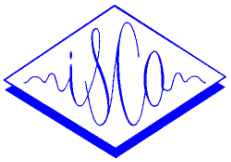
6th International Conference on Spoken Language Processing (ICSLP 2000)
Beijing, China
October 16-20, 2000

DATA-DRIVEN IMPORTANCE ANALYSIS OF LINGUISTIC AND PHONETIC INFORMATION

Achim F. Müller[1,3], *Jianhua Tao*[1,2], *and Rüdiger Hoffmann*[3]

(1) Siemens Corperate Technology, Otto-Hahn-Ring 6, D-81739 Munich, Germany,
email: achim.mueller@mchp.siemens.de
(2) Tsinghua University, Beijing, China, tjh@tts.cs.tsinghua.edu.cn
(3) Dresden University of Technology, D-01062 Dresden, Germany



ABSTRACT

In this paper the weight decay concept known from neural network theory is applied to the two modules involved in prosody generation within our text-to-speech system Papageno. Both modules are based on neural networks (NN). Preprocessing layers are inserted connected to the inputs of specialized NN architectures via diagonal weight matrices. The weight decay concept is applied to the weights of these diagonal weight matrices. This allows an importance analysis of the used input parameters in the context of the used NN architectures.
In the symbolic prosody module the importance for phrase break prediction of part-of-speech (POS) tags could be evaluated. Further, the necessary length of a POS context window could be analyzed and optimized.
For f0-generation for Mandarin language an importance analysis of the phonological information could be performed. The importance analysis led to an optimized input feature set, reducing the squared error of the used NN architecture.

1. INTRODUCTION

The realization of a multilingual and domain independent text-to-speech (TTS) system is typically based on data-driven modules. These modules, e.g. prosody generation module, apply some kind of learning technique. The learning techniques usually can be applied to new domains and languages. It is, however, often difficult to choose and optimize input parameters for new domains and languages.

For the case of symbolic prosody (phrase break prediction and accent prediction), the following learning techniques have been applied: For phrase break prediction, approaches are based on classification and regression trees (CARTs) [5][10][13], hidden markov models [2], or NNs [8]. For accent assignment, CARTs [4][7], and NNs [14] have been used. While the CART-based approaches allow an interpretation of the importance of the used input information, an interpretation cannot be given easily for the other approaches. This is especially true for NNs. For the case of f0-generation, this problem is also known. In [11] the input parameters are heuristically optimized.

In this paper the weight decay concept known from NN theory [1] is applied to overcome the above mentioned shortcomings when applying NNs. A preprocessing layer is inserted and the weight decay concept is applied to a diagonal matrix connecting this layer with the input. Thus, the importance of input parameters for the NN can be evaluated. This method has been applied for importance analysis within the symbolic prosody generation module and within the module for f0-generation for Mandarin language.

This paper is organized as follows. In section 2 the weight decay concept is briefly outlined. This concept will be applied in section 3. In section 4 experiments and results from symbolic prosody prediction for German language and from f0-generation for Mandarin language are presented and discussed. Finally, section 5 gives a conclusion of the presented work.

2. REGULARIZATION - WEIGHT DECAY

The weight decay concept is known in neural network theory as a type of regularization [1]. Regularization is typically used to reduce the complexity of a NN by adding a penalty term $P(\mathbf{w})$ to the error function $F(\mathbf{w})$:

$$\tilde{F}(\mathbf{w}) = F(\mathbf{w}) + \lambda \cdot P(\mathbf{w}) \qquad (1)$$

with $\mathbf{w}$ denoting a vector containing all weights in the NN. λ controls the influence of the penalty term. In the scope of this work $P(\mathbf{w}) = \sum_i w_i^2$, which is known as standard weight decay ($i = 1, \ldots,$ number of weights). λ is then typically referred to as decay rate. For the case of standard weight decay the extended error function $\tilde{F}(\mathbf{w})$ reads as $\tilde{F}(\mathbf{w}) = F(\mathbf{w}) + \lambda \cdot \sum_i w_i^2$.
In our application the standard weight decay principle is not applied to all weights within the NN. Instead certain weights are selected and form a set $\mathcal{W}_{sel}$. For this case the extended error function reads as

$$\tilde{F}(\mathbf{w}) = F(\mathbf{w}) + \lambda \sum_{\{k|w_k \in \mathcal{W}_{sel}\}} w_k^2 \quad . \qquad (2)$$

