Neural network-based seismic event detector: an application to swarm-like earthquake in West Bohemia and South-west Iceland

J. Doubravová and J. Horálek

Workshop CzechGeo/EPOS, 22.11.2017



EUROPEAN UNION European Structural and Investment Funds Operational Programme Research, Development and Education



• □ ▶ • • □ ▶ • • □ ▶ •

J. Doubravová and J. Horálek Neural network-based seismic event detector:

Outline



Artificial neural networks - basics

Data and method 2

- WEBNET data
- Single Layer Recurrent Neural Network
- Training of the SLRNN

3 Training results

Application to Reykjanet



continual data are not suitable for direct manual processing

- high quality detection needed for
 - manual processing, i.e. we need minimum number of false alarms
 - automatic processing detection of weak events
- artificial neural network can extract useful information
- forward computation of trained network is fast



- continual data are not suitable for direct manual processing
- high quality detection needed for
 - manual processing, i.e. we need minimum number of false alarms
 - automatic processing detection of weak events
- artificial neural network can extract useful information
- forward computation of trained network is fast

• • • • •



- continual data are not suitable for direct manual processing
- high quality detection needed for
 - manual processing, i.e. we need minimum number of false alarms
 - automatic processing detection of weak events
- artificial neural network can extract useful information
- forward computation of trained network is fast

A 3 b



- continual data are not suitable for direct manual processing
- high quality detection needed for
 - manual processing, i.e. we need minimum number of false alarms
 - automatic processing detection of weak events
- artificial neural network can extract useful information
- forward computation of trained network is fast



- continual data are not suitable for direct manual processing
- high quality detection needed for
 - manual processing, i.e. we need minimum number of false alarms
 - automatic processing detection of weak events
- artificial neural network can extract useful information
- forward computation of trained network is fast



- continual data are not suitable for direct manual processing
- high quality detection needed for
 - manual processing, i.e. we need minimum number of false alarms
 - automatic processing detection of weak events
- artificial neural network can extract useful information
- forward computation of trained network is fast

Neural network

artificial neural networks has been inspired by biological neural networks

- acts as an interface between the organism and environment, reacts to inner and outer stimuli
- sensors (=receptors), information is spread through the network to effectors (muscles, glands)
- in cerebral cortex 15-33 billion neurons, each neuron connected to up to 5000 other neurons

→ < Ξ → <</p>

Neural network

- artificial neural networks has been inspired by biological neural networks
- acts as an interface between the organism and environment, reacts to inner and outer stimuli
- sensors (=receptors), information is spread through the network to effectors (muscles, glands)
- in cerebral cortex 15-33 billion neurons, each neuron connected to up to 5000 other neurons

• • • • •

Neural network

- artificial neural networks has been inspired by biological neural networks
- acts as an interface between the organism and environment, reacts to inner and outer stimuli
- sensors (=receptors), information is spread through the network to effectors (muscles, glands)
- in cerebral cortex 15-33 billion neurons, each neuron connected to up to 5000 other neurons

Neural network

- artificial neural networks has been inspired by biological neural networks
- acts as an interface between the organism and environment, reacts to inner and outer stimuli
- sensors (=receptors), information is spread through the network to effectors (muscles, glands)
- in cerebral cortex 15-33 billion neurons, each neuron connected to up to 5000 other neurons

- body, dendrites (inputs), axon and synaptic terminals (outputs)
- synaptic weight between dendrite and axon (inhibition or excitation)
- synapsis can be build (learning) or disconnected (forgetting)
- neuron generates an electrical impulse if the activity of dendrites is strong enough (information propagates)



- body, dendrites (inputs), axon and synaptic terminals (outputs)
- synaptic weight between dendrite and axon (inhibition or excitation)
- synapsis can be build (learning) or disconnected (forgetting)
- neuron generates an electrical impulse if the activity of dendrites is strong enough (information propagates)



- body, dendrites (inputs), axon and synaptic terminals (outputs)
- synaptic weight between dendrite and axon (inhibition or excitation)
- synapsis can be build (learning) or disconnected (forgetting)
- neuron generates an electrical impulse if the activity of dendrites is strong enough (information propagates)



