Pattern Recognition Using Finite-Iteration Cellular Systems

(Invited Paper)

Maciej Ogorzałek
Department of Electrical Engineering
AGH University of Science and Technology
Kraków, Poland
Email: maciej@agh.edu.pl

Christian Merkwirth
Department of Information Technologies
Jagiellonian University
Kraków, Poland
Email: ChristianMerkwirth@web.de

Joerg Wichard
Institute of Molecular Pharmacology
Berlin, Germany
E-mail: JoergWichard@web.de

***Abstract*— Cellular Systems are defined by cells that have an internal state and local interactions between cells that govern the dynamics of the system. We propose to use a special kind of Cellular Neural Networks (CNNs) which operates in finite iteration discrete-time mode and mimics the processing of visual perception in biological systems for digit recognition. We propose also a solution to another type of pattern recognition problem using a non-standard cellular neural networks called Molecular Graph Networks (MGNs) which offer direct mapping from compound to property of interest such as Physico-Chemical, Toxicity, logP, Inhibitory Activity MGNs translate molecular topology to network topology. We show how to design/train by backpropagation CNNs and MGNs in their discrete-time and finite-iteration versions to perform classification on real-world data sets.**

I. Introduction

The discrete time version (DT-CNN) of the CNN principle has been introduced [1]. In both analog and discrete-time networks the underlying principle is that computation is done by system dynamics, i.e. the response of the network to given stimuli is defined by the limit behavior in time – convergence of all responses to some fixed point [2]. The achieved steady state is predefined by the connection templates and by the the network input and initial conditions. Recently we have proposed a new mode of operation of DT-CNN — namely the finite iteration DT-CNN [3] and its stationary version [4]. The network was used as a one-step-forward computing engine. Efficient procedures were proposed for template design and optimum selection of number of time steps.

II. The DT-CNN classifier

To build a pattern classifier we use spatio-temporal dynamical system with cells on an square grid - the size of DT-CNN receptive field is fixed here to 16 by 16 cells. State evolution of DT-CNN cells y_{ij}^t for iterations $t = 0, \ldots, T-1$:

$$\begin{aligned} x_{ij}^{t+1} &= \sum_{l,m=-\frac{K-1}{2}}^{\frac{K-1}{2}} (A_{l,m}^t y_{i+l,j+m}^t \\ &\quad + B_{l,m}^t u_{i+l,j+m}) + b^t \\ y_{ij}^{t+1} &= \sigma(x_{ij}^{t+1}) \end{aligned}$$

In our finite-iteration DT-CNN we use different template weights A^t, B^t and offsets b^t for each iteration. The number of template layers (CNN iterations) is limited to T. Input pattern is presented during all iterations through u_{ij} and A^0 is not used since $y_{ij}^0 = 0$ by initialization. Elements outside the boundary are treated as zero. The input pattern is presented to DT-CNN during every iteration by means of the B template.

A. Deriving the decision variable

(DT-)CNN is a spatio-temporal processing engine. We need to convert the spatial output y_{ij}^T into scalar output value z suitable as decision variable. Simple averaging of all 256 cell states y_{ij}^T

$$z = \frac{1}{256} \sum_{ij} y_{ij}^T$$

Resulting z is position independent

III. Molecular Graph Networks

Molecular Graph Networks [5] are discrete-time (cellular) spatio-temporal dynamical systems in which in contrast to CNN, the network topology changes for every compound processed. Compound can be described as graph where each atom is a node and each chemical bond is an edge - thus molecular structure is translated into cellular network structure: Each atom becomes a cell, each bond a local interaction. Weights do not depend on the position in graph but on element and bond types. Each atom i has an initial state y_i^0 depending on its element type e_i (6 for C, 92 for U) Evolution of atoms states y_i^t for iterations $t = 0, \ldots, T-1$ is described by equations

$$\begin{aligned} x_i^{t+1} &= \sum_{j \text{ is connected to } i} A_{e_i,b_{ij}}^t y_j^t + c_{e(i)}^t \\ y_i^{t+1} &= \sigma(x_i^{t+1}) \end{aligned}$$

A. Molecular Graph Networks - Computation

To calculate output for one compound different weights tables A^t and offsets c^t have to be computed for each iteration. b_{ij} is the bond table storing bond type (single, double, aromatic) between atom i and atom j in this molecule. State information of each atom spreads through the network along the bonds. After T iterations information about each atom is propagated along T edges. Detection of functional groups is possible without explicitly defining the groups. To convert the