During training, weights will be adopted on the basis of this function (using gradient descent):

$$\tilde{\mathbf{w}}^{i+1} = \tilde{\mathbf{w}}^{i} - \eta \nabla \tilde{F}(\mathbf{w}) \qquad (3)$$

$$= \quad \tilde{\mathbf{w}}^i - \nabla \left[\eta F(\mathbf{w}) + \eta\lambda \sum_{\{k|w_k \in \mathcal{W}_{sel}\}} w_k^2 \right] \quad (4)$$

The parameter η is generally referred to as learning rate and controls the step size used to adapt the weights. In our case η and λ are kept constant in all steps i.

3. APPLICATION OF WEIGHT DECAY

3.1. Symbolic Prosody Unit

To find out which of the l input parameters are important to the NN for a specific task, a preprocessing layer is inserted between the input and auto-associator classificator network (for preprocessing see [9], for auto-associator classificator network see [8]). The l input signals $\mathbf{x} \in \mathcal{R}^l$ are propagated to this preprocessing layer via a diagonal matrix $\mathbf{w}_{diag} = \text{diag}(w_1, \ldots, w_l)$ to give the output signals $\mathbf{x}' \in \mathcal{R}^l$ of the preprocessing layer. Figure 1 shows the resulting network architecture. The weight decay concept is

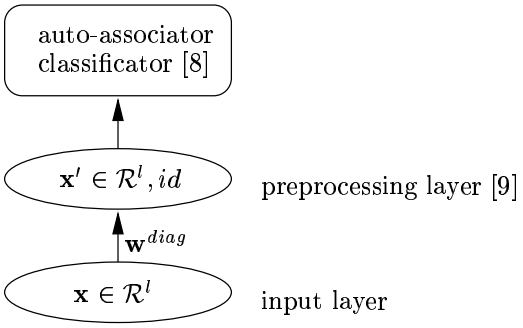


Figure 1: The auto-associator classificator architecture extended by a preprocessing layer.

applied only to weights of the diagonal matrix. Thus, $\mathcal{W}_{sel}$ contains all elements of $\mathbf{w}_{diag}$: $\mathcal{W}_{sel} = \mathcal{W}_l = \{w_1, \ldots, w_l\}$. For the neurons of the preprocessing layer, the identity function is chosen as activation function. This way possible difficulties during the learning process are avoided (such as drying out of the error signal which is known to arise when many hidden layers with $tanh(\cdot)$ as activation functions are applied in a row).

At the beginning of the training process all elements of the diagonal matrix are initialized with 1. Thus, the input signals are transferred to the hidden layer without modification. In each training epoch i eq. (3) is applied to obtain the new values for epoch $(i+1)$ for the weights in the diagonal matrix. It is important to carefully choose the decay rate λ. λ should generally be as small as possible. This way the influence of the learning rate η on weight adaptation (eq.(3)) is stronger than the influence of the decay rate λ. Therefore, non-linear relations hidden in the data can be captured. On the other hand λ should be large enough, so that it effects the weights in the diagonal matrix. After several training epochs and application of eq. (3) to the weights in $\mathcal{W}_l$, the following behavior is observed: For some elements of $\mathcal{W}_l$ the influence of the learning rate η is stronger than the influence of the decay rate λ. For other elements of $\mathcal{W}_l$, however, the influence of the decay rate λ is stronger than the influence of the learning rate η. By choosing λ/η right, some weights can be pushed towards zero, while others range higher. Those weights close to zero, or below a certain threshold, are considered to be of less importantance to the training success of the auto-associator classificator network. All weights of the auto-associator classificator network are trained without the penalty term $P(\mathbf{w})$ of eq. (1) at the same time as the weights in $\mathcal{W}_l$. The concept of adding a preprocessing layer connected to the input via a diagonal matrix is applied for importance analysis of POS tags and for analysis of the necessary size of the POS context window.

