Fast training of self organizing maps for the visual exploration of molecular compounds

Proceedings of International Joint Conference on Neural Networks, Orlando, Florida, USA, August 12-17, 2007

Fast Training of Self Organizing Maps for the Visual Exploration of Molecular Compounds

Antonino Fiannaca, Giuseppe Di Fatta, Riccardo Rizzo,
Alfonso Urso, and Salvatore Gaglio

***Abstract*— Visual exploration of scientific data in life science area is a growing research field due to the large amount of available data. The Kohonen's Self Organizing Map (SOM) is a widely used tool for visualization of multidimensional data. In this paper we present a fast learning algorithm for SOMs that uses a simulated annealing method to adapt the learning parameters. The algorithm has been adopted in a data analysis framework for the generation of similarity maps. Such maps provide an effective tool for the visual exploration of large and multi-dimensional input spaces. The approach has been applied to data generated during the High Throughput Screening of molecular compounds; the generated maps allow a visual exploration of molecules with similar topological properties.**

The experimental analysis on real world data from the National Cancer Institute shows the speed up of the proposed SOM training process in comparison to a traditional approach. The resulting visual landscape groups molecules with similar chemical properties in densely connected regions.

I. Introduction

A crucial step in drug discovery remains the so-called High Throughput Screening (HTS) and the subsequent analysis of the generated data. In this screening, hundreds of thousands of potential drug candidates are automatically tested for a desired activity, such as blocking a specific binding site or attachment to a particular protein. This activity is believed to be connected to, for example, the inhibition of a specific disease. Once all these compounds have been automatically screened, a large amount of data have to be analyzed and explored in order to select a few hundred promising candidates for further, more careful and cost-intensive analysis. This step is critical for the success of the entire drug discovery process. Recent approaches based on data mining techniques focus on the analysis of the molecular structure and the extraction of pieces of molecules that are correlated with activity. Such fragments can be used to directly identify groups of promising molecules (clustering). They can also be used to predict activity in other compounds (classification) [1] and to guide the synthesis of new ones. The number of these relevant molecular fragments is often very large and they cannot be directly visualized nor exhaustively explored by the biochemists. However, these fragments can be used to identify groups of molecules with similar characteristics. In this context, visualization and indexing techniques for large data spaces can provide a powerful tool for the overall HTS analysis process. A general data mining framework to generate similarity maps of molecular compounds has been introduced in [2]. The approach uses the frequent molecular fragments to define a high-dimensional feature space and adopts the SOM for a 2-dimensional representation. In the present work, we introduce a new training algorithm based on the simulated annealing technique in order to speed up the unsupervised SOM learning process. The overall data analysis process has been applied to a well-known set of real molecular compounds, the NCI HIV screen dataset. The resulting map has produced distinct clusters of similar compounds. In order to verify the quality of the map we have adopted a publicly available classification of a small subset of the compounds.

In the next section, we briefly discuss some related works in the field of the visual exploration of biochemical data and in field of the training of Self Organizing Maps. In section III, we describe the overall framework for the generation of maps of molecular compounds. In section IV, we present the proposed learning algorithm. In section V, we present and discuss the experimental results. Finally, in the last section we provide conclusive remarks.

Giuseppe Di Fatta is with School of Systems Engineering, The University of Reading, United Kingdom.

Antonino Fiannaca, Riccardo Rizzo and Alfonso Urso are with ICAR-CNR, Consiglio Nazionale delle Ricerche, Italy.

Salvatore Gaglio is with DINFO, University of Palermo, Italy.

II. Related Work

Self-Organizing Maps have been used extensively in chemistry [3], [4] and biology applications, with analysis and classification purposes, in the field of Quantitative Structure-Activity Relationships (QSAR) [5], [6], [7]. In other cases (e.g., [8]), the SOM have been used to select the best subset of features to carry out a subsequent QSAR analysis. In some application related to the gene expression clusterization and visualization ([9], [10]) the SOM neural network was used as a tool for clustering and visualization in order to obtain an *"executive summary"* of a massive gene expression data set.

