Yerevan State University

FACULTY OF INFORMATICS AND APPLIED MATHEMATICS

SOFTWARE AND INFORMATION TECHNOLOGY
CHAIR

Final work

Subject: Word-depth knowledge of language teaching methods

Student, Mr. Harutyunyan

Head Armen Andreasyan

YEREVAN 2016

**Page 2**

Contents

1

Page 3

1 Introduction

Speech recognition language into the language of the speech recording decision. A person with this problem can be solved without the knowledge of the language. Computer image of the other. computer speech recording is presented as a sequence of numbers, which should be determined on the basis of the tongue: The set of data classification task (data classification), and its applicable to machine learning (machine learning) methods. These methods need to be a part of the intervention specialist. The specialist apart from recording such features (feature extraction), which he believes that it may be useful for solving the problem. These features classification of machine learning is used in some way.

Over the past decade, increasing capacity Computer Engineering contributed to deep learning (deep learning) the development of methods that are almost require specialist intervention and are able to get the data a presentation (data representation), which is based on machine learning standard methods to achieve sufficiently good results. For example, recent During deep convolution neural networks (deep convolutional neural networks) The record results are recorded in the pictures and objects recognition Many problems of speech recognition. Deep Neural Network by 2012 successfully won [5] pictures of objects recognition ImageNet Large Scale Visual Recognition Competition Competition (ILSVRC), which lead to the specialist intervention methods were employed.

1.1 Performance Data

Machine learning is a key problem in the presentation of new data Finding. For the purposes of finding a new presentation,

- Data compression
- Noise removal





- A "good" performance search

More "good" performance implies a presentation of data on which The application of machine learning methods more effective. commonly It is believed that the new presentation should contain data describing important features and contain noise.

Data presentation can be found on the new multi-level, hierarchical approach, the initial presentation of data, every step of the previous Based on the construction of a new presentation.

1.2 deep learning and deep neural networks

Deep learning machine learning branch, which examines in depth Presentation-finding methods. Deep learning of neural models for family networks. Neural networks are machine learning models that are similar to biological neural networks are used for different types of functions approximation for. Neural network is composed of many neurons, between which There are one-way links. This information is transmitted via connections of a neuron another neuron. Each neuron has an input, output and activation of function. Access to some figures, neuron is linear combination of the numbers (the combination of factors call neuron parameters), which applies its activation function. Received from the neuron called activation, which

a neuron discharge and can then be transferred to another neuron. network.
The people will call all the parameters of the neurons of the network settings.
Together with the connections between their neurons constitute a directional graph. network
the neurons, which do not receive any information call neural network
input neurons, and their multitude of network access. input neurons
Rather than use the phrase in neuron activation
call neuron term value. The network also separated
a subset of neurons in some, whom we will call the network gateway.





If there is no directed cycle graph neural network, then we will say,
the network of direct propagation neural network (feedforward neural network), otherwise
If we say that the neural network is a network of recurrence (recurrent neural network).
Recurrence neural network neurons are calculated aktivatsianere
each discrete point in time, based on the access point and network
the last time on aktivatsianeri neurons.

Teach a neural network function approximation means that some network
select settings so that when neurons values Access
arguments assigned function, the network exit approximate calculations
If the value of the function arguments. When the network is an approximation
. → function, where the crowd is finite, we will say that the network is trying to
solve classification problems.

Neural networks differ in their structure, how
contain neuron, activation of neurons which have functions as
connected with each other. It features neural networks, which do not change
training during called are network hiperparametrer. network
hiperparametrer the network size, the number of neurons contained in the network,
network teaching algorithm parameters, and so on.

1.2.1 activation function

Neural networks are selected as a function of differential activation and monotone functions. Frequently used functions are:

- () =
- () = max (0, )
- () = 1 / (1 + -)
- *h* ()
- () =

Σ= 1

4



The 4 functions of one variable depending on the functions and multivariate from application of the members and the masses are applied. function Vector and returns the vector. action from all vector of positive numbers, their sum is equal to 1. this Neural activation function is used to solve classification problems networks when the last layer needs to have a vector of probabilities, each component of which will show some probability of belonging to the class.

1.2.2 data, loss of function, loss of function of empirical

Target: → function that will estimate usually given

There Couples collection, = {(,) | = 1, .., = ()}, which

call data. Neural network is trying to find a function for good

approximation of functions of some class. Neuronal Network Function

approximation appoint whoever.