Figure 2 shows the NN architecture used for importance analysis of POS tags. The input neurons are split

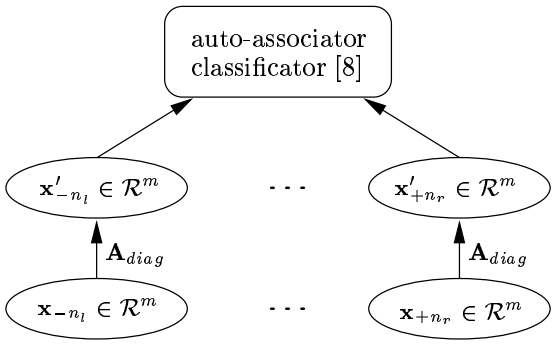


Figure 2: Each POS position is connected via a shared weight matrix $\mathbf{A}_{diag}$.

into clusters, such that each cluster represents one position in the POS context window. For symbol coding a 1-out-of-m code is used. Therefore, each cluster contains m neurons, representing the m possible tags at position n_i ($\mathbf{x}_{n_i} \in \mathcal{R}^m, \forall n_i$). n_i can take on integer values from $-n_l$ to $+n_r$, representing the POS sequence from the left (n_l steps to the left) and right (n_r steps to the right) context. The diagonal matrices $\mathbf{A}_{diag} = \text{diag}(w_1^A, \ldots, w_m^A)$ in figure 2 are all shared weight matrices. This is necessary to ensure that a certain weight in $\mathcal{W}_{sel} = \mathcal{W}_m^A = \{w_1^A, \ldots, w_m^A\}$ is effected in all possible positions within the POS context window. Since every weight in $\mathcal{W}_m^A$ corresponds to exactly one tag in the tag-set, a direct interpretation of the importance of a certain tag is possible.

For analysis of the necessary size of the POS context window no clustering of input neurons and no weight sharing is used, i.e. the inputs are all treated individually as indicated in figure 1. To detect which POS positions within the POS context window are important, the mean weight value of the weights at a certain position is computed for each POS position. High mean values indicate that the corresponding position is important, low mean values indicate that the position is less important.

3.2. F0-Generation Unit

The method related to figure 1 has also been applied to analyze the importance of parameters for f0-generation for Mandarin language. For this application the auto-associator classificator network in figure 1 is replaced by our standard NN architecture for f0-generation for Mandarin

language [12]. The procedure is analog to the procedure for analysis of the necessary size of the POS context window for the application in symbolic prosody. The input parameters in this case represent phonetic information. The mean values of the weights in the diagonal matrix for certain groups of input parameters are calculated. These mean values represent the importance of the group of input parameters.

4. EXPERIMENTS AND RESULTS

4.1. Symbolic Prosody

In this section results from the importance analysis of POS tags and from the analysis of the necessary size of the POS context window are presented. Further, results are presented that demonstrate that the proposed method can be used to optimize the POS context window length. It has also been shown experimentally that the number of tags in a given tag-set can be reduced without performance loss All experiments are related to the problem of major phrase break prediction as addressed in [8].

The goal in the experiment related to importance analysis of POS tags is to detect which tags in the tag-set of our prosody training corpus [6] are important and which ones are less important and can be combined with other tags. The corpus has been tagged semi-automatically and the tags have been hand-corrected. There are 35 tags in the tag-set ($m_l = 35$). However, our own tagger can currently only predict 14 tags ($m_s = 14$). The information gained from the importance analysis with the proposed method will be used to choose tags that need to be added to the small tag-set.

The plot in figure 3 shows the values $w_i \in \mathcal{W}^A_{m_l}$. Some relatively high values can clearly be identified (e.g. positions 8 and 30). These are the positions of punctuation marks in the tag-set, which is an obvious result. The next lower values represent tags such as proper names and nouns in all cases (German: Kasus). The low values represent adverbs, adjectives, and pronouns.

