OPTIMIZATION OF NEURAL NETWORK INPUTS BY FEATURE SELECTION METHODS

Michal Prochazka, Zuzana Oplatkova, Jiri Holoska, Vladimir Gerlich
Tomas Bata University in Zlin
Faculty of Applied Informatics
Department of Informatics and Artificial Intelligence
Nam. T.G. Masaryka 5555, Zlin 760 05, Czech Republic
E-mail: {mprochazka, oplatkova, holoska, gerlich}@fai.utb.cz

KEYWORDS
Feature selection, dimension reduction, artificial neural networks, steganalysis.

ABSTRACT

The main idea of this paper is to compare feature selection methods for dimension reduction of the original dataset to reach optimization of steganalysis process by artificial neural networks (ANN). Feature selection methods are tools based on statistic exploited in pre-processing step of data mining workflow. These methods are very useful in a dimension reduction, removing of insignificant data, increasing comprehensibility and learning accuracy. Dimension reduction leads to reduced computational resource consumptions, which is validated by ANN simulations. Steganalysis is a field of the computer security, which deals with a discovering of hidden information in images which is normally unrecognizable. All dataming processes, which reduce the dimension of ANN input layer, should keep accuracy of steganalysis on the original level.

INTRODUCTION

This paper deals with a comparison with feature selection methods and their influence on steganalysis by means of artificial neural networks (ANN).
Steganalysis is connected with information security. Mainly in companies, information security is a very seriously solved problem nowadays. All employers have to pay attention to their employees if company secrets and know-how are not spread out of the company. One of the possibilities how to leak the information is to use a steganography (Cole, 2003). Steganography (Cole, 2003) and cryptography (Goldwasser, 2001) are connected together. Cryptography is strong in key usage and codes messages with a high security. But if the message is sent unsecure, attacker will notice it very soon and will try to break it. Therefore steganography helps with secure transfer of secret messages. It codes a message inside the picture or other multimedia files which can be sent e.g. via emails. If you see such a picture, normally you do not recognize that there is a secret message. And this is the point. Crackers will go through and will not give the attention to the message.
Therefore to have a detector of steganography content in the multimedia files is very important. To reveal a steganography content is called steganalysis, i.e. a detection of files with hidden information of without hidden information which was inserted by means of steganography.
Firstly, classification of stego and cover images by means of ANN was introduced in (Oplatkova, 2008a, Oplatkova 2008b, Oplatkova 2009). The results have shown possible area of the ANN structure optimization, mainly the number of input parameters. In (Prochazka, 2010) a J48 tree like selector for attributes was used as a preliminary research paper in this field. A different dataset have been used in current paper then in (Prochazka, 2010). The dataset has been created with width diversity in image resolutions, sizes (amount) of hidden information and used software for hiding information into image. This paper extends the research in optimization of number of neurons in ANN input layer for this different dataset and different feature selection tools.
The aim of this paper is to verify if the chosen approach – joining of modern tack into data analysis and artificial intelligence potential - can contribute to establishment of robust solution which could be applicable onto real problems of internet security.
The paper is divided into part with description of data mining methods and Huffman coding in following paragraph. ANN description and results continue in next parts.

DATASETS

Compared to a paper (Prochazka, 2010), both datasets were prepared again, with cover (without hidden information) and stego (with hidden information) images. Training set contains 20 000 instances (10 000 cover, 10 000 stego) and testing set works with 19 000 (9 500 cover and 9 500 stego), i.e. both sets are balanced.
In stego group images with different length of the hiding message and different resolution are balanced too. Length of messages are: 5, 10, 15, 30, 75, 150, 300, 600 Bytes and used resolutions: original and changed to 800x600, 1024x768, 1280x1024, 1440x900, 1680x1050, 1920x1200, 1920x1440, 2560x1600. Used stego algorithms are OutGuess (www.outguess.org), Steghide (Hetzl, 2008), CipherAWT (F5 algorithm) (Fridrich, 2002).
The datasets are randomly shuffled before the usage of dataming techniques.

Proceedings 25th European Conference on Modelling and Simulation ©ECMS Tadeusz Burczynski, Joanna Kolodziej Aleksander Byrski, Marco Carvalho (Editors)
ISBN: 978-0-9564944-2-9 / ISBN: 978-0-9564944-3-6 (CD)

Huffman coding

Huffman coding was used to extract information from images as ANN needs numerical values to run. Huffman coding was designed by David Huffman in 1952 (Cormen, 2001). This method takes symbols represented e.g. by values of discrete cosine transformation (which is one of methods how to present information in pictures like colour, brightness etc.) and coded it into changeable length code so that according statistics the shortest bit representation to symbols with the most often appearance. It has two very important properties – it is a code with minimal length and prefix code that means that it can be decoded uniquely. On the other hand, the disadvantage is that we must know appearance of each symbol a priori.
Following table (
Table 1) shows an example of dataset used within datamining techniques.

