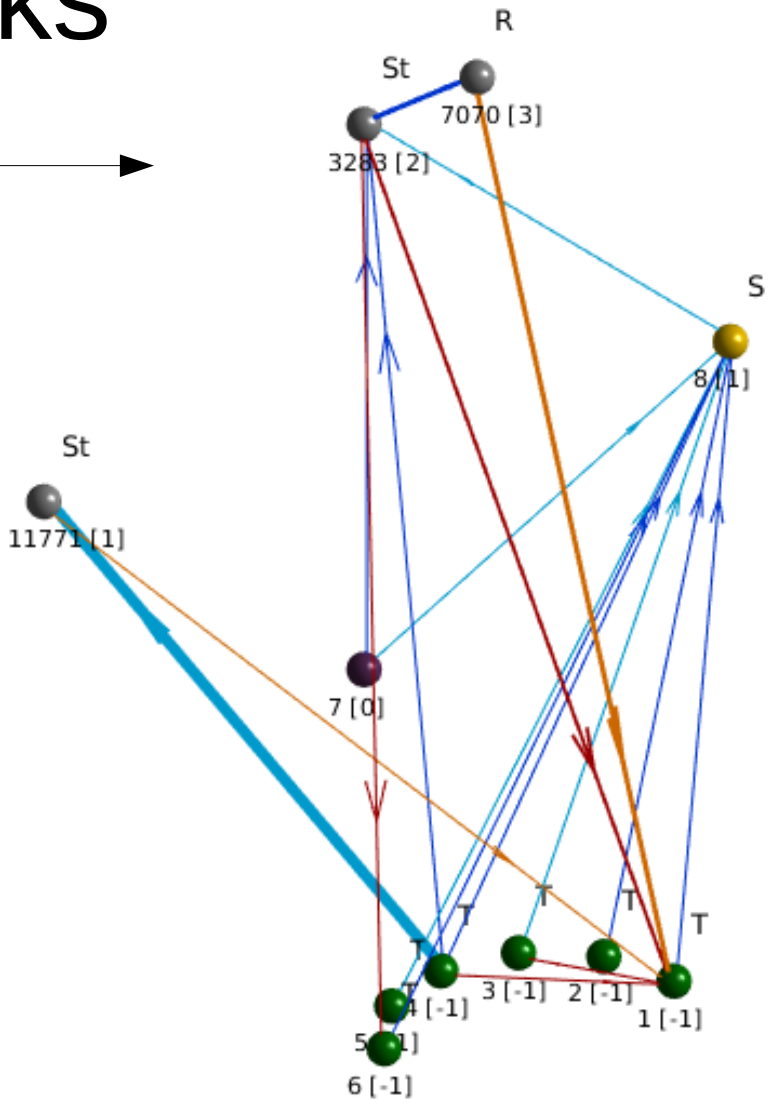
- 400 individuals per population
- 500 generations
- 250 constraint randomly selected input samples simultaneously evaluated at each step (~ mini-batch)
- 44 consecutive steps evaluated (→time series, recurrent nodes)
- Also tested a version without cooperative co-evolution
- And for comparison: Minimal feed-forward network (learning vs evolving)

Results

Networks

'people walking' detector (1):

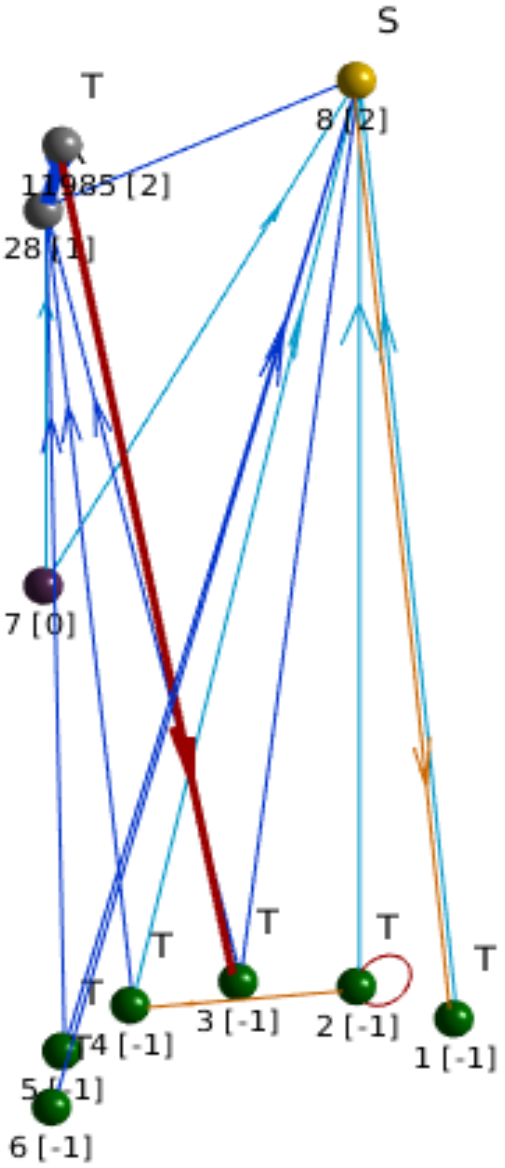
- Blue lines: forward connections (light blue negative, dark blue positive weights)
- Red/orange lines: recurrent connections (orange negative, red positive weights).
- Line thickness: relative magnitude of the weight
- Input nodes: green,
- Bias nodes: dark purple
- Output node: yellow.
- Letters coding the activation function, where S = sigmoid, St = a steeper sigmoid function used in NEAT, T = tanh, I = identity, R = rectified linear, RL = leaky rectified linear and P = softplus.