: 2 → loss of function of the type of the function call, if (,) does

describes the size of the error by approximation. For example may be

defining an arbitrary function of distance in space. loss

5

**Page 7**

Based on empirical loss of function is defined by a function () =

$^{1}\Sigma_{=1}$ (()) Where the vector of all parameters of the network.

Neural Networks goal is to find a vector which empiric

loss of function will be as small as possible. Usually - and just happen

differential, and () are applicable to minimize the mathematical

optimization methods.

1.3 multilayer neural network

Neural network multilayer neural network call when neurons

grouped in successive layers so that the neurons in the same layer.

There are no connections, and each neuron can be received only as an access layer to its preceding

aktivatsianere layer neurons. The first layer will be the access network, and the last the network access layer. Neural networks having four or more layers are called Deep Neural Networks. Defines the different types of layers,

differ from the previous layer neurons layer neyronnerin

To turn on the form. It should also be noted that usually one layer of neurons

aktivatsianeri functions are the same, therefore, often activation

The idea of the function will return the strip. Layer neurons aktivatsianerits

call layer consisting of a mass exit.

Multilayer neural networks learning is done Backpropagation algorithm [6] . It should be noted that the nonlinear neurons enough When using activation functions, neural network can be accurately approximate any arbitrary function [7 ].





### 1.3.1 Full coherent layer

If the full increase appoint coherent layer (fully connected layer) output vector, and compared to the previous output vector layer, the layer in the calculation of fully connected shall be as follows:

= (+) (∈, ∈, ∈ x, ∈)

and has a layer of settings are called the deviation (bias): Is a layer of activation function. The number of layers of neurons, has a hiperparametr coherent layer. The layer can be fully coherent picture follows:

Two slices of fully connected neural network. The picture is taken from [ 20] from.

1.3.2 Convolution layer

Convolution layer (convolutional layer) neurons are arranged into three-dimensional structure, like a three-dimensional array of cells, so it can be arbitrary neuron
3 in the index,. Convolution layer suggests that the earlier
layer solution should also have a similar structure, but may be different

7

**Page 9**

size. Convolution layer discharge means respectively, and the outcome of the previous layer. appoint cent. Layer exit convolution calculation takes place as follows:

= (Σ * +)

= 1

(∈ x x $h$, ∈ x (- + 1) x ( $h$ -$h$ + 1) , ∈ x $h$,

= 1 .., = 1 .., ∈)

A layer activation function. * Is a discrete convolution operation (discrete

convolution): ,, $h$ hiperparametrern the numbers scroll bars.

,,, = ,, + 1-, $h$ + 1- - and there will be a four-dimensional array from the vector

Convolution network settings. . matritsnerin called Magnet nuclei.

Discrete convolution matrices are operandnere action, action implemented as follows:

$$-1 \quad h\text{-}1$$
$$(*) = \Sigma\,\Sigma +, + -, h-$$
$$0 = \quad 0 =$$

$$(\in \text{x}, \in \text{x}\ h, (*) \in (- + 1)\ \text{x}\ (-h + 1)\ , \leq, h \leq)$$

and discrete convolution operation between the matrix and is equal to

discrete correlation between matrix operation, which is defined as follows: somehow,

$$-1 \quad h\text{-}1$$
$$(\star) = \Sigma\,\Sigma +, +,$$
$$0 = \quad 0 =$$

$$(\in \text{x}, \in \text{x}\ h, (\star) \in (- + 1)\ \text{x}\ (-h + 1)\ , \leq, h \leq)$$

Convolution layer turns out that neurons are divided into groups (group index from the state), and every group of neurons are arranged in a two-dimensional array. each





neuron receives information from all groups in the previous layer, but each group x $h$ size only to the local domain neyronnerin. In the same group neurons settings are the same (this phenomenon is called parameter sharing): Convolution layer can be depicted as follows:

These groups are large rectangles of black lines in the picture,
neurons are depicted as small squares. $_1$ is a matrix containing 4
a three-dimensional array. 4 matrices that are depicted in the picture on the left side. As shown in the picture settings are the same neurons in the same group. all blue neurons
settings are $_1$, and all red, $_2$. The picture is taken from [21] from.