Table 1: Dataset sample

P1	P2	...	P64	CLASS
0	2915	...	1	1
0	2024	...	3	0
0	7997	...	27	0
0	1566	...	0	1
0	25882	...	157	0
0	16265	...	99	0
0	8995	...	90	0
0	10621	...	221	0

DIMENSION REDUCTION

Reduction of dataset dimension was done into two steps: firstly columns 11-16 and 27-32 were deleted because the values there were equal to zero.
The other dimension reductions were done by means of algorithm J48, Principal Component Analysis (PCA) and a supervised attribute selection filter. Both methods are available through freely distributed software WEKA (Witten 2011, http://weka.sourceforge.net/wekawiki/).

Algorithm J48

Algorithm J48 (Witten, 2011) is a filter of a tree decision algorithm. Tree decision methods are used for their good readability and understandability of a found solution because of graphical representation. Such approach is comprehensible also to users who are not familiar with these issues. This method was used for a comparison with (Prochazka, 2010) if the changes in the composition of the datasets will cause changes in results.

Principal Component Analysis

Principal Component Analysis (PCA) (Witten, 2011) creates new attributes from the original so that new attributes represents an equation (eg. (1)) with parts which contain coefficients and values of the former vectors.

$$V1=0.189*A40 + 0.188*A37 + ... \quad (1)$$

Supervised attribute selection filter

Supervised attribute selection filter is one of the basic implemented filters in software Weka (Witten, 2011). This tool uses the information about output for the purpose of the input dimension reduction.

ARTIFICIAL NEURAL NETWORKS

Artificial neural networks are inspired in the biological neural nets and are used for complex and difficult tasks. The most often usage is classification of objects as also in this case. ANN are capable of generalization and hence the classification is natural for them. Some other possibilities are in pattern recognition, control, filtering of signals and also data approximation and others.
Simulations were performed with feedforward net with supervision. ANN needs a training set of known solutions to be learned on them. Supervised ANN has to have input and also required output. ANN with unsupervised learning exist and there a capability of selforganization is applied.
The neural network works so that suitable inputs in numbers have to be given on the input vector. These inputs are multiplied by weights which are adjusted during the training. In the neuron the sum of inputs multiplied by weights are transferred through mathematical function like sigmoid, linear, hyperbolic tangent etc. Therefore ANN can be used for data approximation (Hertz, 2001).
These single neuron units (Figure 1) are connected to different structures to obtain ANN (e.g. Figure 2). These networks were design for different tasks.

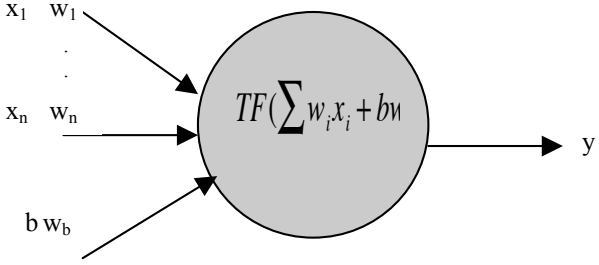
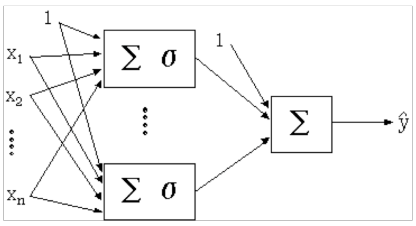
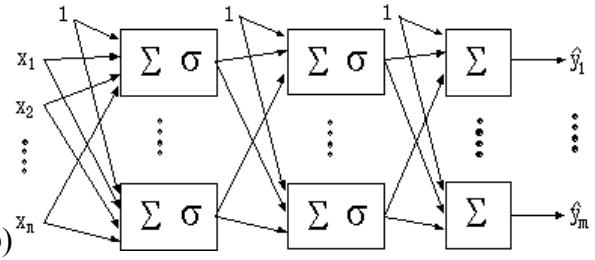


Figure 1: Neuron model, where TF (transfer function like sigmoid), x_1 - x_n (inputs to neural network), b – bias (usually equal to 1), w_1 – w_n, w_b – weights, y – output



a)



b)

Figure 2: ANN models with a) one hidden layer and b) with two hidden layer net and more outputs where $\sum \delta = TF[\sum (w_i x_i + b w_b)]$ and in this case $\sum = TF[\sum (w_i x_i + b w_b)]$, where TF is sigmoid. These pictures are taken from Neural Networks Toolbox for Mathematica environment (www.wolfram.com) since this tool was used during the simulations. Also names are taken from this tool to avoid other speculations what it means.

