ISBN 978-83-60810-14-9
ISSN 1896-7094

Neuronal Groups and Interrelations

Michał Pryczek
Institute of Computer Science
Technical University of Lodz
Wolczanska 215, 90-924 Lodz, Poland
Email: michalp@ics.p.lodz.pl

***Abstract*—Recent development of various domains of *Artificial Intelligence* including *Information Retrieval* and *Text/Image Understanding* created demand on new, sophisticated, contextual methods for data analysis. This article formulates *Neuronal Group* and *Extended Neuron Somatic* concepts that can be vastly used in creating such methods. Neural interrelations are described using graphs, construction of which is done in parallel with neural network learning. Prototype technique based on *Growing Neural Gas* is also presented to give more detailed view.**

***Index Terms*—neural networks, pattern recognition, structural pattern recognition, pattern classification.**

I. Introduction

IN GENERAL, modern research on classifiers tends to develop into three non excluding directions. The first one, commonly known as *Kernel Methods* performs kernel based indirect mapping to possibly unknown feature space, known as *kernel trick*. The most notable representatives of this group are *Support Vector Machines* [1]. The second direction makes use of *classifier committee* concept, giving *AdaBoost* and others [2], [3] as well recognized examples.

These techniques can with some success be applied also to structural pattern recognition task next to dedicated solutions. Commonly considered tasks of this family are text and image segmentation. The one plays vital role in *Information Retrieval* field, the other in dynamically developing *Image Understanding* [4] paradigm. New tasks in these fields, like *Opinion Mining* [5] and *Image Symbolisation* [4] prove the amount of work that is still to be done, yielding new models, concepts and techniques that would face problem complexity.

Modern probabilistic models of structural pattern recognition, namely family of methods referred to as *Conditional Random Fields* [6] prove vitality of contextual approach to these problems [7]. Advances made in this field in recent years encourage exploration of possible expert knowledge and context utilisation. What is important, this is done outside *vector space* model of object representation, often referred to as *feature space*, that is the most widely used in *Artificial/Computational Intelligence* community.

Section II contains general discussion about data handling different kinds of knowledge utilization while computing object representation. Section III describes conceptual model of *neuronal group* and *graph based description* of *neuronal interrelations*. This kind of knowledge about data is inferred during neural network learning, building a context that can be used during the rest of learning phase as well as can form part of the solution to the addressed problem.

A prototype method taking benefits from *Neuronal Group Learning* concept is defined in section IV using *Adaptive Active Hypercontour* [8] frame. Other possibilities and promising learning process duality considerations are presented in section V. Supervised classification task will be referred to as an example field of applications.

II. Extended Neuron Somatic Description

Analysing most of contemporary *pattern analysis* methods one can come to the conclusion, that in almost all cases input space definition plays crucial role in entire process.

A. Object space transformation

Construction of *input space* $\mathcal{X}$ can be usually described as object transformation from domain space $\mathcal{O}$ to $\mathcal{X}$, here denoted as a function $\phi : \mathcal{O} \to \mathcal{X}$ [9]. It often includes sophisticated, domain dependent feature extraction and selection techniques. It is typically required, that $\mathcal{X}$ fulfils all requirements of the algorithm we wish to use to analyse it. This usually means, that $\mathcal{X}$ is either *inner product*, *normed* or *metric* space. Various types of elements' *similarity* can also be found in literature and are enough for some methods.

Meaningful construction of $\mathcal{X}$ together with target operations on space elements (e.g. metric) are hard tasks, requiring substantial domain knowledge. Their vitality in entire process can additionally be emphasised by the fact, that mistakes done on this phase of research usually cannot be overcome by choosing more powerful algorithms which operate on $\mathcal{X}$, as they cannot supplement the knowledge that has been lost during transformation ϕ (if it was not redundant).