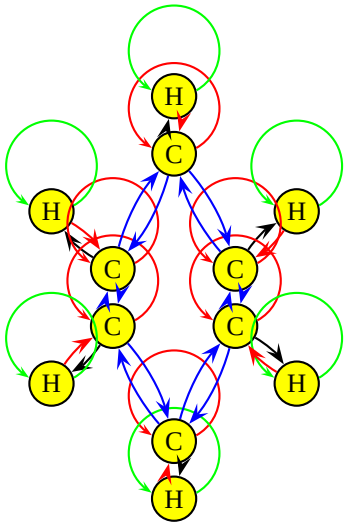

Fig. 1. Typical molecular structure mapped onto Molecular Graph Network

atom states y_i^T into scalar output value z (e.g. decision variable for classification) averaging of atom states y_i^T weighted by element type is performed

$$z = \frac{1}{\#\text{atoms}} \sum_i g_{e_i} y_i^T$$

Result z is independent of atom positions, orientation of molecule etc. No successful methods for training these networks were proposed so far. Several feature nets can be combined to one large network. Output of each feature net z_n is fed into conventional fully connected neural network (supervisor network). The training problem becomes very expensive for large networks with many weights ($> \ 50000$)

IV. Training as Optimization Problem

Both for CNN and MGN we are cellular systems with many parameters (weights. These parameters govern dynamical behavior of these systems. We want system output z_i to match observed outputs Y_i (training data). Supervised learning task can be treated as optimization problem.

Sample-wise loss function $L(Y_i, z_i)$ that is added to total loss $\mathbf{L} = \sum_i L(Y_i, z_i)$. Numerical optimization algorithm that is able to cope with many parameters is needed.

ϵ-insensitive absolute loss is combination of absolute loss (L_1 norm) with ϵ-insensitivity. Very high setting of ϵ is used for binary classification. Smaller ϵ degrades classification performance. Patterns are classified correctly as soon as sign of decision variable is correct. Training patterns that can't be correctly classified have only linear contribution to the loss.

V. Recognition of handwritten digits

As an example for a multi-class classification problem we consider recognition of isolated handwritten digits. This is a benchmark problem in the neural network and machine learning community [6], [7]. DT-CNNs featuring non-stationary and stationary templates are investigated. The results are compared with a polynomial support vector classifier [8].

A. The ZIP Code Data Set

The ZIP Code Data Set consists of normalized handwritten digits, automatically scanned from envelopes by the U.S. Postal Service. The images, originally binary and of different sizes and orientations, have been deslanted and normalized in size, resulting in 16 by 16 gray-scale images (see [6]). Some examples of these images are shown in Figure 2. The entire data set consists of 7291 training observations and 2007 test observations. The 2007 samples of the ZIP Code test set are

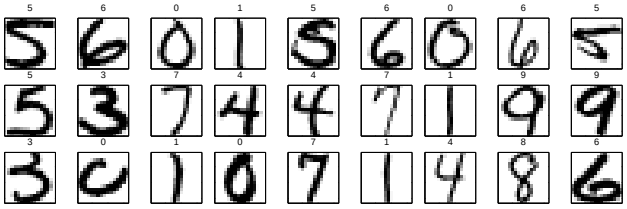

Fig. 2. Examples of input patterns from the ZIP code dataset. Each image is a 16×16 grayscale representaion of a handwritten digit, the original labels are shown on top of each.

neither used for training nor to derive stopping criteria nor for model selection in classifier ensembling. The patterns are presented during the training as pairs together with the desired output (the class label out of 0,1,...,9).

B. Classification with non-stationary templates

First we investigate the DT-CNN classifier with non-stationary templates. In the following simulations we choose a template size of $K = 5$. With a standard nearest-neighbor coupling $K = 3$ we could not achieve adequate classification performance. For template size $K = 7$ in term we observed a gap emerging between training and test errors. We interprete this as an overfitting effect, for the number of template weights has the same magnitude as the number of training samples. One can observe that with T ranging from 1 to 10 layers, the classification rate increases significantly. For more template layers the rate saturates around 96.5%. The stochastic gradient decend went in the non-stationary case over 50 epochs, which was suffient to achieve good classification results. For comparison, we constructed a polynomial SVM classifier of degree 4 with $C = 100$ (see [8]) on the ZIP Code training set and applied it to the test set. The classification rates are shown in figure 3. In figure 4 we show an example of the internal states of the DT-CNN classifier trained to recognize the digit **3**. The upper row shows examples of input patterns. The rows below show successive iterations- up to 10. Eventually the classifier averages over the 256 entries of these 16×16 pictures.