1.3.3 Foundry layer

Casting layer (pooling layer) is usually put on the roll after layer.

If growth appoint convolution output layer, and the layer casting by the outcome, it's

The calculation takes place in the following way:

$$_{,,} = \quad {}_{0 \leq <, 0 \leq < h} \quad {}_{,(-1)+,(}-1)\,h + \text{(max pooling layer)}$$

or

$$_{,,} = \frac{1}{h} \quad {}_{0 \leq <, 0 \leq < h} \quad {}_{,(-1)+,(}-1)\,h + \text{(average pooling layer)}$$

$\in$ x, ,, $\in$ x, which has the largest natural number that

(- 1) + - 1 ≤ , and has the largest natural number that $_{(-1)\,h+h}$ - 1 ≤ .





$h$ casting call numbers and domain sizes (pooling region size), and $h$ figures

moving a step size of the first and second axes (stride size): Foundry

layer has no parameters. Layer 4 hiperparametrer,,, $h$, $h$:

1.4 Convolution neural network

Multilayer neural networks is called the convolution, the main

Convolution component layers. Magnet Networks is the first significant pattern-5 LeNet [8 ] The network, which was handwritten digits classification MNIST famous handwritten digits database.

The picture shows the 6 slices of LeNet convolution neural network, which is getting 2 56x46 size photos colorless entrance, exit 50 from the vector length.

1.5 recurrence neural network

Recurrence neural networks are working with sequential data. Access to such data, as opposed to direct distribution networks, do not necessarily have a fixed length. Recurrence neural networks input data sequences whose members dimensional are vectors. th-th member of the network at the time of entry of the sequence.





Recurrence neural network to determine the outcome determined by the first output neurons. Th time at the network exit point of the output neurons t-aktivatsianeri combination of the vector. Recurrence network exit sequence, which is th-th member of the network solution.

Recurrence neural networks are taught Backpropagation through time (BTT) through algorithm. Network call ordinary recurrence following calculations

Enforcement Network,

$$h = (h \text{ -1} + +)$$

$(h \in, \in, \in x , \in x, \in, = 1 .., h\ 0 = 0\ )$

$h$ vector network called the internal state of the vector (hidden state): and has a network

the settings are. This network recurrence neural network parzaguynn it

2 ways to depict,

The left side of the network depicted in unopened condition, and openly shown recurrence relation. In the right part of the same network is opened to the time axis condition. The picture is taken from [ 22] from.

1.5.1 GRU recurrence neural network

During the regular recurrence networks are learning algorithm BTT

the problems that are impossible for ordinary recurrence networks

The use of long sequences. It turns out that ordinary

recurrence neural network time *2nd* derivative according to *1* of *the* exit point

(1 "2) at the exit or tends to 0, or tends to infinity. in the first case

The network can not 1 the information received is kept up th 2 th





moment and therefore does not learn a long time, key dependencies, and

BTT second case, the algorithm works just frustrated, because all

aktivatsianere and derivatives very quickly recognize the great values. all this

detail can to meet [9] Art.

To overcome the above problem have been developed several new recurrence neural networks that have a "long memory". These are LSTM and GRU networks [10] [11]. This work used only GRU recurrence neuronal networks, as they are faster and the results obtained are not LSTM- who obtained results of yields. GRU major networks and LSTM the difference between the usual recurrence of neural networks is that the network internal state vector is blocked by locks that control information on entry and exit. GRU network shall be made through the following somehow,

$$= ( {}_{()} + {}_{()} h_{-1} + {}_{()} )$$

$$= ( {}_{()} + {}_{()} h_{-1} + {}_{()} )$$

$$\tilde{h} \quad h = ( + h_{-1} + {}_{(h)} )$$

$$h = \circ h_{-1} + (1 - ) \circ h$$

$$(h \in , \in ,, {}_{()} , {}_{()} \in {}_{x} ,, {}_{()} , {}_{()} \in {}_{x} ,$$

$$_{()} , {}_{()} , {}_{(h)} \in , = 1 .., h_{0} = 0)$$

« ∘ » action vectors members and multiplication operation.