Future simulations expect a usage of soft computing algorithms for optimization of suitable parameters or structure called evolutionary, e.g. Self-Organizing Migrating Algorithm (Zelinka, 2004), Differential Evolution (Price, 1999) or HC12 (Matousek, 2010). Also a different kinds of neural networks will be used, e.g. RBF nets with implementation of GAHC algorithm (modification of HC12) (Matousek, 2011) for design of structure and estimation of parameters.

RESULTS

For comparison 6 training sets were prepared (2-6 in software WEKA):

1. original dataset with 64 attributes

2. reduction of original dataset about columns with zero values (columns 11-16 and 27-32)

3. attributes (45, 3, 61, 43, 22, 19, 25, 60, 5, 8, 6, 64, 38, 50, 9, 26, 34, 24, 63, 2, 35, 44, 47, 51, 40, 42, 37, 4, 36) selected by means of J48 algorithm. To confront with [Prochazka 2010], this time a quite complicated tree was built up with number of leaves 34 and size of the tree 67. To build the tree 30 attributes from the original set was needed which means the significant increase then in previous research (4 components were required).

4. and **5.** attributes chosen by means of PCA. In this case PCA creates 25 new vectors and for tests it was opt for 15 and 10 attributes (in 5 from each vector). Selection of parameters was done in this way because the requirement was to decrease the number of inputs as much as possible. Since algorithm J48 cut parameters down only on 30, PCA extracts attributes from 3 most important vectors for test 4 and from 2 for test 5. Test 4 then uses parameters 40,37,39,34,41,45,61,8,63,62 and test 5 parameters 40,37,39,34,41,45,61,8,63,62,55,57,58,61,7.

6. parameters 2, 45, 46, 47, 55, 61. Components were selected by means of supervised attribute selection filter.

Each set except the original one was prepared in WEKA software with similar results as follow (Table 2):

Table 2: Example of results from Weka software for J48

Correctly Classified Instances	19861	99.305 %
Incorrectly Classified Instances	139	0.695 %
Relative absolute error	1.9563 %	
Root relative squared error	15.5774 %	

After results and proposal of reductions were then tested in artificial neural networks. These results show that the time required for the training is less as the dimension of the neural networks was decreased. The training of ANN means to change weights to find an optimal solution and reduce the training global error to zero. It means that to reduce the number of weights means less parameters for optimization, i.e. faster optimization.

Settings of ANN

Feedforward ANNs with one hidden layer and different numbers of neurons (Table 3) in a hidden layer were used for the testing of found models by means of datamining techniques. As transfer function for output neuron a sigmoid was chosen, for hidden neurons hyperbolic tangent function was set up. Number of iterations was set up in all cases to 30. The differences between results in this paper and (Prochazka, 2010) appeared because of different dataset and different settings of ANN. In the test 5 the number of iterations was increased up to show that the convergence is slower than in other cases (eg.
Figure 3) and still the root mean square error (RMSE) did not decrease even after longer time (Figure 4).