C. Classification performance

To summarize the performance of the DT-CNN digit recognition system we can confirm that the optimal template size K should be fixed to 5. Standard nearest-neighbor coupling $K = 3$ was too small while $K = 7$ provokes overfitting. For $T = 10$ we have total of 4850 free parameters to compute.

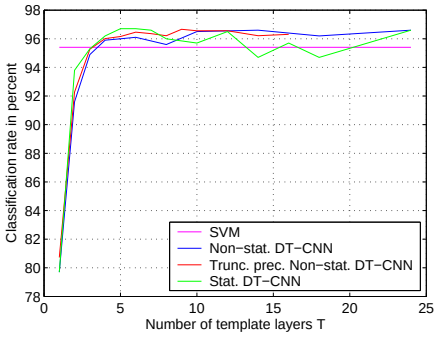

Fig. 3. For T from 1 to 5 layers classification rate increases significantly. At higher T classification rate saturates around 96.5%.

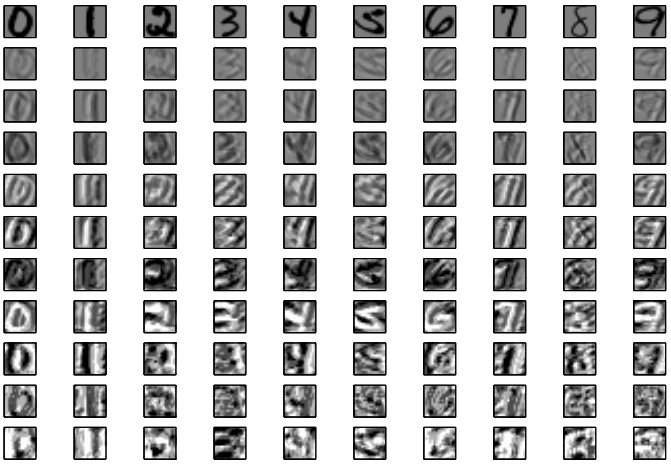

Fig. 4. Variation the internal states of a DT-CNN classifier trained to recognize the digit **3**. The last column shows the final output after ten iterations.

Polynomial SVM classifier of degree 4 and $C = 100$ yields classification rate of 95.4%. We obtained better performance of the classifier however igher computational effort was required for training ensembles of DT-CNNs than constructing SVM classifier.

VI. NCI AIDS Antiviral Screen Data Set

We considered a data set of more than 42000 compounds from the DTP AIDS antiviral screen data set of the NCI Open Database `http://cactus.nci.nih.gov/ncidb2/download.html`. The antiviral screen utilized a soluble formazan assay to measure the ability of compounds to protect human CEM cells [9] from HIV-1-induced cell death. The activities of the compounds tested in the assay fall into three classes: confirmed active (CA) for compounds that provided 100 % protection, moderately active (CM) for compounds that provided more than 50 % protection, and confirmed inactive (CI) for the remaining compounds. The data set consisted of 42682 2D structures with AIDS test data as of October 1999 and was provided in SDF format. 41179 compounds were confirmed inactive, 1080 compounds were confirmed moderately active and only 423 compounds were confirmed active.

When using classification loss, we assigned a training output of zero to all compounds of class CI, 0.8 to all compounds of class CM and 1.0 to all compounds of class CA. A value of 0.8 instead of 0.5 was chosen for the confirmed moderately active compounds to indicate the algorithm that these compounds should be considered rather being active than being inactive. Using classification loss simplifies the training since only one ensemble of classifiers has to be trained on all compounds of the training data set. However, converting the continuous output of the classifier ensemble back into class labels necessitates the choice of two thresholds τ_1 and τ_2 in order to discriminate between the three classes. We bypassed this problem by either considering classes CI and CM as inactive and class CA as active or considering only class CI as inactive and classes CM and CA as active, thus converting the problem to a binary classification problem.