One of the most popular approach to input space definition is so called *vector space model* (VSM) - considering $\mathcal{X} = \mathbb{R}^n$ for some natural n. Its usefulness cannot be underestimated, especially that its properties have been subjected to many research initiatives and successfully applied to many tasks. The most important benefit of *VSM* usage is availability of numerous standard operation definitions (including sophisticated inner products and metrics) that can be directly used on input space elements regardless ϕ actual definition. On the other hand, this model puts extreme importance on feature extraction, selection and informative power of feature set. The other approach considers more complex representation domains, e.g. including graph structures. It aims to preserve more information stored in original object representation in set $\mathcal{O}$. However, more sophisticated and tightened to $\mathcal{X}$ operations are required. In the end, this approach needs similar infield expert knowledge level, which is introduced in $\mathcal{X}$ operation level rather than in ϕ transformation definition. Last years of research on *structural kernels* replacing inner products brought huge progress in this field.

Even though vital to final results, *input space* definition utilises mostly expert/engineer knowledge and experience. It is worth mentioning, that ϕ usually consists of subsequent steps. Basic feature extraction can, and usually is followed by operations like weighting, normalization, feature selection or chained feature extraction. This transformations often utilise additional knowledge coming not only from expert, but also from data. It is usually inferred from some set of objects accompanied by additional information, e.g. considering supervised classification task this role is typically played by training set, or elements of specially prepared tuning set. It is surprising, that extraction and utilisation of knowledge about interrelations of different areas of $\mathcal{X}$ did not find equally big interest among research teams. One of the efforts worth mentioning are contextual textual document processing ideas presented in [10]

B. Extensions In Neuron Somatic Description

Let the input space be real vector space, to comply with classic *Hebb*'s model of neuron [11]. Somatic parameters of neuron are also required to be real numbers, with additional parameter—*bias weight*. However, other phenomena can have impact on neuron excitement (see *lateral feedback* example in section III). The biggest potential lies in sensor neurons, which operate directly on input data. Currently, somatic information (parameters) stored by neuron are the one required to compute its activity after input presentation. Reminding *Hebb*'s model these are weights (one value for each input including *bias*). There are also other factors that directly or not can influence neuron's behaviour, like position in neural cortex (absolute or relative to neighbouring neurons), activation function parameters not related to input etc. Neuron potential used in *Kohonen* networks to deal with dead neuron problem shows commonness of this approach, which capabilities seems to be underestimated by the researchers.

Only some of these additional parameters can properly be considered *somatic* in biological meaning of this word. Nevertheless, it is proposed to extend somatic description of neuron by data that not necessarily, as input weights or even mentioned *potential*, have direct influence on excitement computation. In contrast, they may play important role during network model adaptation or final result formulation, e.g. by defining context of computation. See sections III and V for more details.

III. Neuronal Groups

Let neural network be seen as a set of neurons V, their parameters and some kind of its architecture description A, that defines *synaptic* neuronal interconnections. These connections meet reflection in somatic description of adjacent neurons. In the concept presented it is proposed, that every set $G \subseteq V$ can form a *neuronal group*. Internal relations between group members and with other groups can be used in both network output computation and its learning.

A. Neurophysiological Background: Neuronal Group Selection Theory (NGST)

Rapid development of neurophysiology in 20^{th} century brought numerous theories about how brain is organised and cognitive process proceed. One of the most interesting and revolutionary ones is called *Neural Darwinism* [12]. Even though concepts presented do not refer directly to evolutionary process proposed by *Edelman*, it may be considered a neurophysiological and, more notably, *philosophical* analogy to the presented work.

The *Neuronal Group Selection Theory* states [12], that brain is dynamically organised into (sub)networks, structure and function of which arises from its development and behaviour. These networks, consisting of thousands of strongly interconnected neurons, are called *neuronal groups* and considered as *functional units*. *Edelman* considers three phases/processes during neural map development. The first one, aimed to build *primary repertoire* of *neuronal groups*, is done within neurobiological ant anatomical constraints dictated by genotype. *Primary variability* of neurons and their synaptic connections form a basis to further process of self-organisation. This phase takes place mostly during foetal life and infancy, when neural system is mainly exposed to self-generated activity and consequently self-afferent information. Subsequent experiment based selection catalyses development of best performing *group prototypes*, leading to formation of primary *functional units* (*primary neural repertoire*). Even though genetically and anatomically determined, adaptation and selection process is itself epigenetic and it is extremely improbable for two organisms to have identical neural structure- this is called *primary variability*.