- body, dendrites (inputs), axon and synaptic terminals (outputs)
- synaptic weight between dendrite and axon (inhibition or excitation)
- synapsis can be build (learning) or disconnected (forgetting)
- neuron generates an electrical impulse if the activity of dendrites is strong enough (information propagates)





- n real inputs x = dendrites, threshold input x₀ = 1
- weights w = synaptic weights, bias w₀ = -h threshold
- activation function $V=g\left(egin{array}{c}{
 m '}{
 m '}{
 m v}_{i}
 ight)$



Artificial neuron



 n real inputs x = dendrites, threshold input x₀ = 1

- weights w = synaptic weights, bias $w_0 = -h$ threshold
- activation function $V = g\left(\sum_{i=0}^{n} w_i x_i\right)$





- n real inputs x = dendrites, threshold input x₀ = 1
- weights w = synaptic weights, bias $w_0 = -h$ threshold
- activation function $V = g\left(\sum_{i=0}^{n} w_i x_i\right)$





- n real inputs x = dendrites, threshold input x₀ = 1
- weights w = synaptic weights, bias $w_0 = -h$ threshold







- n real inputs x = dendrites, threshold input x₀ = 1
- weights w = synaptic weights, bias $w_0 = -h$ threshold
- activation function $V = g\left(\sum_{i=0}^{n} w_i x_i\right)$





- n real inputs x = dendrites, threshold input x₀ = 1
- weights w = synaptic weights, bias $w_0 = -h$ threshold
- activation function $V = g\left(\sum_{i=0}^{n} w_i x_i\right)$



Artificial neural network

• for more complex problems

• typical application: classification, pattern recognition, regression



Artificial neural network

- for more complex problems
- typical application: classification, pattern recognition, regression



WEBNET data

Outline



Artificial neural networks - basics

2 Data and method • WEBNET data

- Single Layer Recurrent Neural Network
- Training of the SLRNN

Application to Reykjanet

A - E - A - E

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

WEBNET data





< /₽ > < E >

< ∃→

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

WEBNET data





< /₽ > < E >

< ∃→

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

WEBNET data





WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

WEBNET data





э

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

WEBNET data





э

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

WEBNET data





Single Layer Recurrent Neural Network





Artificial neural networks - basics

2 Data and method

- WEBNET data
- Single Layer Recurrent Neural Network
- Training of the SLRNN

Application to Reykjanet

A - E - A - E

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

SLRNN - architecture

- single layer network
- outputs used as inputs in the next time step = recurrence, memory
- various delays D₁...D_d



3

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

SLRNN - architecture

- single layer network
- outputs used as inputs in the next time step = recurrence, memory
- various delays $D_1..D_d$



э

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

SLRNN - architecture

single layer network

 outputs used as inputs in the next time step = recurrence, memory

• various delays $D_1..D_d$



э

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

SLRNN - architecture

- single layer network
- outputs used as inputs in the next time step = recurrence, memory

• various delays $D_1..D_d$



-
WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

SLRNN - architecture

- single layer network
- outputs used as inputs in the next time step = recurrence, memory
- various delays $D_1..D_d$



-

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

SLRNN - architecture

- single layer network
- outputs used as inputs in the next time step = recurrence, memory
- various delays $D_1..D_d$



-

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

SLRNN - input signals

vertical and horizontal component.

 STA/LTA in 9 half-octave frequency bands



< ロ > < 同 > < 回 > < 回 >



WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

SLRNN - input signals

- vertical and horizontal component
- STA/LTA in 9 half-octave frequency bands



< E



J. Doubravová and J. Horálek Neural network-based seismic event detector:

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

SLRNN - input signals

- vertical and horizontal component
- STA/LTA in 9 half-octave frequency bands



< □ > < 同 > < 回 > <</p>

< ∃→



J. Doubravová and J. Horálek Neural network-based seismic event detector:

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

SLRNN - input signals

- vertical and horizontal component
- STA/LTA in 9 half-octave frequency bands



A (1) < (1) < (1) </p>

-



WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

SLRNN - input signals

- vertical and horizontal component
- STA/LTA in 9 half-octave frequency bands



A (1) < (1) < (1) </p>

-



WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

- 8 neurons, 18 inputs, 3 outputs (event, P-wave, S-wave)
- delay 1, 2, 4, and 8 samples
- each neuron 18+(4x8)+1=51 inputs
- 8x51=408 weights to adjust



WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

- 8 neurons, 18 inputs, 3 outputs (event, P-wave, S-wave)
- delay 1, 2, 4, and 8 samples
- each neuron 18+(4x8)+1=51 inputs
- 8x51=408 weights to adjust



WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

- 8 neurons, 18 inputs, 3 outputs (event, P-wave, S-wave)
- delay 1, 2, 4, and 8 samples
- each neuron 18+(4x8)+1=51 inputs
- 8x51=408 weights to adjust



WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

SLRNN - architecture

- 8 neurons, 18 inputs, 3 outputs (event, P-wave, S-wave)
- delay 1, 2, 4, and 8 samples
- each neuron 18+(4x8)+1=51 inputs

• 8x51=408 weights to adjust



WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

SLRNN - architecture

- 8 neurons, 18 inputs, 3 outputs (event, P-wave, S-wave)
- delay 1, 2, 4, and 8 samples
- each neuron $18+(4\times8)+1=51$ inputs

• 8x51=408 weights to adjust



WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

- 8 neurons, 18 inputs, 3 outputs (event, P-wave, S-wave)
- delay 1, 2, 4, and 8 samples
- each neuron $18+(4\times8)+1=51$ inputs
- 8×51=408 weights to adjust



WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

- 8 neurons, 18 inputs, 3 outputs (event, P-wave, S-wave)
- delay 1, 2, 4, and 8 samples
- each neuron $18+(4\times8)+1=51$ inputs
- 8×51=408 weights to adjust



Training of the SLRNN

Outline



Artificial neural networks - basics

2 Data and method

- WEBNET data
- Single Layer Recurrent Neural Network
- Training of the SLRNN

Application to Reykjanet

A 3 b

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

Training

- supervised learning: adjusting weights w_{ij} to get the best fit with desired output
- we define the desired outputs for training data



・ 同 ト ・ ヨ ト ・ ヨ

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

Training

- supervised learning: adjusting weights w_{ij} to get the best fit with desired output
- we define the desired outputs for training data



A (1) > (1) > (1)

-

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

Training data

- earthquake swarm 2008 (events-positive examples) and year 2010 (disturbances-negative examples)
- events: different magnitudes, locations, focal mechanisms
- disturbances: quarry blasts, regional or teleseismic events, wind, storms

・ 同 ト ・ 三 ト ・

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

Training data

- earthquake swarm 2008 (events-positive examples) and year 2010 (disturbances-negative examples)
- events: different magnitudes, locations, focal mechanisms
- disturbances: quarry blasts, regional or teleseismic events, wind, storms

・ 同 ト ・ 三 ト ・

WEBNET data Single Layer Recurrent Neural Network Training of the SLRNN

Training data

- earthquake swarm 2008 (events-positive examples) and year 2010 (disturbances-negative examples)
- events: different magnitudes, locations, focal mechanisms
- disturbances: quarry blasts, regional or teleseismic events, wind, storms

3.1

Sensitivity and specificity

- sensitivity (true positive rate TPR) TPR= TP/TP+FN = identified events / all events
- specificity (true negative rate TNR) TNR=<u>TN</u> = rejected disturbances / all disturbances
- ROC (Receiver operation characteristic) a relation between the sensitivity and the specificity

- A 🗐 🕨 - A

Sensitivity and specificity

- sensitivity (true positive rate TPR) TPR= TP/TP+FN = identified events / all events
- specificity (true negative rate TNR) TNR= TN TN+FP = rejected disturbances / all disturbances
- ROC (Receiver operation characteristic) a relation between the sensitivity and the specificity

• • = • • = •

Sensitivity and specificity

- sensitivity (true positive rate TPR) TPR= TP/TP+FN = identified events / all events
- specificity (true negative rate TNR) TNR= TN TN+FP = rejected disturbances / all disturbances
- ROC (Receiver operation characteristic) a relation between the sensitivity and the specificity

ROC diagram



э

ROC diagram



J. Doubravová and J. Horálek Neural network-based seismic event detector:

э

False detection

- blind test on data from 2011, single station detection only
- false detections are often weak events
- but many detections could not be verified