$, {}_{()} , {}_{()} , {}_{()} , {}_{()}$ and Arrays $_{()} , {}_{()} , {}_{(h)}$ vectors network the settings are. Antsakhutsere both, z and r vectors.

one step GRU recurrence of the network can be depicted as follows:

12

**Page 14**

Information flows are shown by arrows, rectangles, activation
Functions of places, regions, application of arithmetic operations
respectively. The picture is taken from [ 22] from.

### 1.5.2 recurrence neural network is a multilayer neural network layer

Recurrence neural network can be viewed as a multilayer neural network layer. The entrance to the two-dimensional array and returns a two-dimensional mass. Access th row as a recurrence neural network time th at entry and exit th line network will be defined at the exit. this recurrence neural networks used in this work comment, which allows to build multilevel recurrence Neural networks and use the recurrence neuronal layers of convolution networks.

Objective 2 and Data

Word recognition of the language recommended by TopCoder organization from. 67176 PCs have been given 10-second speech patterns which for each of the language specified in the sounds, and another 12320 pcs for example, which is indicated for the language. it should be predicting. Almost all instances, noise There is no one person's voice and sounds. 176 recordings in different languages, which are relatively little-known languages (mainly in Africa, Asia and Oceania Nations languages).

All the examples given in the mp3 format, which we convert to wav format. Wav format stored in the discrete version of the sound wave, 44100 Hz frequency. Ten seconds of speech audio The sequence contains the number 441000. such a sequence for neural network is still hard to find you the language, you have data new presentation. That is why, for example, building a recording spektrograme. It turns out the pictures 256x858 in size,

Picture of horizontal axis is time, vertical axis, frequency, and The picture at the top of the lower hachakhakanutyunern. At any time corresponding column is written by the sound wave Fourier series coefficients (Black to white ratio increasing direction). The maximum frequency of 11 kHz to.

This form of data presentation is suitable for deep convolution neural networks. The network receives input x $h$ size [0,1] numbers composed matrix and final stratum returns 176 Dimension vector of probabilities. This applies only to the picture the upper half of which corresponds to frequencies [0 - 5.5 kHz] range. Human speech basically that the range of harmonics happen.

## 3 Approaches

In this paper we use 3 types of neural network,

- convolution neural network
- GRU recurrence neural network
- comparison of the previous two Network

Below we will present the most successful model descriptions of each type.

### 3.1 Convolution neural network

| type | The core group | Width | Length | size / step | activation function | |
|---|---|---|---|---|---|---|
| 0 Input | 1 | 128 | 858 | | | |
| 1 Conv | 16 | 122 | 852 | 7x7 / 1 | Relu | |
| 2 MaxPool 16 | | 62 | 427 | 3x3 / 2 | | Pad = 2 |
| 3 Batch Norm | 16 | 62 | 427 | | | |
| 4 Conv | 32 | 58 | 423 | 5x5 / 1 | Relu | |
| 5 MaxPool 32 | | 30 | 213 | 3x3 / 2 | | Pad = 2 |
| 6 Batch Norm | 32 | 30 | 213 | | | |
| 7 Conv | 32 | 28 | 211 | 3x3 / 1 | Relu | |
| 8 MaxPool 32 | | 15 | 107 | 3x3 / 2 | | Pad = 2 |
| 9 Batch Norm | 32 | 15 | 107 | | | |
| 10 Conv | 64 | 13 | 105 | 3x3 / 1 | Relu | |
| 11 MaxPool 64 | | 8 | 54 | 3x3 / 2 | | Pad = 2 |
| 12 Batch Norm | 64 | 8 | 54 | | | |
| 13 Conv | 64 | 6 | 52 | 3x3 / 1 | Relu | |
| 14 MaxPool 64 | | 4 | 27 | 3x3 / 2 | | Pad = 2 |
| 15 Batch Norm | 64 | 4 | 27 | | | |
| 16 Fully connected | 256 | | | | Relu | |
| 17 Batch Norm | 256 | | | | | |
| 18 Dropout 256 | | | | | | Prob = 0.5 |

| | | | |
|---|---|---|---|
| 19 Fully connected | 176 | | Softmax |

«Batch Norm» and «Dropout» layers are interpreted in the 4th and 5th sections.