The data set has been randomly partitioned into a training set of 35000 compounds and test set of 7682 compounds. We constructed an ensemble of 15 MGNs. Each MGN consisted of 19 individual feature nets with iteration depths ranging from 3 to 10 and a supervisor network with 24 hidden layer neurons. The MGNs were trained by stochastic gradient descent with a fixed number of 10^6 gradient calculations. The global step size μ was decreased by a factor of 0.8 every 70000 gradient updates. Each MGN was trained on a random 80 % of the 35000 training samples. Thus the OOT output for every sample of the training set was computed by averaging over three models, the output for the held-out test set by averaging over all 15 models of the ensemble of MGNs.

A. Results and Discussion

Results for the classification experiments on NCI data set with classification loss function are given in Figures 5 and 6. Both Figures display two pairs of ROC curves. The lower pair of ROC curves in Figure 5 was obtained by using the ensemble of classifiers to discriminate between CI on the one hand and CA and CM on the other, while the upper pair details the ROC curves when using the same ensemble of classifiers to discriminate between CI and CM on the one hand and the confirmed actives CA on the other. The remarkable coincidence of the curves obtained by validation on the training part and from the held-out test part of more than 7000 compounds indicates that the validation was performed properly and does not exhibit overfitting. Figure 6 details the ROC curves for false positives rates up to 5 %.

AUC values of the approach employing classification loss and of the one-versus-all approach are similar to the best results of several variants of a recent classification method based on finding frequent subgraphs[10]. (experiments H2 and H3 when omitting class CM from the test set for the ensemble constructed to discriminate CA versus the two other classes). Wilton et al.[11] compare several ranking methods for virtual screening on an older version of the NCI data set. The best performing method there, binary kernel discrimination, is able to locate 12 % of all actives (CM and CA pooled) in the first 1 % and 35 % of all actives in the first 5 % of the

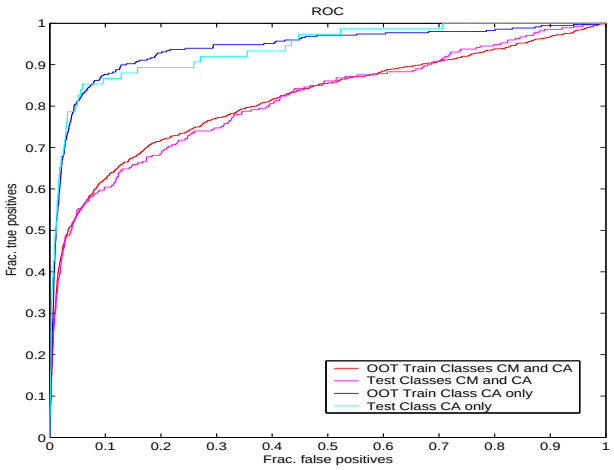

Fig. 5. ROC curves for the classifiers constructed on the NCI AIDS Antiviral Screen Data Set with ϵ-insensitive absolute loss. The Figure displays two pairs of ROC curves. In this computational experiment we trained an ensemble of molecular graph networks on a data set consisting of three classes of molecules (CI, CM and CM). To be able to generate ROC curves, we had to reduce the number of classes to two by pooling the molecules of two classes into a single class. The lower pair of ROC curves was obtained by using the ensemble of classifiers to discriminate between CI as one class and CA and CM as second class, while the upper pair details the ROC curves when using the same ensemble of classifiers to discriminate between CI and CM as one class and the confirmed actives CA as the second. The AUCs of the respective pairs of curves are 0.82 resp. 0.81 for classification of CI versus CA and CM and 0.94 resp. 0.94 for classification of CI and CM versus CA.