Development of neural system unavoidably leads to increase of quantity and variability in information processed by it. During postnatal life an epigenetic process of synaptic weight adaptation takes place, leading to composition of *primary groups* into more complex structures resulting in forming *secondary neural repertoire* in a subsequent selective process. This phase is even more ontogenetic, as it bases on individual experiences which create different predicates for *neuronal group selection*. Temporal correlations between various groups leads to creation of dynamically emerging neural maps, whose re-entrant interconnections maintain spatio-temporal continuity in response to re-entrant signals. A resulting neural maps can be considered an effective functional units of brain.

Seeking for a more detailed and intuitive example, please refer to *Mijna Hadders-Algra*'s article [13], which presents human motor development in light of *NGST*.

The idea presented in this article is convergent with *Neuronal Group* concept presented by *Edelman* and thus shares its name. However, at least two important differences must be outlined. The first one is that in the concept presented *neuronal groups* do not have to reflect degree of interconnection between neurons from the group; various approach to group distinguishing was presented in subsection III-C. The second difference is related to network and group development. Evolutionary process analogous to *NGST* might be considered as one of numerous possible algorithms and does not exhaust concept perspectives.

B. Lateral Feedback

An other inspiration for presented model was *lateral feedback* mechanism and facts about formation of cortical map [11]. *Lateral feedback* refers to observed impact which activity of latent neurons have on other neuron excitement and plays important role in topographic organisation of cerebral cortex. Depending on selected interaction model we generally can have positive and negative influence (increasing and decreasing neuron activation), which usually reflects distance between neurons. Dynamical properties lead to formation of bubbles-groups of neurons that tend to recognise similar signals.

This very simple mechanism, although impractical from computational point of view because of dynamical stability problems, focuses on cooperation of neurons and group (bubble) activity as potential research field. This biological motivation can be presented as inspiration of widely used self-organising neural networks called *Self Organising Feature Maps*, *Kohonen networks* [11] and similar derivative models. However, the approach presented addresses different aspects of neural interactions, possibly receding physiological inspirations.

C. Neuronal Group concept

Let $\mathbb{S}$ denotes a set of *neuronal groups*, which can also be considered as a set of group labels. In the concept presented all neurons $n \in V$ have associated set of groups to which they belong, here denoted as function $l : V \rightarrow 2^{\mathbb{S}}$, also referred to as *labelling of neurons*. It can be considered *somatic description extension* (see section II-B) but this is not crucial to the expression of the idea. Limitations on l's codomain can be imposed to simplify model or tune it to specific needs. For instance, condition

$$\forall_{n \in V} \overline{\overline{l(n)}} = 1 \qquad (1)$$

enforces mutual exclusions of groups and can also be described functionally as $\hat{l} : V \rightarrow \mathbb{S}$. In practice, it is desired to introduce or infer groups of different kinds, playing different roles in algorithms that can utilise information coming from grouping of neurons. As it will be shown later, mutual exclusion condition is important tool, but to avoid tainting the concept it should be imposed only on subsets of $\mathbb{S}$, in cases when these group distinguishing policies exclude themselves.

In general we can consider three possible approaches to group identification:

1) predefined groups—identified by an expert/engineer contexts in which interrelations of neurons should be considered.
2) supervised group identification—in which an external premise/knowledge to distinguish groups or perform neuron grouping; this is typical situation in supervised classification tasks, where category labels can act as group descriptors.
3) unsupervised group identification—possibly the most interesting approach; similarly to object grouping tasks, the number of groups can be predefined or not.

Predefined groups might be intentionally compared to *primary neuronal repertoire* included in *NGST*, whereas the other two corresponds to *secondary repertoire*. *Unsupervised group identification* is in general correspondence with *NGST* in approach and philosophy, as premises to groups selection are formed by *internal* relations between neurons and inputs (experiences) related phenomena. However, logical weakness of such analogies should be emphasised.

D. Neuronal interrelations and their descriptions

The most natural way to describe relations between neurons is using graphs of different kind. Obviously, entire network architecture A can be described as weighted, directed graph. Another example of the often analysed relation type would be topological neighbourhood, utilised by self-organising neural networks during learning process, naming *Growing Neural Gas* algorithm (*GNG*) as most straightforward implementation of this idea.