A 3 b



- blind test on data from 2011, single station detection only
- false detections are often weak events
- but many detections could not be verified

3 N

False detection

- blind test on data from 2011, single station detection only
- false detections are often weak events
- but many detections could not be verified

Undetected events - examples events $M_L = 2,3$ and $M_L = 2,2$ hidden in the coda of $M_L = 3,8$



J. Doubravová and J. Horálek Neural network-based seismic event detector:

Undetected events - examples undetected event $M_L = -0.3$ on station with high noise



J. Doubravová and J. Horálek Neural network-based seismic event detector:

-

Undetected events - examples

missing detection of event $M_L = 0, 2$ on station with low P- and S-wave amplitudes



J. Doubravová and J. Horálek Neural network-based seismic event detector:

Training results

• SLRNN architecture is suitable for local event detection

- individual training must not be better than joint training
- for a good reliability the data throughout the network must be combined
 - reject false detections
 - eliminate undetected events

• • • • •

Training results

- SLRNN architecture is suitable for local event detection
- individual training must not be better than joint training
- for a good reliability the data throughout the network must be combined
 - reject false detections
 - eliminate undetected events

• • • • •

Training results

- SLRNN architecture is suitable for local event detection
- individual training must not be better than joint training
- for a good reliability the data throughout the network must be combined
 - reject false detections
 - eliminate undetected events

Training results

- SLRNN architecture is suitable for local event detection
- individual training must not be better than joint training
- for a good reliability the data throughout the network must be combined
 - reject false detections
 - eliminate undetected events

Training results

- SLRNN architecture is suitable for local event detection
- individual training must not be better than joint training
- for a good reliability the data throughout the network must be combined
 - reject false detections
 - eliminate undetected events
Reykjanet network

• southwest Iceland, Reykjanes peninsula

- 15 off-line broadband stations
- network configuration, number of stations, swarm activity similar to WB



Reykjanet network

- southwest Iceland, Reykjanes peninsula
- 15 off-line broadband stations
- network configuration, number of stations, swarm activity similar to WB



Reykjanet network

- southwest Iceland, Reykjanes peninsula
- 15 off-line broadband stations
- network configuration, number of stations, swarm activity similar to WB



Neural network

• the best network of joint training for WEBNET

- coincidence implemented
- detection on at least 6 stations in a time window (0,8s) required to define an EVENT

• • • • •

Neural network

- the best network of joint training for WEBNET
- coincidence implemented
- detection on at least 6 stations in a time window (0,8s) required to define an EVENT

A 3 3 4

Neural network

- the best network of joint training for WEBNET
- coincidence implemented
- detection on at least 6 stations in a time window (0,8s) required to define an EVENT

3 N

Selected data: 4 swarms



J. Doubravová and J. Horálek

Neural network-based seismic event detector:



SIL - IMO catalog - manually revised automatic locations lcelandic network

- Antelope automatic catalog by Antelope from Reykjanes data (B. Růžek)
- PePiN automatic locations by PePiN (T. Fischer)
- ANN event detections (no locations) using SLRNN trained for WEBNET
- maximum magnitude $M_L = 2,3$

周 ト イ ヨ ト イ ヨ



- SIL IMO catalog manually revised automatic locations lcelandic network
- Antelope automatic catalog by Antelope from Reykjanes data (B. Růžek)
- PePiN automatic locations by PePiN (T. Fischer)
- ANN event detections (no locations) using SLRNN trained for WEBNET
- maximum magnitude $M_L = 2,3$

4 冊 ト 4 三 ト 4 三 ト



- SIL IMO catalog manually revised automatic locations lcelandic network
- Antelope automatic catalog by Antelope from Reykjanes data (B. Růžek)
- PePiN automatic locations by PePiN (T. Fischer)
- ANN event detections (no locations) using SLRNN trained for WEBNET
- maximum magnitude $M_L = 2,3$

4 冊 ト 4 三 ト 4 三 ト



- SIL IMO catalog manually revised automatic locations lcelandic network
- Antelope automatic catalog by Antelope from Reykjanes data (B. Růžek)
- PePiN automatic locations by PePiN (T. Fischer)
- ANN event detections (no locations) using SLRNN trained for WEBNET
- maximum magnitude M_L = 2,3

