Tuning and performance monitoring: Re-tuning artificial neural networks in StaPro

Radek Hofman, Elena Tomuta, Ronan Le Bras

International Data Centre
Preparatory Commission for the Comprehensive
Nuclear-Test-Ban Treaty Organization

US NDC-IDC Bilateral Meeting
8th March 2018
VIC E0541

Presentation outline

- Introduction — neural networks in StaPro
- Algorithm and its inputs
- Why re-tunnig effort?
- First results
- Conclusion and future plans

Neural networks in StaPro

CTBTO PREPARATORY COMMISSION

- Neural networks (NNs) are used in StaPro for initial wave type (iwt) determination for three-component (3C) stations
- The purpose is to assign an initial wave type to each arrival: noise (**N**), regional S (**regS**), regional P (**regP**) or telesesmic (**tele**)
- Given iwt, StaPro further refines the wave type using a Bayesian model:

iwt	subcategories
noise	—
regional S	Sn, Lg, Rg, Sx
regional P	Pn, Pg, Px
teleseismic	P, tx

- This gets saved in IDCX.ARRIVAL table as IPHASE

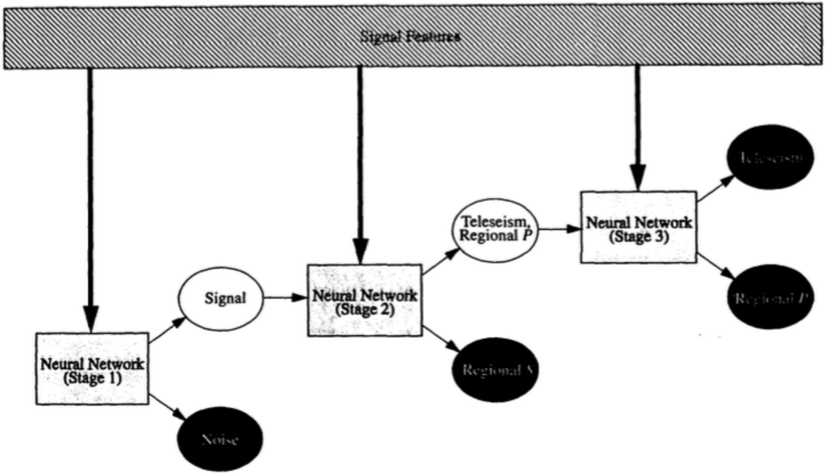
Neural networks in StaPro

CTBTO PREPARATORY COMMISSION

- The system comprises of a cascade of three binary classifiers
  - Step 1: the system tries to distinguish between noise **N** and a signal {**regS**, **regP**, **tele**}
  - Step 2: if signal: is it **regS** or {**regP**, **tele**}?
  - Step 3: if {**regP**, **tele**}: is it regional **P** or **tele**?
- All three classifiers are implemented as a multilayer perceptron (neural network)

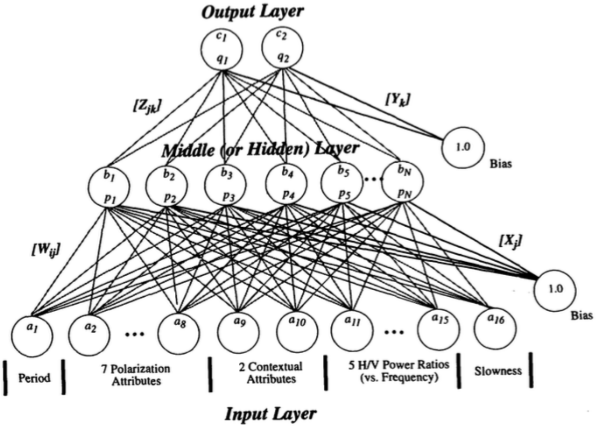
- WANG, J., Adaptive training of neural networks for automatic seismic phase identification. Monitoring the Comprehensive Nuclear-Test-Bant Treaty: Data Processing and Infrasound (2002): 1021-1041.
- SERENO, T. and PATNAIK, G., Initial Wave-type Identification with Neural Networks and its Contribution to Automated Processing in IMS Version 3.0, Technical Report, SAIC-93/1219, 1993

Cascade of three binary classifiers



Multilayer perceptron topology

All three NNs in the cascade have the same topology: input layer of size 15, hidden layer of size 6, output layer of size 2, all dense with sigmoid activation (in total 110 parameters)



Inputs (classification features)