Several approaches for improving performances of SOM have been described in the literature. The key issue is to determine which parameters and conditions need to be considered, in order to obtain a well trained map. In this context a SOM is well trained if clustering is achieved in a short time and, at the same time, it creates a projection of data into the map strongly related to the distribution of data in the input space. One such attempt was done in auto-SOM [11], where Kalman filters have been used to guide the weight vectors toward the center of their respective Voronoi cells in input spaces. Using this method, it is possible to automatically estimate the learning parameters during

the training of SOMs. Indeed the introduction of Kalman filters leads the network to a good training, but it is more computationally expensive than the classic SOM. In [11] the authors confirm that the batch-learning SOM algorithm needs fewer learning steps than the auto-SOM during training process, but they also assert that batch-learning techniques get stuck more often in local minima. Our work aims to improve the batch-learning SOM algorithm in order to avoid local minima. A recent approach to obtain a well trained map is the parameterless SOM (PLSOM) [26]: this technique is based on the ordinary SOM algorithm and revises both classic learning rate function and neighborhood size function. Usually these functions are decreased over time and do not take into account the adaptive capabilities of the network during learning process. PLSOM calculates previous values based on the local quadratic fitting error of the map to the input space. This way only the previous local error is used during the evolution of the map. That means both the first data inputs and the initialization of the map, play a fundamental role in map evolution and, moreover, the selection of the learning rate is modified without evaluating most of neurons. To avoid this drawback, in our work we preserve the basic idea of a learning rate depending both on time and on organization of neurons in the map, but we also use a global quantization error rather than a local quadratic fitting error. An adaptation of habituation mechanism in SOMs was proposed in [28]. In this method the learning is conditioned by the frequency of input stimuli over several neurons of the map, so that the neuron that is habituated to a frequent input pattern will not learn on behalf of another neuron unit. Although the algorithm adds a new parameter, the global computational complexity is not expensive, because it avoids the distance calculation between input and the most used neuron unit in the map. The parameter habituation is a function of a local error so that the habituation is increased when the network is not ordered. Unfortunately the previous characteristic leads to deceleration of the training during the refinement phase of the learning. In the present work we introduce a variation of the standard batch-SOM that does not reduce speed in the last epochs of learning, according to clustering requirements.

III. Framework for the visual analysis of molecular compounds

The overall process of knowledge discovery for the visual exploration of molecular compounds [2] is shown in Figure 1. Input data, undirected labeled graphs representing the molecular compounds, are first processed to extract and select features that might be relevant to the application domain. The frequent molecular fragments are adopted as features of the molecules because they are likely correlated with the chemical activity and, thus, can be used to identify groups of similar compounds.

The selection of molecular fragments in a set of molecules can be formulated in terms of Frequent Subgraph Mining (FSM) in a set of graphs, in analogy to the Association Rule Mining (ARM) problem [13], [14]. Molecules are represented by attributed graphs, in which each vertex represents an atom and each edge a bond between atoms. Each vertex carries attributes that indicate the atom type (i.e., the chemical element), a possible charge, and whether it is part of an aromatic ring. Each edge carries an attribute that indicates the bond type (single, double, triple, or aromatic). Frequent molecular fragments are subgraphs that have a certain minimum support in a given set of graphs, i.e., are part of at least a certain percentage ($minSupp$) of the molecules. To find frequent molecular fragments we have adopted an algorithm based on a depth-first search strategy [15].

However, the set of all frequent fragments is enormous even for relatively small datasets: a single molecule of average size can already contain in the order of hundreds of thousands of different fragments. We, then, adopt the closed frequent subgraphs that more efficiently carry equivalent information as the frequent ones. A closed frequent subgraph is a frequent subgraph whose support is higher than the support of all its proper supergraphs.

Nevertheless, the set of features is still very large and defines a high-dimensional space of the input data, making the analysis process very difficult. Techniques from statistics, machine learning and data mining, can be applied to provide a model of the data that can be more conveniently interpreted by the user. We adopt a multi-dimensional scaling technique, the Truncated Singular Value Decomposition (TSVD) [17], to cope with such a problem.