• • = • • = •



- SIL IMO catalog manually revised automatic locations lcelandic network
- Antelope automatic catalog by Antelope from Reykjanes data (B. Růžek)
- PePiN automatic locations by PePiN (T. Fischer)
- ANN event detections (no locations) using SLRNN trained for WEBNET
- maximum magnitude $M_L = 2,3$

• • = • • = •

Number of events



・ 同 ト ・ ヨ ト ・ ヨ ト

э

Comparison of individual events - March 2015



- Pepin and SIL sorted by the magnitude
- Antelope without magnitudes, sorted in time

Comparison of individual events - March 2015



- Pepin and SIL sorted by the magnitude
- Antelope without magnitudes, sorted in time

Smallest events

10	7.000	- 1.44e-07 r
2	and a set of the set o	-1 528-07
нı		2 966-07 #
Ŀ		
٢z	a a fei george and he was a second a second as a fei george and a second as	-4.792-07
= 1	ĸŧġġġġġġġġġġġġġġġġġġġġġġġġġġġġġġġġġġġġ	-3.462-07
E	with weathing and free how for the standard of the standard and the standard of the standard and	WW -3.11E-07
2	1	-4.69e-08
1	watered watered watered watered watered watered watered to a second bear and the second and	6.34e-08 r
E	wal was done was done and a second or a second water and her second a day of the ball of the second s	5.66e-08 m
1		-7.87e-07
v		-1.08e-06
		-6.36e-07
= `		1.81e-07 r
2		3.46e-07 r
Ľ	ให้เป็นขึ้นขึ้นในที่เป็นขึ้นในขึ้นในขณะที่มีขณะจะสายในกลางจากให้เข้าจะสายให้เห็นของได้ได้ไปหลายที่จะสายจะที่ได้ ให้เป็นที่มีกลายที่ได้มีกลายที่ได้สายจะจะสายในกลางจากไปที่สายจากได้ได้จะการที่ได้ได้ได้ได้ได้ได้ได้ได้ได้ได้ได้	2.72e-07 r
_ '		-1.96e-07
2		-6.27e-07
1		-9 110-07
L		
7	n determine the destruction of the	-1.26e-07
. 1	x/bit/s/y/y/bit/git/git/s/bit/second/bit/second/bit/second/bit/git/git/git/git/git/git/git/git/git/g	AM -1.985-01
6	marghaf ar an air air an	-1.28e-07
۲,	and the second	-2e-06 m/s
н		-1.34e-06
1		-1.35e-06
L		

J. Doubravová and J. Horálek Neural network-based seismic event detector:



• SLRNN is fast and efficient algorithm for event detection

- in case of consistent readings of P and S phases individual training performs better
- for missing readings, joint training is helpful
- station coincidence solved both problems; undetected events and false alarm
- network trained for West Bohemia works very well for Reykjanes

• • • • •



- SLRNN is fast and efficient algorithm for event detection
- in case of consistent readings of P and S phases individual training performs better
- for missing readings, joint training is helpful
- station coincidence solved both problems; undetected events and false alarm
- network trained for West Bohemia works very well for Reykjanes

• • • • •



- SLRNN is fast and efficient algorithm for event detection
- in case of consistent readings of P and S phases individual training performs better
- for missing readings, joint training is helpful
- station coincidence solved both problems; undetected events and false alarm
- network trained for West Bohemia works very well for Reykjanes

A 3 A



- SLRNN is fast and efficient algorithm for event detection
- in case of consistent readings of P and S phases individual training performs better
- for missing readings, joint training is helpful
- station coincidence solved both problems; undetected events and false alarm
- network trained for West Bohemia works very well for Reykjanes



- SLRNN is fast and efficient algorithm for event detection
- in case of consistent readings of P and S phases individual training performs better
- for missing readings, joint training is helpful
- station coincidence solved both problems; undetected events and false alarm
- network trained for West Bohemia works very well for Reykjanes

.

Thank you for your attention !

Sincere thanks to members of Webnet group for providing data, and to CzechGeo/EPOS for supporting our networks

3 N