3.2 GRU recurrence neural network

| type | access length | internal capable neutrons | activation function | |
|---|---|---|---|---|
| 0 Input | 858 | 128 | | |
| 1 GRU | 858 | 500 | Tanh | |
| 2 Batch Norm | 858 | 500 | | |
| 3 GRU | | 500 | Tanh | Only the last exit |
| 4 Batch Norm | | 500 | | |
| 5 Fully connected | | 176 | Softmax | |

«Batch Norm» and «Dropout» layers are interpreted in the 4th and 5th sections.

3.3 Convolution + recurrence Network

| type | The core group | Width | Length | size / step | activation function | |
|---|---|---|---|---|---|---|
| 0 Input | 1 | 128 | 858 | | | |
| 1 Conv | 16 | 122 | 852 | 7x7 / 1 | Relu | |
| 2 MaxPool | 16 | 62 | 427 | 3x3 / 2 | | Pad = 2 |
| 3 Batch Norm | 16 | 62 | 427 | | | |
| 4 Conv | 32 | 58 | 423 | 5x5 / 1 | Relu | |
| 5 MaxPool | 32 | 30 | 213 | 3x3 / 2 | | Pad = 2 |
| 6 Batch Norm | 32 | 30 | 213 | | | |
| 7 Conv | 32 | 28 | 211 | 3x3 / 1 | Relu | |
| 8 MaxPool | 32 | 15 | 107 | 3x3 / 2 | | Pad = 2 |
| 9 Batch Norm | 32 | 15 | 107 | | | |
| 10 Conv | 32 | 13 | 105 | 3x3 / 1 | Relu | |
| 11 MaxPool | 32 | 8 | 54 | 3x3 / 2 | | Pad = 2 |
| 12 Batch Norm | 32 | 8 | 54 | | | |
| 13 GRU | 32 | 15 | 1 | 3x3 / 1 | Tanh | only final exit. |
| 14 Concatenation | 480 | | | | | |
| 15 Batch Norm | 480 | | | | | |
| 16 Fully connected | 176 | | | | Softmax | |

«Batch Norm» layer is explained in Section 5.

This network is important in the 13th layer, in which there are 32 PCs with the same settings GRU recurrence neural networks. Each GRU operates 54x8 size the serial data and returns the 15-dimensional vector. 14th layer 32 pieces of such vectors are attached to each other, and it is 480-dimensional vector. The network the picture on the next page.

18

**Page 20**

176

fully connected

32x15

54

...

1x15

GRU

1x8

32x8x54

Casting 3x3 / 2

32x13x105

3x3 convolution kernel

32x15x107

Casting 3x3 / 2

32x28x211

3x3 convolution kernel

32x30x213

Casting 3x3 / 2

32x58x423

5x5 convolution kernel

16x62x427

Casting 3x3 / 2

16x122x852

7x7 convolution kernel

access
1x128x858

19

**Page 21**

4 Optimization

The last layer of all the models listed in the previous section, it turns out

( 1 , 2 , ..., 176 ) Probability Vector. has demonstratiton give

the probability that a particular pattern of sounds th language. Spoken language

The final assessment of the network will be, = argmax
= 1..176

Data is given for the number of the language pattern . this fact

We can also give probabilities vector; = 1 [=] . choose loss

function so that it evaluate vector cent approximation error. this

loss of function, we choose to work anywhere as discrete mutual

entropy loss function (discrete cross entropy loss function).

$$(,) = -\sum_{=1}^{176} \text{Log}\ (\ )$$

Optimiziatsiayi task will be the following:

$$() = \ \ 1 \ \sum_{=1} (,\ (\ )) \quad \rightarrow$$

(- A vector, composed of all the parameters of the network ∈ )

Mainly for in-depth study into the optimization problems

use gradient decline in species. normal gradient

fall time of each iteration is done by the following:

= -1 - A ∇ (()) (*) , where α- is the only algorithm hiperparametrn and is called a learning step (learning rate):

This algorithm is not applicable in this case because ∇ (()) to calculate the need to all samples run on the network and make new parameters

The change in the net of all data on the operation takes 10 minutes

up to one hour. To solve this problem is to make the (*) has all th





Following his example, calculate the average number of copies of loss of function. Gradients. All models in this work we have chosen = 32.

To improve the quality and speed of optimization exist A number of species falling gradient. Were used in this Two of them, momentum [13 ] and adadelta [12].

Momentum- iteratsiayum done in the case of each of the following actions:

$= _{-1} + A\, \nabla ()$

$= _{-1} -$

This work = 0.9.