Going further, let us consider set $\mathcal{G}$ as set of available graph interrelation descriptions. Graphs $G \in \mathcal{G}$ can differ on type. They can be either directed or not, allow multiple edges and loops, be weighted ore edge-labelled graphs etc. All depending on characteristics of described relations and how knowledge carried by such description is later used. Similarly to three approaches to groups identification also graph descriptions can be defined according to analogous premises. In this cases however, graphs of predefined interpretation or the one distinguished through supervised process seem to be more natural.

It is worth mentioning that presence of a bijective mapping between neurons and nodes from all graphs is meant. Thus sometimes we will refer to neurons as nodes, or even use set V having set of nodes in mind. However, this is not a requirement.

E. Dualism of inference process

During a network inference phase three processes are taking place in parallel. The first consists of neurons somatic parameters adaptation, namely reference points change or weight adaptation, network reorganisation, etc. The second process is an inference of neurons labelling, which covers label set and neuron membership management. The third, possibly the most challenging one, is inference of interrelation descriptions. This itself can be considered some kind of structural pattern analysis problem.

Although using the word "duality" in the following context may be seen as misuse, it intuitively describes character of relations between the inference of neural network and other two mentioned parallel processes. What is more, neuronal groups and interrelation graphs can be integral part of final result, not only the neural network structure and standard somatic parameters themselves.

IV. Labelled Growing Neural Gas

To keep the presentation clear an example usage of *neuronal group* concepts will now be presented. Consequently, supervised single-label classification of $\mathcal{X} = \mathbb{R}^n$ elements will be analysed. To avoid misunderstanding, this problem can be defined as finding classifying function $h : \mathcal{X} \to \mathcal{L}$, where $\mathcal{L}$ denotes set of labels. In general h of the form $h : \mathcal{X} \to 2^{\mathcal{L}}$ could be discussed (multi-label classification), but here we focus on simpler model for clarity and compactness reasons.

A. Classifier model

Let $G = (V, E)$ be a strict (simple) graph, where V and E denote the node and the edge set respectively, and

$$\forall_{e \in E} \exists_{v_1, v_2 \in V} \; e = \{v_1, v_2\}$$

As it was suggested in section III we will consider V as neuron set. Let each neuron $n \in V$ has associated element $w(n) \in \mathcal{X}$ here called node's *position* or *reference element*. Each $w(n)$ describes some or all somatic parameters (here weight vector) of a neuron associated with n.

Let $\tau : \mathcal{X} \to \mathbb{R}$ be a transmission function (composition of an *activity* and an *output* (activation) functions) of every neuron from V. We will restrict to τ being of the form $\tau = \sigma \circ \rho$, where $\rho : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is linear metric in $\mathcal{X}$, σ is non-increasing function. These imply

$$\forall_{c \in V} \; \tau_n(\mathbf{x}) = \sigma(\rho(w(n), \mathbf{x})) \tag{2}$$

Metric can easily be exchanged with inner product without significant influence on other parts of presentation. In such case σ should be non-decreasing function, as interpretation of inner product result is opposite to distance.

In addition, let $\mathbb{S} = \mathcal{L}$ and mutual group exclusion restriction (1) be imposed. To simplify notation $l(n)$ will denote the label of the group to which n belongs (instead of a set having exactly one element- adequate group label).

Classifying function generated by such network can be defined in the following way:

$$\forall_{x \in \mathcal{X}} \; c_V(x) = l\left(\operatorname*{arg\,max}_{n \in V} \tau(x)\right) \tag{3}$$

It is worth explicit mentioning, that this model is equivalent to Nearest Neighbour algorithm, if one neuron would be constructed for every element x in training set, with reference point and labelling set accordingly.

Graph G will store topological neighbourhood interrelation information in context of input distribution of labelled points or training set. This will be the only description used in the prototype presented.

B. Supervised Neuron Labelling

In the presented solution supervised neuron labelling is used. Policy must reflect the fact that it is not only used in learning phase (see section IV-C) but also for construction of final functional classifier hypothesis. The most straightforward and also effective supervised node labelling strategy is called *precision labelling*. It basically chooses the label that maximises hypothesis *precision* inside its *Voronoi* cell. Neurons *precision* is evaluated using training set or its subset/detached part. Obviously, *recall* and especially *F*-measures can be utilised as well.