CTBTO PREPARATORY COMMISSION

#	Feature	Description	Source	Norm.
1	*period*	Dominant period of the detected phase	ARRIVAL	
2	*rect*	Signal rectilinearity	APMA	
3	*plans*	Signal planarity	APMA	
4	$inang_1$	Long-axis incidence angle	APMA	$/ = 90$
5	$inang_3$	Short-axis incidence angle	APMA	$/ = 90$
6	*hmxmn*	Ratio of the maximum to minimum horizontal amplitude	APMA	$\log_{10}(\cdot)$
7	*hvratp*	Ratio of horizontal-to-vertical power	APMA	$\log_{10}(\cdot)$
8	*hvrat*	Similar to *hvratp*, measured at the time of the maximum 3C amplitude	APMA	$\log_{10}(\cdot)$
9	$N_{after} - N_{before}$	Difference between the number of arrivals before and after the arrival within $\pm 60s$	on the fly	$/ = 10$
10	$T_{after} - T_{before}$	Mean time difference between the arrivals before and after within $\pm 60s$	on the fly	$/ = 100$
11	$htov_1$	Horizontal-to-vertical power ratio in an octave frequency band centered at 0.25 Hz	AMP3C	$\log_{10}(\cdot)$
12	$htov_2$	Horizontal-to-vertical power ratio in an octave frequency band centered at 0.5 Hz	AMP3C	$\log_{10}(\cdot)$
13	$htov_3$	Horizontal-to-vertical power ratio in an octave frequency band centered at 1.0 Hz	AMP3C	$\log_{10}(\cdot)$
14	$htov_4$	Horizontal-to-vertical power ratio in an octave frequency band centered at 2.0 Hz	AMP3C	$\log_{10}(\cdot)$
15	$htov_5$	Horizontal-to-vertical power ratio in an octave frequency band centered at 4.0 Hz	AMP3C	$\log_{10}(\cdot)$

A brief history

- According to *Sereno and Patnaik, (1993)*, the iwt classification system was created in early 1990s (as en extension of a two class system P vs. S) and became part of IMS v3 (IMS stands here for *Intelligent Monitoring System*)
- Station-specific weights were trained using analyst-reviewed data for two 3C stations which had enough data
- For the rest of stations "average" weights yielded by training using data from multiple stations were produced
- Around 2002 the weights have been re-trained using data from STKA station, see *(J. Wang, 2002)*
- Since then, these weights are used as weights for all 3C stations in IMS (IMS stands here for *International Monitoring System*)

Why re-training effort?

- Is it correct to use weights derived using a single station data globally?
- Can we do better if the weights were derived for each station individually using station specific data?
- Nowadays, there is much more analyst-reviewed data than before $\sim$ 20 years
- Re-training means just a change in configuration of StaPro, if we keep the original topology of NNs then no code changes are needed

Ingredients needed for successful re-training

1. **Train/Test Datasets** - For neural network it holds true that the more, the better
2. **Back-propagation algorithm** - The algorithm for NN retraining
   - Unfortunately, the original C code used for training we do not have. However, the generic nature of NNs and a domain knowledge allows us (hopefully) to use an arbitrary NN library for the training
3. **A way to translate weights from re-training into StaPro configuration files**
   - We need to produce and alternate weights file and replace with it the original one in StaPro
4. **A simulated pipeline which can be run with the alternate weight file**
   - This gives us an alternate IDCX.ARRIVAL table with new IPHASE
5. **Evaluation script which compares the success rate of the new classification vs. the old one**

Ingredients needed for successful re-training

CTBTO PREPARATORY COMMISSION

1. **Train/Test Datasets** - We chose to start with URZ station
   - It is one of the most data rich stations and its current IWT performance is around 50%
2. **Back-propagation algorithm** - We use Python library Keras backed by Theano
3. **A way to translate weights from re-training into StaPro configuration files**
   - We analysed the C code and created a script puting the new weights into the weights-file template in the right order
4. **A simulated pipeline which can be run with the alternate weight file**
   - We are able to run StaPro using the Station Tuning framework
5. **Evaluation script which compares the success rate of the new classification vs. the old one**
   - This still needs to be done but we plan to match arrivals in the new table with those in REB using proximity of arrival attributes

URZ dataset

CTBTO PREPARATORY COMMISSION

We also rely on analyst-reviewed data. Our URZ dataset is comprised of:

- 27399 noise phases, 9133 regS phases, 9133 regP phases and 9133 tele phases (we aimed for ratio 3:1:1:1 in order to have a balanced dataset)
  - In the dataset used by Wang, (2002) was $\sim$ 1500 phases for all 4 classes
- Signal phases are selected randomly from all automatic arrivals associated to REB
- Noise phases are selected randomly from all automatic arrivals classified as noise and not associated to REB
- The dataset is split on train and test data as 3:1

URZ — preliminary results

Confusion matrix of test data

Re-trained weights				
	N	P	S	T
N	**5687**	238	609	292
P	346	**1687**	14	473
S	653	5	**1651**	374
T	164	318	74	**1115**
accuracy **75%**				

URZ — preliminary results

- Comparison of classification capabilities of regS, regP and tele classes for original and new weights

Old weights				
	N	P	S	T
N	—	337	829	380
P	—	**1655**	170	540
S	—	45	**1312**	351
T	—	211	37	**983**
accuracy **57.66%**				

Re-trained weights				
	N	P	S	T
N	—	238	609	292
P	—	**1687**	14	473
S	—	5	**1651**	374
T	—	318	74	**1115**
accuracy **65.01%**				

Classification of noise phases

- We take all URZ noise phases we have and which were not used for training — 273972 noise phases

Re-trained weights				
	N	P	S	T
N	**226601**	—	—	—
P	14624	—	—	—
S	25745	—	—	—
T	7002	—	—	—
accuracy **82.71%**				

Conclusion and future plans

- The re-training using more data looks promising so far, classification accuracy on test data set is higher than with the old weights
- If URZ re-training successful, our goals is to create a re-training software which would enable us to easily create new station-specific weight files for all 3C stations