Networks

'people walking' detector (2):

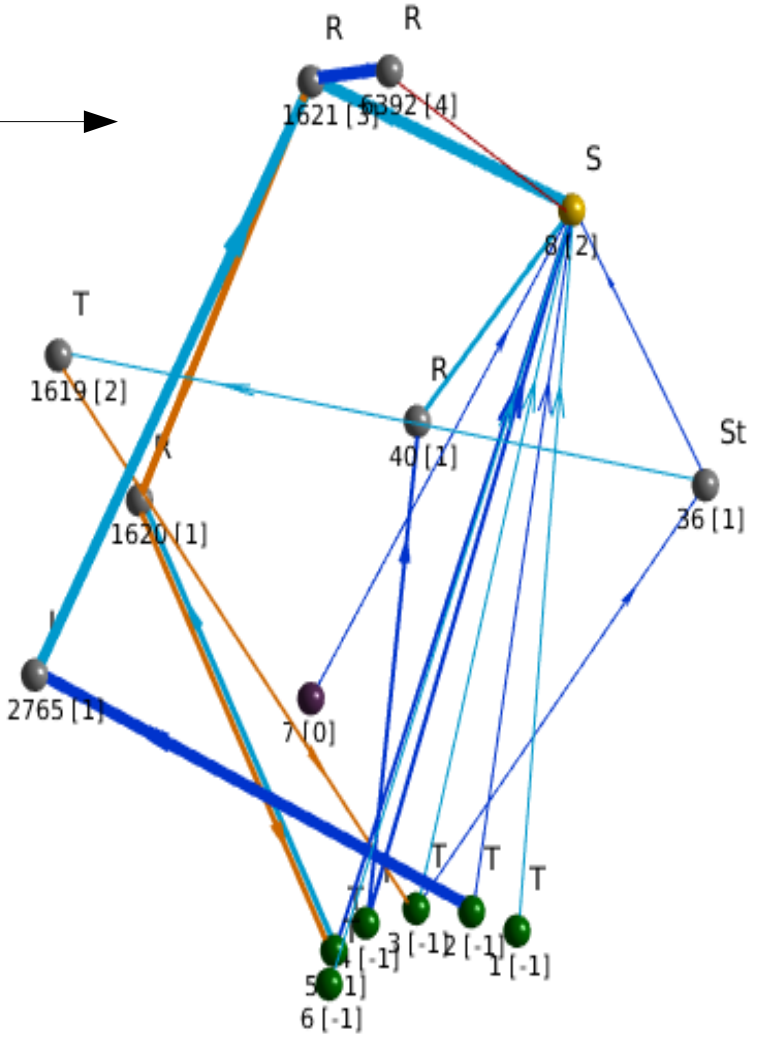
- Blue lines: forward connections (light blue negative, dark blue positive weights)
- Red/orange lines: recurrent connections (orange negative, red positive weights).
- Line thickness: relative magnitude of the weight
- Input nodes: green,
- Bias nodes: dark purple
- Output node: yellow.
- Letters coding the activation function, where S = sigmoid, St = a steeper sigmoid function used in NEAT, T = tanh, I = identity, R = rectified linear, RL = leaky rectified linear and P = softplus.



Networks

'people walking' detector (3):

- Blue lines: forward connections (light blue negative, dark blue positive weights)
- Red/orange lines: recurrent connections (orange negative, red positive weights).
- Line thickness: relative magnitude of the weight
- Input nodes: green,
- Bias nodes: dark purple
- Output node: yellow.
- Letters coding the activation function, where S = sigmoid, St = a steeper sigmoid function used in NEAT, T = tanh, I = identity, R = rectified linear, RL = leaky rectified linear and P = softplus.



Ensemble evolution

Best 'people walking' detector per generation:

[Show movie]

Development data

Four fold evaluation treated as single experiment

Method	Segment ER	Segment F1	Event ER	Event F1
Baseline	0.72	51.40	3.30	6.74
J-NEAT ensemble	0.73	49.24	1.46	6.46
J-NEAT plain	0.72	50.55	1.37	5.66
Single-layer FFN	0.69	56.47	1.40	5.85

Challenge data

Method	Segment ER	Segment F1	Rank ER	Rank F1
Baseline	0.936	42.8	19	8
J-NEAT ensemble	0.898	44.9	15	**1**
J-NEAT plain	0.891	41.6	14	12
Single-layer FFN	1.014	43.8	20	3

Discussion

- First hypothesis confirmed:
  - ✗ J-NEAT was able to evolve operational classifiers/detectors
- Second hypothesis not confirmed
  - ✗ The evolved systems could indeed match the performance of the much larger deep neural networks
- Computational effort: All four training folds and the full data set for the challenge → ~48 hours on a 4-core CPU of Dell Linux desktop machine with only 3 parallel workers

Future research

- Systematic hyper-parameter testing
- Evaluation of variability
- Interleave evolutionary phases (weights and topology) and learning phases (weights)

Conclusion

It works!

LVA ICA 2018

14th International Conference on Latent Variable Analysis and Signal Separation

July 2-6, 2018 **University of Surrey, Guildford, UK**

Paper deadline: January 15, 2018

cvssp.org/events/lva-ica-2018