Activity labelling can be seen as an alternative to *precision labelling*. While computing new label of neuron it takes also activity of neuron inside its *Voronoi* cell, choosing the *most active* group.

Labelling policy selection should reflect a general aim set to learning process. Referring to *Adaptive Active Hypercontour* techniques of classifier inference, an *energy* function can be used to estimate classifier quality during its inference. Let additionally c_i be the classifier hypothesis in i-th classifier step, regardless step division meaning. The very simple energy function might be any measure of $h(c_i)$ *imprecision*. Minimising energy over training set is equivalent to maximising its precision. However, its true power depends on informative relation between training and test set. Other policies are possible and interesting, depending on entire process definition and its aims. The most promising one would be *contextual labelling*, taking advantage of hypothesis about labels of other nodes. Research about such technique is in progress.

Neuron labelling plays important role in adaptation phase, implying neurons reference points strategies. In this case, effective relabelling strategy is vital. Below the most notable approaches to this problem are listed.

1) Constant labelling—we assume that neuron's labelling is not altered during neural network learning. In this case overall algorithm efficiency depends on ability to generate new neurons belonging to a specific group. What is more, mislabelled node can be erased from V. In certain conditions, constant network with set labelling can also be used.
2) Off-line relabelling—in this case nodes' labellings do not follow dynamical changes in network structure and parametrisation. Relabelling is done using selected policy once for cone time, e.g. after each epoch (processing of entire training set).
3) On-line relabelling—where labelling policy is applied after every change in current process description.

Although on-line relabelling seems the most powerful and meaningful, it is also the most computationally expensive. Trade-off between effectiveness and efficiency of inference process should be analysed while planning to follow this way.

C. Adaptation process

As mentioned earlier, the current labelling hypothesis can be used to drive network adaptation. A classic approach did not use such kind of information. Even considering *RBF network* setting, in which a sensor layer's aim is to decrease the space quantisation error rather than to find optimal quantisation for the second layer (learned in supervised process). One of problems addressed by this prototype implementation is to drive network inference in such a way, that would increase quantisation accuracy along surfaces dividing classes (areas surrounding contours).

Let the concept of *Adaptive Active Hypercontour* be presented in short, to build a frame for further considerations

Let $\mathcal{H}^{\mathcal{X},\mathcal{L}} = \{h : \mathcal{X} \to 2^{\mathcal{L}}\}$ denotes the set of (classifier) hypothesis over $\mathcal{X}$ and $\mathcal{L}$; it will be referred to as $\mathcal{H}$. As (in most cases) function realized by the classifier is considered *a solution*, a distinction is made between classifier $c \in C^{\mathcal{H}}$ referred as *hypothesis* and the classifying function $h(c) : \mathcal{X} \to 2^{\mathcal{L}}$ realized by c, which will be referred as *function hypothesis*.

Let κ^p be parametrised by p a task oriented classifier induction function, $C^{\mathcal{H}}_{\kappa}$ be classifier space (the domain of κ), such that $\forall_{c \in C^{\mathcal{H}}_{\kappa}} h(c) \in \mathcal{H}$. *AAH* solves classifier inference through optimisation process, which target function E is commonly called *energy*. E can be considered as function $\mathcal{H} \to \mathbb{R}$, but it is usually desired that it operates on $C^{\mathcal{H}}_{\kappa}$ directly, as various aspects of classifier parameters should be evaluated to achieve valuable solution.

Adaptive Active Hypercontour algorithm consists of the following steps [14]:

1) **Initialization**: during this phase the choice of initial classifier model and subordinate inference method (κ) is made. An initial classifier hypothesis $c_o \in C^{\mathcal{H}}_{\kappa}$ is then generated and its *energy* computed.
2) $\boldsymbol{\alpha - phase}$: This step mainly consists of subordinate algorithm invocation which generates new classifier hypothesis:
$$c_{i+1} = \kappa_p(c_i)$$
3) c_{i+1} ***energy* estimation**
4) $\boldsymbol{\beta - phase}$: The subordinate algorithm (κ) is very often optimisation process itself. However, its target function usually is compatible, but different than E. In this phase **any** change in current process may be performed to ameliorate E optimisation state modification, like:
   - unwarranted current hypothesis change, usually κ and E oriented, like restoring the previous classifier (in case of significant optimisation regression, or its structural change that bypasses κ shortcomings).
   - κ reparametrisation or replacement with different algorithm.
   - classifier model change.