Adadelta- iteratsiayum done in the case of each of the following actions:

$\nabla = ()$

$[\ ^2\ ] = [\ ^2\ ]_{-1} + (1)\qquad 2$

$\Delta = - \dfrac{\sqrt{[\Delta\ ^2\ ]_{-1} +}}{\sqrt{[\ ^2\ ] +}}$

$[\Delta\ ^2\ ] = [\Delta\ ^2\ ]_{-1} + (1 -)\ \Delta\qquad 2$

$() = (-1) + \Delta$

This work, = 0.95 and = $10^{-6}$.

We used to enhance the speed of optimization Batch normalization [ 14]-type layer, which aims to multilayer neural normalizatsnel network layers exits, exits to always be on the same distribution.





5 Germotarkum

Machine learning is a common problem germotarkman (ovefitting) problem. Germotarkman good approximation to the network function of the spots, network used in teaching and comparatively worse approximation for new copies. Almost all used in this work networks to reach 100% accuracy on samples that were used network teaching time. Germotarkume very well seen when we draw iteratsianeri loss of function of the dependence graph, Loss of function values of the diagrams, network training course. Shows in blue the value of the loss of function of the illustrations, used to teach the network, and red, the loss of function of new on copies. The first network germotarkume quite high, blue and red graphs difference is about 0.3. Germotarkum second network exists, but it relatively less, because the difference between blue and red graphs about 0.06

a. Note that the accuracy of the network used and new copies respectively 100% and 92.4%, while the figures for the second network respectively 100% and 98.45% are.

We use 3 ways to reduce Germotarkume,

1. Data artificial enrichment
2. Dropout
3. L2 Magnetoelasticity





5.1 Data artificial enrichment

The artificial enrichment of the database

Examples of specific actions are generated as a result of new examples,

Examples are different from the original but do not change their class (our

If the language is not changed). This work has been applied to data processing 2

weather

1. 10-second speech randomly choose 9 seconds

   the duration of the piece. This operation performs a learning process,

   being absent for extra memory. The audio for the ultimate

   You can get different answers to that record 9 seconds

   duration pieces onto the network probabilities

   calculate the average. In this paper we choose = 20.

2. The initial period of the previous convolution neural network method

   used together [ 15] The method described in the data

   artificially enriched. It is a small amount of deformation

   Police spektrogrami frequency axis, in some places

   leaving it in some places, clicking. Although this method helps to improve

The results, however, it is not for use during training. this
The method requires the use of large amounts of memory,
New pictures of each picture to be kept.

5.2 Dropout layer

Dropout layer [16 ] is a special type of layer, which is itself random neurons in the preceding layer aktivatsianere pays 0, or leave the same. Neural activation probability of a call dropout Layer 0 possibility. the likelihood of another layer Dropout hiperparametr which chosen experimentally. Dropout layer is "forced" to be neyronnerin independently of one another.





5.3 L2 Magnetoelasticity

L2 regularizatsiayi time is added to the empirical loss of function 2 parameter norm coefficient multiplied. coefficient hiperparametr which should be selected experimentally. L2 Magnetoelasticity "forcing" the Network settings may not be very large. Thus the net result is a more Polished functions that are less germotarkum.

## 6 Performance

Almost all the programs needed for this work, written in python programming language. Used in symbolic calculations and automatic Theano for differentiation [ 17] library and neural networks intended for Lasagne [18] library. Initially used Also designed for neural networks Caffe [17 ] library, which is suitable for for rapid design and implementation of standard models. Later in Caffe-Because of the failure to achieve sufficient flexibility in the transition to the Theano Library.

Theano library network learning algorithm translates Nvidia CUDA language designed for graphics cards. All calculations are carried out on the Nvidia GTX 980 vidokarti. Training lasts on average 5 hours per model up to 1 day. networks on a recording of works by an average of 0.02 seconds.

# 7 Results

In this section, we will present only on the unused copies networks
The results showed, the amount of error.

## 7.1 Convolution Network

Convolution of the best network model is presented in section 3.1. these
The model gives a 3.5% uncertainty. The network 2 times larger in size and of the same type
The network provides a 3.8% uncertainty.