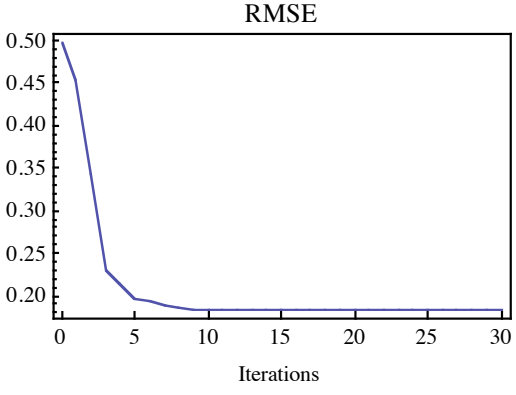


Figure 3: Decrease of RMSE in test 6

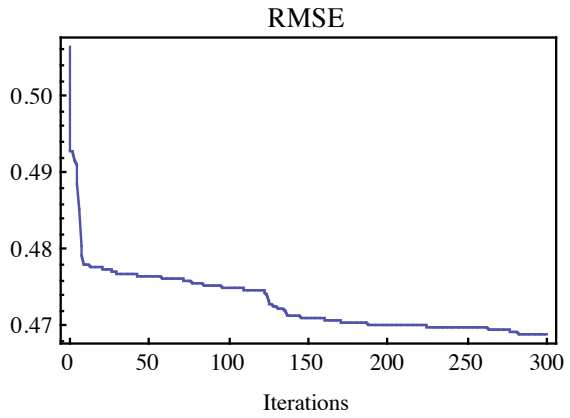


Figure 4: Decrease of RMSE in test 5

DISCUSSION

Table 3 shows comparison on training. Number of test is according above explanations. Number of parameters means the number of remained parameters used for design of a datamining model and used in neural network training and testing. As can be seen the number of hidden neurons were decreased according reducing the number of parameters. Root mean square error (RMSE) was low except test 5 where training took the longest time and the RMSE has not still decreased. The time consumed during training was lower several times in the test 6 compared to test 1. The time used for training was so small because of the reduction weights (26x in test 6 then in test 1) which should be optimized the net to be trained well.

Table 4 shows the testing results of ANN. It is interesting that test 2 where only parameters 11-16 and 27-32 were deleted because the real values were zero has a worse testing error than the test 1 with original data. We have no explanation for that result. Maybe more training would finish with a different score. Because of better results in other tests, the number of iterations in this case has not been increased.

Test 5 in Table 4 shows an unacceptable solution, as the classification error is too high. The other tests have the testing error only about 0.65 % to 1.4 % worse compared to test 1. This decline is acceptable, as time consumption much more lower during the training.

CONCLUSION

The paper deals with datamining technique for a reduction of input neurons in ANN and therefore also the number of weights which should be optimized to obtain a best settings of ANN during training phase. In this paper 6 test cases were prepared – for the original dataset and 5 datasets changed by means of algorithm J48, PCA and supervised attribute selection filter. All models decreased the number of input neurons and the time for training of the ANN. As the best model the test 6 with 6 parameters was selected. In this case the total error was higher than in cases 2 and 4 but the error for stego detection was lower. The false positive classification in cover images is a problem but to find image with stego information is much more important from the view of security. Results show that the preprocessing is very important. To employ datamining techniques here is a useful tool. The optimization of consumed time in all computations is very important and required in all fields.

FURTHER RESEARCH

Future research will be focused on better settings of neural networks as the percentage for detection of stego and cover images would be better as close to zero as possible. Here the simulations have different length of inserted messages which might cause problems with 100% classification as was reached in previous research. Also an interesting direction is to visualize separability of both classes by means of PCA.

Table 3: Comparison of training

Test	1	2	3	4	5	6
Number of parameters	64	53	30	15	10	6
Hidden neurons	10	8	6	4	4	3
RMSE	0,08	0,08	0,08	0,11	0,46	0,18
Iterations	30	30	30	30	300	30
Learning time [min:sec]	17:45	10:20	3:50	1:17	5:39	0:28
Number of weights	661	441	187	69	49	25

Table 4: Comparsion of testing

Test	1		2		3		4		5		6	
Image type	cover	stego	cover	stego	cover	stego	cover	stego	cover	stego	cover	stego
Number of Instances	9500	9500	9500	9500	9500	9500	9500	9500	9500	9500	9500	9500
False Classified	41	293	61	397	126	370	64	396	4432	100	463	137
Error	0,43%	3,08%	0,64%	4,18%	1,33%	3,89%	0,67%	4,17%	46,65%	1,05%	4,87%	1,44%
Test err.	1,76%		2,41%		2,61%		2,42%		23,85%		3,16%	

AKNOWLEGMENT

This work was supported by the grants: internal grant agency of Tomas Bata University in Zlin IGA/44/FAI/10/D, grant NO. MSM 7088352101 of the Ministry of Education of the Czech Republic, grant of Grant Agency of Czech Republic GACR 102/09/1680 and by the European Regional Development Fund under the Project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089.

.

REFERENCES

Cormen, T.H., Leiserson, Ch. E., Rivest, R. L., Stein, C.: Introduction to Algorithms, Second Edition. MIT Press and McGraw-Hill, Section 16.3, pp. 385–392, 2001. ISBN 0-262-03293-7.