The solution presented will now be described as an instantiation of presented meta-algorithm. Initialisation of the process consists of creating initial neurons with random reference points. Initial neuron labelling must also be performed; neuron interrelation graph has empty edge set E.

D. Phase $\boldsymbol{\alpha}$

During α phase all inputs from training set or λ random points generated according to training distribution are processed. On-line adaptation using all inputs is considered as one, atomic κ invocation. For each input $\mathbf{x}$ having associated training label denoted as $l_{\mathbf{x}}$ the following steps are performed:

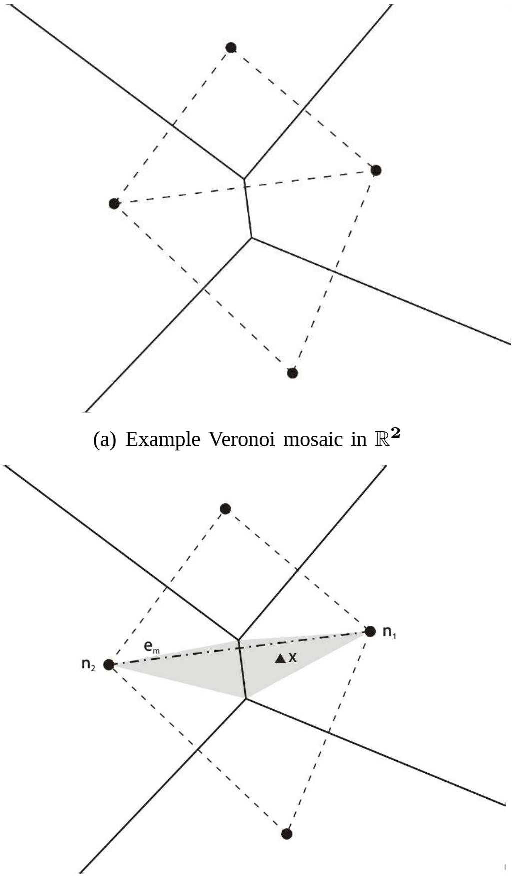
1) Find most active and second-most active neuron from V for given $\mathbf{x}$, let them be n_1 and n_2 respectively.
2) Increase the age of all edges emanating from n_1 in G by 1.
3) If n_1 and n_2 are connected with an edge e_m, set its age to zero. Otherwise, create edge $e_m = \{n_1, n_2\}$.
4) Apply neuronal position adaptation strategy. Few situations can be considered, presented here form "a straightforward example"
   - $\boldsymbol{l(n_1) \neq l(n_2)}$. In this case e_m crosses the surface separating differently labelled areas of $\mathcal{X}$ according to the present function hypothesis. If one of the labels matches $l_{\mathbf{x}}$, then we perform a single-side surface approximation by adapting reference points of matching neuron using standard Kohonen rule:
   $$\Delta w_{n_1} = \alpha_1(\mathbf{x} - w(n_1)) \tag{4}$$
   and its adjacent nodes in G using same rule with different, smaller step:
   $$\Delta w_a = \alpha_a(\mathbf{x} - w(n_1)) \tag{5}$$
   If neither $l(n_1)$ nor $l(n_2)$ are equal to $l_{\mathbf{x}}$, then we perform adaptation of n_1 (together with its neighbourhood).
   - $\boldsymbol{l(n_1) = l(n_2)}$. In this case, $\mathbf{x}$ lays in "internal" region of label $l(n_1)$. If $l(n_1) = l_{\mathbf{x}}$, then no adaptation, or adaptation with negative steps should be performed. Negative learning is optional and, if enabled, should be used only when all neighbours of n_1 and n_2 have the same label equal to $l_{\mathbf{x}}$. If $l(n_1) \neq l_{\mathbf{x}}$ then standard adaptation of n_1 is applied.
5) Remove edges from E with an age larger than a_max. If this results in neurons having no adjacencies—remove them from V.