Our ultimate goal is a visual summary of the set of compounds that provides information on their similarity. Thus, the dimensionally-reduced space is further projected into a two-dimensional gray scale bitmap by means of the Self-Organizing Map. The efficiency of the SOM training process is the object of the present work. We refer to [2] for further details of the general framework and its components.

IV. SOM Adaptive Learning Rate Algorithm

In this section an adaptive training algorithm for SOMs is introduced. Although the SOM learning procedure is well known it is necessary to briefly highlight some points of the original algorithm to better understand the proposed one.

A. Self-Organizing Map basic algorithm

Self-Organizing Maps [18] are neural structures capable of building maps of the input data, which preserve neighborhood relationships. These maps can be used as a visualization tool that allows to identify clusters or structures in input data [19]. In the following the main formulas of the SOM algorithm are reported; see [18] for further details.

At each step of the learning stage an input vector x is submitted to the neural network and the winner unit, called best matching unit (*bmu*) is selected according to:

$$bmu = \arg\left(\min_{i \in N} \|x - w_i\|\right). \quad (1)$$

Neural weights are updated using the following rule:

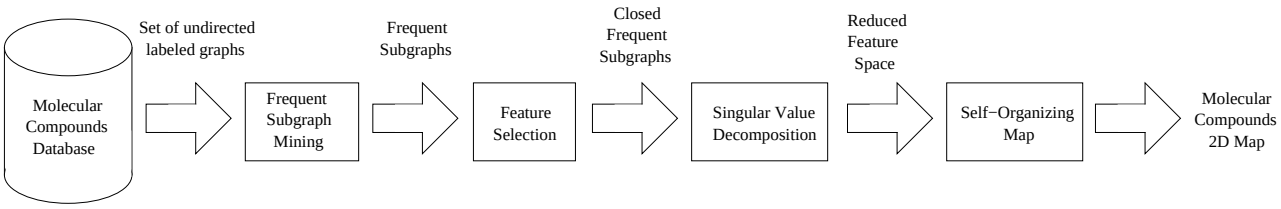


Fig. 1. Framework for the generation of the molecular compounds map

$$w_i(t+1) = w_i(t) + \alpha(t) h_{ci}(t) \left[x - w_i(t)\right], \qquad (2)$$

where h_{ci} is the neighborhood kernel around the best matching unit. One of the most common shapes of the kernel is the gaussian shape:

$$h_{ci}(t) = \exp\left(\frac{d\left(r_{bmu}, r_i\right)}{2\sigma^2(t)}\right), \qquad (3)$$

where the term $d\left(r_{bmu}, r_i\right)$ stands for the distance between the bmu unit and the generic unit i on the SOM lattice.

The learning parameter $\alpha(t)$ in eq. (2) is the learning rate factor and makes the "movement" of the units toward the input patterns less effective during time. At the end of the learning stage the network is "frozen" in the input space. The parameter $\alpha(t)$ is typically given by the formula:

$$\alpha(t) = \alpha_{MAX}\left(\frac{\alpha_{MIN}}{\alpha_{MAX}}\right)^{\left(\frac{t}{t_{max}}\right)}. \qquad (4)$$

These equations can be used in two ways: the on-line or the batch learning. Advantages and drawbacks of these implementations has been discussed by [27]; algorithm introduced in this work can be applied to both learning methods. In this paper the so-called batch learning algorithm is used, a SOM trained with the batch-algorithm is referred as batch-SOM. A SOM with more units than clusters is often referred as Emergent-SOM (E-SOM) [20].

B. The simulated annealing learning algorithm

In this section, we present a variant of the training algorithm based on simulated annealing. The algorithm has been developed in order to both improve the quality of learning and speed up the training process of the SOM. The proposed algorithm uses an heuristics process to reach the above goals, thus we do not have an objective measure against which we can compare different weight configurations. To solve this issue, a study about transformation of learning rule to some kind of gradient descent has been carried out by [21], in which an energy function during the learning phase has been introduced.