Cole E., Krutz D. R.: Hiding Sight, Wiley Publishing, Inc., USA, 321 s., 2003 ISBN 0-471-44449-9

Fridrich, J., Goljan, M., and Hogea, D. "Steganalysis of JPEG Images: Breaking the F5 Algorithm." 5th Information Hiding Workshop, Noordwijkerhout, The Netherlands, Oct. 2002. URL: http://www.ws.binghamton.edu/fridrich/Research/f5.pdf. Last accessed: 2003-12-24.

Goldwasser S., Bellare M.: Lecture Notes on Cryptography, Cambridge, 283 s., 2001

Hertz J., Kogh A. and Palmer R. G.: Introduction to the Theory of Neural Computation, Addison – Wesley 1991

Hetzl S.: Steghide (1) - Linux man page [online]. [cit. 2008-05-21]. available from WWW: <http://steghide.sourceforge.net/documentation/manpage.php>.

Matousek, R.: Using AI Methods to Find a Non– Linear Regression Model with a Coupling Condition. Engineering Mechanics, 2011, vol. 17, No. 5/ 6, p. 419–431. ISSN: 1802– 1484

Matousek, R. HC12: The Principle of CUDA Implementation. In MENDEL 2010. Mendel Journal series. 2010. Brno: VUT, 2010. p. 303–308. ISBN: 978–80–214–4120– 0. ISSN: 1803– 3814.

Oplatkova 2008a: Oplatkova, Z., Holoska, J., Zelinka, I., Senkerik, R.: Detection of Steganography Content Inserted by Steghide by means of Neural Networks, VUT v Brne, FME, MENDEL 2008 14th International Coference on Soft Computing, Brno, 2008, 166-171, ISBN 978-80-214-3675-6

Oplatkova 2008b: Oplatkova, Z., Holoska, J., Zelinka, I., Senkerik, R.: Steganography Detection by means of Neural Networks, IEEE Operations Center, Nineteenth International Workshop on Database and Expert Systems Applications, Piscataway, 2008, 571-576, ISBN 978-0-7695-3299-8

Oplatkova 2009: Oplatkova, Z., Holoska, J., Zelinka, I., Senkerik, R.: Detection of Steganography Inserted by OutGuess and Steghide by means of Neural Networks, AMS2009 Asia Modelling Symposium 2009, IEEE Computer Society, Piscataway, 2009, ISBN 978-0-7695-3648-4

Price, K. (1999), 'An Introduction to Differential Evolution', In: (D. Corne, M. Dorigo and F. Glover, eds.) New Ideas in Optimization, (pp. 79–108), London: McGraw-Hill

Prochazka M., Oplatkova Z., Holoska J.: Datamining Optimization of Steganalysis by means of Neural Network, in Internet, Competitiveness and Organizational Security 2010, UTB Zlín, Czech Republic, ISBN 978 - 83 - 61645 - 16 – 0

Software OutGuess, www.outguess.org

WEKA software http://weka.sourceforge.net/wekawiki/

Witten I.H., Frank E., Hall M. A.: Datamining – Practical Machine Learning Tools and Techniques, Kaufmann Morgan, 2011, ISBN: 978-0-12-374856-0

Zelinka I., "SOMA – Self Organizing Migrating Algorithm", In: New Optimization Techniques in Engineering, (B.V. Babu, G. Onwubolu (eds)), chapter 7, 33, Springer-Verlag, 2004

AUTHOR BIOGRAPHIES

MICHAL PROCHAZKA was born in Kyjov, Czech Republic and went to the Tomas Bata University in Zlin, where he studied information technologies and obtained his degree in 2008.He is now studying doctoral program on TBU in Zlin in the field of data mining. His e-mail address is: mprochazka@fai.utb.cz

ZUZANA OPLATKOVA was born in Czech Republic, and went to the Tomas Bata University in Zlin, where she studied technical cybernetics and obtained her MSc. degree in 2003 and Ph.D. degree in 2008. She is a lecturer (Artificial Intelligence) at

the same university. Her e-mail address is: oplatkova@fai.utb.cz

JIRI HOLOSKA was born in Horovice near to Prague, Czech Republic and went to the Tomas Bata University in Zlin, where he studied security technologies, systems and management, he obtained his degree in 2008. He is now studying doctoral program on TBU in Zlin in the field of steganalysis and artificial intelligence. His e-mail address is: holoska@fai.utb.cz

VLADIMIR GERLICH was born in Uherske Hradiste, Czech Republic and went to the Tomas Bata University in Zlin, where he studied automatic control and obtained his degree in 2008. He is now studying doctoral program on TBU in Zlin in the field of heat transfer in buildings. His e-mail address is: gerlich@fai.utb.cz