## 7.2 recurrence Network

Recurrence of the best network model is presented in Section 3.2. these The model gives a 1.58% band. Germotarkume minimize training Magnetoelasticity process than half of imported L2, = 0.001 coefficient. one layer containing recurrence net the best result that could have been 8.12%.

7.3 Convolution + recurrence Network

Recurrence and roll together the best from the utilization of networks The model is presented in section 3.3. This model gives a 2% band. In this model, used 4 convolution layer. When used in convolution layer 3, The results are worse, uncertainty is 3.04%. When the model is removed the same condition, uncertainty becomes a 7.6% recurrence network parameters. In all cases, the reason for the increase is not a model error of "capacity building" decreasing, but increasing germotarkman. These types of L2 networks Magnetoelasticity and dropout layer almost does not help in terms of reducing germotarkume. It helps The first version of the artificial enrichment (5.1.1). That means 3.3 the uncertainty of the model described in section managed to reach 1.32 percent.





7.4 Voting

Forecasts for all models can be represented at each other to combine and get better results. To this end, all models predicted probability vector derived vectors ktsagrumits the use of a fully connected neural network layer, which gives Final answer? In this way managed to get a 0.64% error that about two times less than the best individual model uncertainty.

## 8 Conclusion

Neural networks, as expected, high accuracy consider recording of the numbers you language. Magnet Networks successfully develop data, step by step to get data for new performances, but not good use the time factor. Recurrence neural networks openly based on the time factor and the results are better than the Convolution networks. task recurrence of the network is that the network begins Ranking almost immediately on spektrogrami that although

Performance recording is quite good, but not yet sufficiently developed
and does not contain unnecessary information for problem solving. that
Therefore, we consider 3 model, which is developing a network of convolution
spektrograme and transfer recurrence of the Web. This model is still very
has been subjected to the tests, but the results are rather good.
In the future we are going to continue developing this model.





9 References and Links

[1] Christopher Bishop. Pattern Recognition and Machine Learning. Springer , 2006.

[2] Ian Goodfellow, Yoshua Bengio and Aaron Courville. Deep Learning. Press MIT, 2016 .

[3] Gregoire Montavon. Deep learning for spoken language identification. Microsoft Research, NIPS 2009.

[4] Wouter Gevaert, Georgi Tsenov, Valeri Mladenov. Neural Networks used for Speech

Recognition, Journal of automatic control, University of Belgrade, Vol. 20: 1-7, 2010 .

[5] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. NIPS 2012 .

[6] David E. Rumelhart, Geoffrey E. Hinton, Ronald J. Williams. Learning representations by back-propagating errors. Nature Vol. 323, 1986 .

[7] K. Hornik, M. Stinchcombe, H. White. Dermition. Multilayer feedforward networks are universal approximators. Neural Networks Vol. 2, pp. 359-366, 1989 .

[8] LeNet-5 convolutional neural networks. URL http://yann.lecun.com/exdb/lenet.

[9] Razvan Pascanu, Tomas Mikolov, Yoshua Bengio. On the difficulty of training recurrent. neural networks. JMLR Workshop and Conference Proceedings, Vol. 28 , 2013.

[10] Sepp Hochreiter, Jurgen Schmidhuber . Long Short-Term Memory. Neural Computation 9 (8): 1735 {1780, 1997 .

[11] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, Yoshua Bengio. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. arXiv: 1412.3555 . 2014 .

[12] Matthew Zeiler. ADADELTA: An Adaptive Learning Rate Method. arXiv: 1212.5701, 2012 .

[13] Ilya Sutskever, James Martens, George Dahl, Geofrey Hinton. On the importance of initialization and momentum in deep learning.

[14] Sergey Ioffe, Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. arXiv: 1502.03167, 2015.

[15] Navdeep Jaitly, Geoffrey E. Hinton. Vocal Tract Length Perturbation (VTLP) improves speech recognition. ICML 2013 .





[16] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky Ilya Sutskever, Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research, Vol. 15 , 2014 .

[17] Theano. http://deeplearning.net/software/theano

[18] Lasagne. https://github.com/Lasagne/Lasagne

[19] Caffe. http://caffe.berkeleyvision.org

[20] http://cs231n.github.io/neural-networks-1

[21] http://deeplearning.net/tutorial/lenet.html

[22] http://colah.github.io/posts/2015-08-Understanding-LSTMs