Simulated annealing (SA) is an optimization method that performs well with non-linear functions. It is typically used for large scale problems, especially the ones where a desired global minimum is hidden among many local minima. At each step the SA algorithm replaces the current solution by a random nearby solution, chosen with a probability depending on two factors that will be explained below. This heuristic offers good chances of finding configuration with lower internal energy than the initial one. The simulated annealing uses the Metropolis algorithm [24], where a slowly decreasing temperature makes the system to converge into a ground state. SA was used as an alternative training algorithm for topographic map in [16], where it was replaced by deterministic annealing. The work in [16] is not focused on a speed up technique, but to substitute the selection of the winning neurons, solved in SOM in a simple way, with a more sophisticated technique. In the present work, we use SA to speed up the training process of the SOM, preserving its unsupervised characteristic. The adopted approach provides an adaptive learning rate factor $\alpha(t, QE)$, steered by the simulated annealing heuristic over the current resolution of the map.

Using the batch-SOM, at the end of each learning epoch, the Quantization Error (QE) can be identified with the parameter "temperature" T and the evolution of the network can be identified with a perturbation of the system. The QE of a SOM is defined as the expectation euclidean distance between a data vector and its best matching unit according to:

$$QE = E\{\|x - m_c(x)\|\}, \qquad (5)$$

where x is the data vector input and $m_c(x)$ is the *bmu*. The system evolution can be delineated by the QE progression because, if the QE at the end of each learning epoch is smaller than the QE computed in previous epoch, then the projection of the samples on the SOM map is closer to the original positions in the input space.

Thus we obtain a linear cooling schedule as $QE_{new} = QE_{old} - \Delta QE$, where ΔQE is the variation of the total energy of the system.

The pseudocode of the algorithm is given in table I. In this algorithm the term *Training(SOM)* refers to a learning epoch of the batch training process; the result of the training is a candidate SOM that is tested using QE. At the beginning all parameters, including the range of the learning rate, are initialized and the first epoch of the batch-SOM algorithm is executed (steps 1,2). At the end of each learning epoch the QE is calculated and if the difference between the QE calculated at the end of current epoch and the QE calculated at the end of previous epoch is under a threshold δ then the learning process stops (steps 5,5.d). Each learning epoch generates a perturbation of the status of neurons in the map. If this perturbation satisfies low-energy criteria

according to the simulated annealing (step 5.f), then the current configuration is accepted and a new perturbation is calculated. Otherwise, if the perturbation does not satisfy low-energy criteria, then the first perturbation will be used for the next epoch (step 5.g.i.A). If the previous configuration is better than the current one, then the previous one is restored (step 5.g.ii.A). The size of perturbations depends on the evolution of network resolution or, in other words, by the ratio of the current QE and the maximum QE (step 5.f.2). The values of the learning rate factor are adapted according to this ration (steps 5.f.iii, 5.f.iv).

TABLE I
ALGORITHM

1) Initialize the $SOM_{current}$ with random weights, the SOM parameters: $\alpha_{MAX}, \alpha_{MIN}$, and the epoch counter $p = 1$
2) Start with first learning epoch: $SOM(p) = Training(SOM_{current})$
3) Set $QE_{MAX} = QE(p)$
4) Initialize $\Delta QE(p) = QE_{MAX}$
5) While $\Delta QE(p) \geq \delta$
   a) $p = p + 1$
   b) Run a new learning epoch: $SOM(p) = Training(SOM_{current})$
   c) Calculate $QE(p)$
   d) Calculate $\Delta QE(p) = QE(p) - QE(p-1)$
   e) Get a random value $0 < rand < 1$
   f) if the configuration satisfies low-energy criteria (i.e. $e^{-\frac{\Delta QE}{QE}} < rand$), or the configuration is better than the one of previous epoch ($\Delta QE(p-1) > \Delta QE(p)$)
      i) use current state i.e. set $SOM_{current} = SOM(p)$
      ii) Calculate $\alpha_{inc}(QE) = \Delta\alpha * \left|1 - \frac{QE(p)}{QE_{MAX}}\right|$
      iii) Set $\alpha_{MAX} = \alpha_{MAX} + \alpha_{inc}(QE)$
      iv) Set $\alpha_{MIN} = \alpha_{MIN} + \alpha_{inc}(QE)$
   g) Else
      i) if ($e^{-\frac{\Delta QE}{QE}} > rand$))*configuration does not satisfy low-energy criteria*
         A) Use the initial vales of α_{MAX} and α_{MIN} in eq. 4
      ii) if ($\Delta QE(p-1) < \Delta QE(p)$) *previous configuration is better than the current configuration*
         A) Set $SOM_{current} = SOM(p-1)$
6) End of learning after *p* epochs