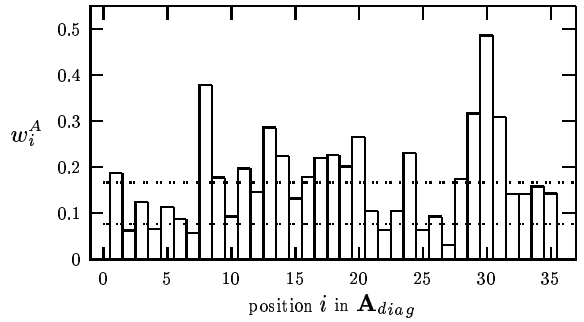


Figure 3: Analysis of the weights in $\mathcal{W}^A_m$.

The proposed method only identifies less important tags. The mapping (combination of tags), however, need to be done by hand. Two mapping strategies have been applied. In the first, only tags with $w_i < 0.075$ (see lower dotted line in figure 3) are mapped. In a second mapping strategy, groups of tags such as adjectives in all possible cases have been examined. If the tags in such a group were found to be all equally important, associated with a low average value of the group below 0.165 (see upper dotted line in figure 3), they were all mapped into one group. If, however, some tags in a group are associated with a high w_i, while others in the group range much lower, the tags with the low values get mapped and the ones with high values not. This can e.g. be observed at tags of the group pronouns ($i = 25, \ldots, 28$) where tags 25-27 would get mapped, while tag 28 would not get mapped. In this concrete case this would mean that it is important to the NN whether a pronoun occours in the case nominative or not.

Results from experiments to evaluate the POS context window length are displayed in figure 4. The plot shows

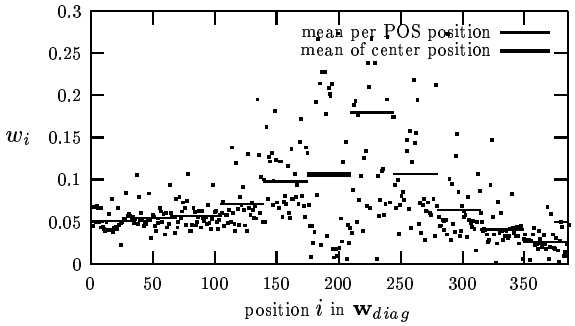


Figure 4: The values of $w_i \in \mathcal{W}_l$ ordered according to the position in the POS context window.

the values $w_i \in \mathcal{W}_l$ of the corresponding position i in the diagonal matrix ($n_l = n_r = 5, m = 35$). The bars indicate the mean values of the weights per POS position in the POS context window. The bar representing the center of the POS context window is printed bold. As can be seen the position one to the right is associated with the highest mean value. This indicates that it is most important for phrase break prediction. Further, it can be seen that POS positions of more than two to the left and more than three to the right have low associated mean values and are thus not important for phrase break prediction.

data set	35 tags	mapping 1	mapping 2
validation	92.0%	92.4%	91.9%
generalization	92.6%	92.6%	92.3%

Table 1: Results for the mapped tag-sets.

In the first column of table 1 the results achieved with the optimized POS context window length of two POS tags to the left and three POS tags to the right are displayed. With a POS context window length of five POS tags to the left and five to the right, prediction accuracy ranged lower: for the generalization data prediction accuracy was 92.2% and 92.3% for the validation data.

Table 1 also shows the prediction results for the two mapping strategies explained above for our validation and generalization data. All experiments for table 1 are performed with the optimized POS sequence length of two to the left and three to the right. In the first mapping (mapping 1), six tags were mapped, reducing the tag-set size by four (the tags were mapped into two groups). In the second mapping (mapping 2), 20 tags were mapped, reducing the

tag-set size by 14 (the tags were mapped into six groups). As can be seen, mapping 1 improved the prediction result for the validation data slightly while it performed equally on the generalization data. With mapping 2 insignificantly lower prediction accuracy was achieved. This shows that the proposed method can be used for tag-set size reduction without performance loss.

4.2. Mandarin f0-generation

In this section results from the importance analysis of input parameters for f0-generation for Mandarin language are discussed. For f0-generation for Mandarin language the input parameters are organized in groups representing certain phonetic information (for detailed describtion of input parameters see [12][3]). These groups contain features. Features can be associated with several input parameters. A group would e.g. describe all information related to tones. A feature would be e.g. the tone type of the center syllable (as in symbolic prosody a certain context window is used). The mean values for the parameters of a feature are calculated and serve as a basis for importance analysis.