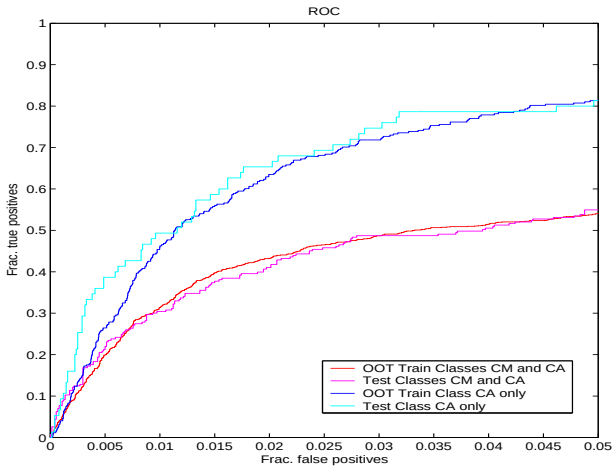

Fig. 6. Zoom of the ROC curves given in Figure 5 for false positives rates up to 5 %. Larger false positive rates usually involve screening of too many compounds of a large database and are therefore of lesser interest in drug discovery or lead optimization. At a false positive rate of 5 % both validation and test true positive rate for detecting class CA reach 80 %. At a false positive rate of 1 % both rates exceed 40 % which would result in more than 40% of all actives found when screening up to this threshold.

ranked NCI data set. Results of the one-against-all approach tend to be slightly better than that of the first approach, which might be caused by the more straightforward way in which the multi-class problem is converted into more easily to solve binary classification problems. The universally better results for the held-out part of the NCI data set could be a result of the ensembling. While for the OOT validation the output for every sample is computed as average of only two independent MGNs, for every sample of the test fraction the average is computed over six MGNs, resulting in an increased ensemble gain and thus in an improved prediction.

VII. Conclusions

Spatio-temporal dynamical system with local interactions can be used for pattern recognition and statistical learning in different application domains. Thay can be seen as an intermediate between statistical learning and explicit physical simulation. Weights (templates) can be found by an optimization problem in which gradient can be calculated by back-propagation. Stochastic gradient descent is fast and yields good solutions while ϵ-insensitive loss function with high ϵ is more appropriate for classification than quadratic loss. Boosting and ensemble methods can be used for improving generalization performance.

Acknowledgment

Parts of this work are supported by the Deutsche Forschungsgemeinschaft (DFG) grant LE 491/11-1 (*Analyse von Wirkungszusammenhängen bei der Medikamentenentwicklung*) and by the Research Training Network *COSYC of SENS* No. HPRN-CT-2000-00158 within the 5th Framework Program of the EU. We thank the people of the AGH University of Science and Technology and the Jagiellonian University in Kraków and of the Max-Planck Institute for Computer Science in Saarbrücken for support.

References

[1] H. Harrer and J. Nossek, "Discrete-time cellular neural networks," *Int. J. Circuit Theory and Applications*, vol. 20, pp. 453–467, 1992.

[2] A. K. S. Arik and F. A. Savaci, "Global asymtotic stability of discrete-time cellular neural networks," in *Proceedings of IEEE Int. Workshop on Cellular Neural Networks and Their Applications*, 1998, pp. 52–55.

[3] C. Merkwirth, M. Ogorzalek, and J. Wichard, "Finite iteration dt-cnn - new design and operation principles," in *Proc. ISCAS*. Vancouver, Canada: IEEE, 2004.

[4] J. Wichard, M. Ogorzalek, C. Merkwirth, and J. Bröcker, "Finite iteration dt-cnn with stationary templates," in *Proc. IJCNN*, Budapest, Hungary, 2004.

[5] C. Merkwirth and T. Lengauer, "Automatic generation of complementary descriptors with molecular graph networks," 2004.

[6] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. [Online]. Available: citeseer.nj.nec.com/lecun98gradientbased.html

[7] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, ser. Springer Series in Statistics. Springer-Verlag, 2001.

[8] C. C. Chang and C. Lin, "Libsvm - A library for support vector machines (http://www.csie.ntu.edu.tw/˜cjlin/libsvm)."

[9] O. Weislow, R. Kiser, D. Fine, J. Bader, R. Shoemaker, and M. Boyd, "New soluble formazan assay for hiv-1 cytopathic effects: application to high flux screening of synthetic and natural products for aids antiviral activity," *J. Nat. Cancer Inst.*, vol. 81, pp. 577–586, 1989.

[10] M. Deshpande, M. Kuramochi, and G. Karypis, "Frequent sub-structure-based approaches for classifying chemical compounds," in *Proceedings of the Third IEEE International Conference on Data Mining ICDM 2003*, Melbourne, Florida, November 2003, pp. 35–42.

[11] D. Wilton, P. Willett, K. Lawson, and G. Mullier, "Comparison of ranking methods for virtual screening in lead-discovery programs," *J. Chem. Inf. Comput. Sci.*, vol. 43, pp. 469–474, 2003.