V. EXPERIMENTAL RESULTS

The proposed fast training algorithm is evaluated against the original procedure in order to appreciate the quality of the approximation and the "smoothness" of the lattice. Moreover the evaluation of the whole framework is carried out using a molecular compounds dataset.

A. Quality Criteria

Two evaluation criteria are used to measure the quality of the map resulting by the SA training algorithm: "quantization error" and regularity degree [25].

The former measures the resolution of the map and the latter the local distortion. The regularity degree can be calculated at the end of each learning epoch and is useful for evaluating the topological organization of the lattice during the training. Distortions are easily calculated as the position of neurons with respect to their neighbors. According to this criterion, an ideal value of the regularity degree should be close to zero (true for a quite flat map). In real applications configurations with lower degree of regularity are considered better.

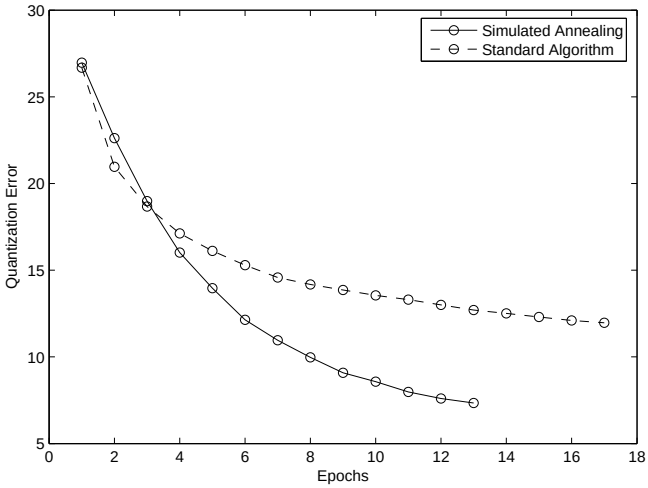


Fig. 2. Quantization error versus number of epochs. The SA algorithm reaches the stop condition at the epoch $p = 13$ in 68.6 s, while the standard algorithm reaches the stop condition at the epoch $p = 17$ in 91.3 s.

B. Evaluation of the proposed Algorithm

The approach is evaluated using two E-SOMs: one with the standard SOM algorithm and one with the adaptive learning rate algorithm. Both maps have a 80×80 square lattice and the training phase for each epoch is done with $\alpha_{MAX} = 0.75$, $\alpha_{MIN} = 0.15$, $\sigma_{MAX} = 7$, $\sigma_{MIN} = 2$, $\delta = 0.15$. In the adaptive algorithm the values of the learning parameters are dynamically increased up to $\alpha_{MAX} = 1$ and $\alpha_{MIN} = 0.75$.

Figure 2 shows the evolution of quantization error during the training process of both maps. The chart clearly shows the effectiveness of the adaptive algorithm.

The adaptive algorithm provides a lower QE value and reaches the stop condition ($\Delta QE < \delta$) in a smaller number of epochs, although for adaptive algorithm pseudo-epochs have also been taken in account, i.e. estimated execution time of adaptive algorithm is about 25% less than standard algorithm.

Figure 3 shows the evolution of the degree of regularity during the training process of both maps. Once again the algorithm based on SA works better than the classic one. The experimental results confirm the good performance of the proposed approach in terms of the degree of regularity and quantization error.