Features in three groups of input information are associated with a low mean value: first, features in the group describing the duration of the current and surrounding syllables. Second, features in the group describing the tones of the current and surrounding syllables, and third, features in the group describing the internal structure of the current syllable.
In the duration group, it is observed that the mean values for features describing the duration of syllables to the right range low. In the tone group the features describing the tones of the syllables one and two to the right and second to the left are associated with low mean values. Normally, we thought that the current tone, the left tone and the right tone context plays an important role in f0-generation for Mandarin language. From our result, it shows at least the current tone and one to the left tone context are essential in F0 prediction for the use within our NN architecture. In the syllable structure group features describing the head structure of the syllable have low mean values.
If the above explained features from the three groups are not used as input for the NN architecture, the squared error of the NN architecture could is reduced by 3.1%. This shows that the method is appropriate to find out input parameters that the NN architecture used for f0-generation does not need.

5. CONCLUSION

In this paper the weight decay concept known from neural network theory has been applied in the symbolic prosody module and for Mandarin f0-generation.
Preprocessing layers connected to the inputs of specialized NN architectures via diagonal weight matrices were used. By applying the weight decay concept to these diagonal matrices, in the case of symbolic prosody, the importance of tags and POS context length could be evaluated. This way the tag-set size could be reduced by 4 tags with no performance loss and by 14 tags with insignificant performance loss.
For the case of Mandarin f0-generation, the input parameters could be analyzed and optimized. If the optimized input parameters are used as input for the NN architecture used for f0-generation [12], the squared error is reduced by 3.1% in our application.

This demonstrates that the proposed method is appropriate to detect important and less important linguistic and phonetic information contained in the input parameters of a NN. The significance of the gained information can, however, possibly only be seen in the context of the used NN architecture.

6. REFERENCES

[1] Christopher M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[2] Alan W Black and Paul Taylor. Assigning phrase breaks from part-of-speech sequences. In *Eurospeech*, 1997.

[3] Ralf Haury and Martin Holzapfel. Optimization of a neural network for speaker and task dependent f0-generation. In *ICASSP*, 1998.

[4] Julia Hirschberg. Pitch accent in context: Predicting prominence from text. *Artificial Intelligence*, 63:305–340, 1993.

[5] Julia Hirschberg and Pilar Prieto. Training intonational phrasing rules automatically for english and spanish text-to-speech. *Speech Communication*, 18:281–290, 1996.

[6] http://www.phonetik.uni-muenchen.de/Bas/.

[7] K.Ross and M. Ostendorf. Prediction of abstract prosodic labels for speech synthesis. *Computer Speech and Language*, 10:155–185, 1996.

[8] Achim F. Müller, Hans Georg Zimmermann, and Ralph Neuneier. Robust generation of symbolic prosody by a neural classifier based on autoassociators. In *ICASSP*, 2000.

[9] Ralph Neuneier and Hans Georg Zimmermann. How to train neural networks. In G. B. Orr and K.-R. Müller, editors, *Neural Networks: Tricks of the Trade*, pages 373–423. Springer Verlag, Berlin, 1998.

[10] M. Ostendorf and N. Veilleux. A hierarchical stochastic model for automatic prediction of prosodic boundary location. *Computational Linguistics*, 20:27–54, 1994.

[11] Gerit P. Sonntag, Thomas Portele, and Barbara Heuft. Prosody generation with a neural network: Weighing the importance of input parameters. In *ICASSP*, 1997.

[12] Jianhua Tao, Cai Lianhong, Martin Holzapfel, and Herbert Tropf. A neural network based prosodic model of mandarin tts system. In *ICSLP*, 2000.

[13] Michell Q. Wang and Julia Hirschberg. Automatic classification of intonational phrasing boundaries. *Computer Speech and Language*, 6:175–196, 1992.

[14] Christina Widera, Thomas Portele, and Maria Wolters. Prediction of word prominence. In *Eurospeech*, 1997.