C. Validation of the proposed framework

The validation of the proposed framework has been carried out using a set of real molecular compounds. We have used the publicly available DTP AIDS Antiviral Screen dataset [22] from the National Cancer Institute. The screen measures

TABLE II

COMPOUNDS CLASSIFICATION USED TO TEST THE SOM TRAINING

Chemical Groups						
Azido Pyrimidines (P)	Natural Products or Antibiotics (NA)	Benzodiazepines, Thazolobenzimidazoles and related compounds (BT)	Pyrimidine Nucleosides (Y)	Dyes and Polyanions (DP)	Heavy Metal Compounds (M)	Purine Nucleosides (PN)
10 compounds	8 compounds	10 compounds	11 compounds	13 compounds	3 compounds	2 compounds

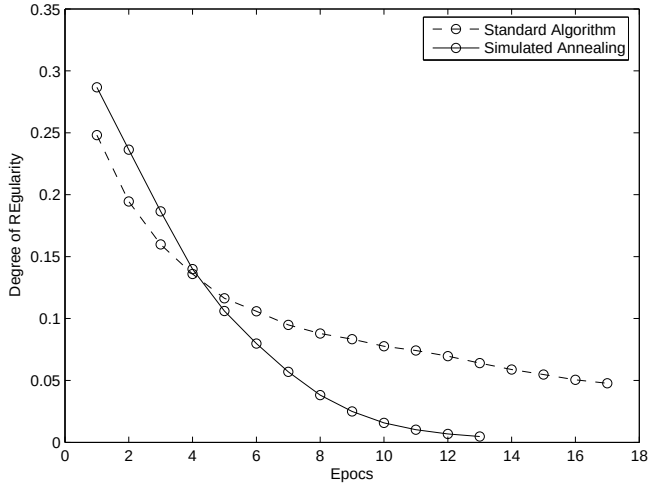


Fig. 3. Degree of Regularity versus number of epochs. The SA algorithm reaches the stop condition at the epoch $p = 13$ while the standard algorithm reaches the stop condition at the epoch $p = 17$.

the protection of human CEM cells from HIV-1 infection [23]. In particular, we used the 325 compounds belonging to the confirmed active class.

The feature vectors are obtained by means of the frequent subgraph mining process (see section III) with minimum support of 10%. The feature generation and selection phases provide 640 molecular fragments. Consequently, each of the 325 compounds is represented by means of a vector of 640 binary components.

In order to obtain a data model that can be better interpreted and exploited, a multidimensional scaling technique (see section III) has been applied. The original vector space has been reduced, using the *Truncated Singular Value Decomposition*, in the vector space defined by means of the 20 most significant singular values over the total 640.

The SOM used for visualization is a 80×80 square lattice, the parameters value are the same of the section V-B.

The result is shown in Figure 4 using the U-Matrix representation [12]. A partial classification of the active compounds in the dataset is available at [22]. The classification identifies 7 chemical groups (table II) for only 57 compounds of the 325 used in the training. In order to verify the correctness of the unsupervised learning process, in the map of figure 4 we have shown in the map the labels of these 57 compounds. Several clusters of compounds with the same labels appear.

In the center of Figure 4(a), the area with the three labels "P" has been highlighted; this area is made by many neurons and represents a cluster of 14 compounds (Figure 4(b)), partially overlapped. Only 7 of the 14 compounds have been assigned to the class of Azido Pyrimidines (label P). The others have no apriori classification, while it is evident from their chemical structure that they do belong to this class. In particular, the compound 602670 is the Zidovudine, a.k.a. the azidothymidine (AZT) group, which is a well known antiretroviral drug, the first one approved for treatment of HIV.

To test consistency of fast learning algorithm, another map has been realized with standard SOM. Both maps show same results.

VI. CONCLUSIONS

We have presented an architecture for the visual exploration of molecular compounds. The overall knowledge discovery process is based on the generation of features, multidimensional scaling and self-organizing maps to generate a visual and interactive representation of the feature space and the similarity among molecules.

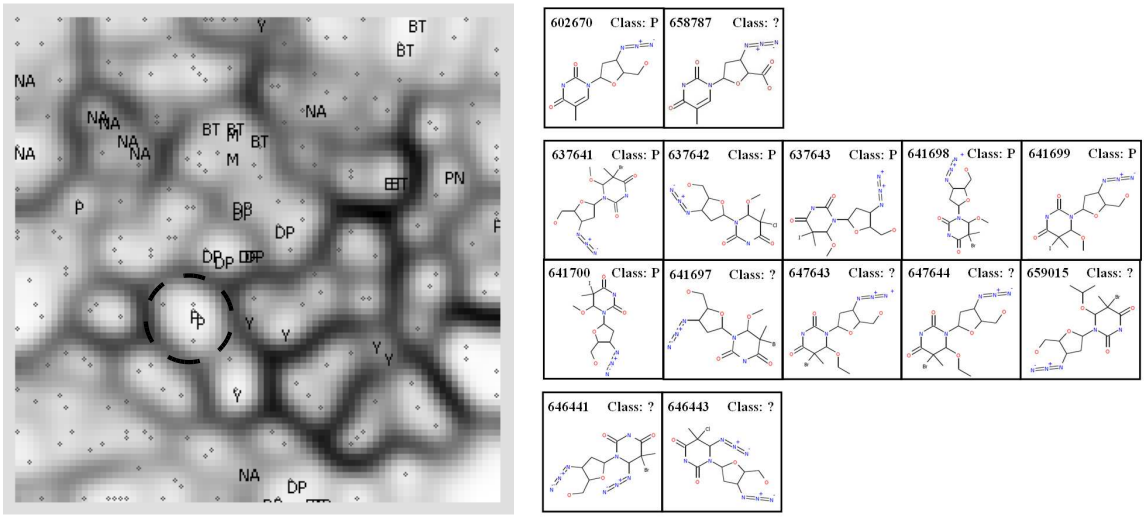
The SOM was trained using a technique that modify the learning rate factor in an adaptive way. This technique uses the simulated annealing as a method to select a candidate SOM during a batch training procedure. This training techniques is faster than the standard batch training. The produced maps were compared to the ones obtained with the standard training by using two evaluation parameters. Further research efforts will focus on the analysis of the topological distortion in the generated maps, on different features selection approaches and on the evaluation of the processing time of the overall data analysis framework.

REFERENCES

[1] M. Deshpande, M. Kuramochi, and G. Karypis, "Frequent sub-structure-based approaches for classifying chemical compounds," *Proceedings of IEEE International Conference on Data Mining (ICDM'03)*, Nov. 19–22, 2003.

[2] G. Di Fatta, A. Fiannaca, R. Rizzo, A. Urso, M. R. Berthold and S. Gaglio, *Context-Aware Visual Exploration of Molecular Databases*, Workshops IEEE International Conference on Data Mining (ICDM 2006), dec 18-22, 2006, pp.136-141.

[3] Gasteiger, J.; Zupan, J., "Neural networks in chemistry," *Angew. Chem. Int. Ed. 32*, pp. 503–527, 1993.



(a) The map with the class labels of 57 compounds. Dashed circle highlights an area where some neurons forming a cluster.

(b) A neighborhood of 14 compounds, 7 of which belong to the group of Azido Pyrimidines (P)

Fig. 4. Representation of 325 HIV-active molecular compounds

[4] Manallack, D.; Livingstone, D., "Neural networks in drug discovery: Have they lived upto their promise?" *Eur. J. Med. Chem. 34*, pp. 195–208, 1999.

[5] Rose, V.; Croall, I.; Macfie, H., "An application of unsupervised neural network methodology kohonen topology-preserving mapping to qsar analysis," *Quant. Struct.-Act. Relat. 10*, pp. 6–15, 1991.

[6] Bienfait, B., "Applications of high reolution self organizing maps to retrosynthetic and qsar analysis," *J. Chem. Inf. Comput. Sci. 34*, pp. 890–898, 1994.

[7] Tetko, I.; Kovalishyn, V.; Livingstone, D., "Volume learning algorithm artificial neural networks for 3d qsar studies," *J. Med. Chem. 44*, pp. 2411–2420, 2001.

[8] Espinosa, G.; Arenas, A.; Giralt, F., "An integrated som fuzzy artmap neural system for the evaluation of toxicity," *J. Chem. Inf. Comput. Sci. 42*, pp. 343–359, 2002.

[9] Tamayo P., Slonim D., Mesirov J., Zhu Q., Kitareewan S., Dmitrovsky E., Lander E. S., Golub T. R., "Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation," *Proceedings of the National Academy Science of USA* **96**, pp. 2907–2912, 1999.

[10] Nikkila J., and Toronen P. and Kaski S. and Venna J. and Castren E. and Wong G., "Analysis and visualization of gene expression data using self-organizing maps," *Neural Networks*, **15**.

[11] K. Haese, "Kalman filter implementation of self-organizing feature maps," Neural Comput., vol. 11, no. 5, pp. 1211-1233, 1999.

[12] Ultsch, A., "Self-organizing neural networks for visualization and classification," *In O. Opitz, B. Lausen, & R. Klar (Eds.), Information and Classification.*

[13] R. Agrawal, T. Imielinski, and A. N. Swami, "Mining association rules between sets of items in large databases," *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pp. 207–216, May 26–28, 1993.

[14] M. Zaki, S. Parthasarathy, M. Ogihara, and W. Li, "New algorithms for fast discovery of association rules," *Proceedings of 3rd Int. Conf. on Knowledge Discovery and Data Mining (KDD'97)*, pp. 283–296, 1997.

[15] C. Borgelt and M. R. Berthold, "Mining molecular fragments: Finding relevant substructures of molecules," *IEEE International Conference on Data Mining (ICDM 2002)*, pp. 51–58, Dec. 9–12, 2002.

[16] T. Graepel, M. Burger and K. Obermayer, "Self-organizing maps: generalizations and new optimization techniques", *Neurocomputing*, vol.21, pp. 173-190, 1998.

[17] J. K. Cullum, R. A. Willoughby, *Real rectangular matrices, in Lancozos algorithms for large symmetric eigenvalue computations, Vol. 1 Theory*. Boston: Brikhauser, 1985.

[18] T. Kohonen, *Self-Organizing Maps*. Berlin: Springer Verlag, 1995.

[19] Kaski, S., Kangas, J., & Kohonen, T., "Bibliography of self-organizing map (som) papers: 1981–1997. neural computing surveys, 1(3&4), 1-176," *available in electronic form at http://www.icsi.berkeley.edu/ jagota/NCS/:* **1** *102-350.*

[20] Ultsch A. and Morchen F., "Esom-maps: tools for clustering, visualization, and classification with emergent som," *Technical Report 46, CS Department, Philipps-University Marburg*, 2005.

[21] T. Heskes. "Energy functions for self-organizing maps", In E. Oja and S. Kaski, editors, *Kohonen Maps*, pages 303316. Elsevier, Amsterdam, 1999.

[22] National Cancer Institute, "DTP AIDS antiviral screen dataset [online]. http://dtp.nci.nih.gov/docs/aids/aids/data.html."

[23] 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," *Journal of the National Cancer Institute, University Press, Oxford, United Kingdom*, vol. 81, pp. 577–586, 1989.

[24] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller, Equation of state calculations by fast computing machines, J. Chem. Phys. 21 (6), 1087-1092, 1953.

[25] J. Goppert, W. Rosenstiel, Regularized SOM-Training: A Solution to the Topology-Approximation dilemma, *ICNN96,*New York, pp. 38-43, 1996.

[26] E. Berglund, J. Sitte, "The parameterless self-organizing map algorithm", IEEE Transactions on neural networks, vol. 17, no. 2, pp. 305-316, 2006.

[27] J.-C. Fort, P. Letrmy, M. Cottrell, "Advantages and drawbacks of the batch Kohonen algorithm", *European Symposium on Artificial Neural Networks* , Bruges, Belgium. pp. 223-230, 2002.

[28] R. Rizzo, A. Chella, "A Comparison between Habituation and Conscience Mechanism in Self-Organizing Maps", IEEE Transactions on neural networks, vol. 17, no. 3, pp. 807-810, 2006.