To illustrate a typical situation during adaptation figures 1(a) and 1(b) are supplied, depicting a typical *Voronoi* cells structure and the active area of one of the edges.

E. Phase $\boldsymbol{\beta}$

Once every phase α, or rarely if desired, additional steps should be taken, e.g. to add new neurons to V. To enforce generation of new neurons in areas of our interest, namely along class discriminating surfaces, two following strategies are proposed.

Intermediate node generation is similar to the node generation mechanism used in *GNG* method. It can be done according to the subsequent steps:



(a) Example Veronoi mosaic in $\mathbb{R}^2$

(b) Example edge activity area

- Determine node p of lowest precision in training set or randomly generated representative set of labelled inputs (if distribution is analysed).
- Insert new node q with $w(q) = \frac{1}{2}(w(p) + w(r))$ where r is neighbour of p in G having lowest precision.
- Insert edges $\{p, q\}$, $\{q, r\}$ into E and remove edge $\{p, r\}$ from it.
- Relabel all neurons from V

This strategy requires global node relabelling. We can avoid it by using different policy of similar power called **node replication**. It simply replicates neuron of lowest precision, thus not altering Voronoi cell distribution.

F. Summary and possible extensions

Presented *Labelled GNG* algorithm uses few concepts presented in previous section. At first, it uses supervised neuron labelling to infer classifying function of simple single-layer neural network. Although such network can be used as classifier (with formulae 3), its output can also be processed by other chained classifier model. In this case we will benefit from increased quantisation rate along decision surfaces.

The second important extension is *Neuronal Group Learning*, which utilises information about inferred *neuronal groups* during network adaptation process. This is used in both α and β phases of the prototype method presented.

Empirical results and their analysis will be presented during the conference.

V. NEURONAL INTERRELATIONS AS INFERRED KNOWLEDGE

The previous section presented method of *Labelled Growing Neural Gas* inference that can later be used as classifier. However, parallel inference of *neuronal groups* and *neuronal interrelation descriptions* can be the true problem solved by this method. At first, some other concept presented in [14] will be considered.

A. Neuronal Group Activity

As suggested at the beginning, a formation of neuronal group does not have to be used only to facilitate learning process. Neurons can cooperate within one group and even compete with neurons from other groups during both learning and use stages.

This was first used in [14] in the concept of competitive learning of neural networks. Although differently described in original paper, let the concept of *neuronal group activity* be here introduced.

Let $\mathbb{F}$ be a common codomain of all neurons' transmission function, later referred as *field* or *output space*. In addition, let $\mathcal{L} \subset \mathbb{S}$, and $\forall(s \in \mathbb{S})V(s)$ denote all neurons from group s. Function hypothesis generated by neural networks with *neuronal group* information can be defined using the following formulae:

$$c(\mathbf{x}) = \arg\max_{l \in \mathcal{L}} \ \Psi_l(\mathbf{x}) \tag{6}$$

where Ψ_l is activity of group l. It was proposed in [14] that Φ_l was of the form

$$\Psi_l(\mathbf{x}) = norm^{|V(l)|} \ (\Phi_{V(l)}(\mathbf{x})) \tag{7}$$

where $\Phi_{V(l)} : \mathcal{X} \rightarrow \mathbb{F}^{|V(l)|}$ be group activation function returning a vector of activations of neurons from specified group and $norm^k : \mathbb{F}^k \rightarrow \mathbb{R}$ be a proper norm (or semi-norm) in $\mathbb{F}^k$ space.

Interpreting set $\{(v, l(v)) : v \in V\}$ as a set of labelled control points, an appropriate definition of τ and $norm$ will result in classifier model equivalent to the one introduced be *potential method* and used in *Adaptive Potential Active Hypercontour* technique. Its properties allows efficient *energy* driven network inference, a solution to some optimisation problems faced during work presented in [9].

B. Structural Knowledge Acquisition

One of the most promising fields of application of presented concepts are those connected with structure discovery and relation recognition, in which neural network is a backbone of *structural pattern analysis* process. Node distribution in space $\mathcal{X}$ induce some Voronoi cell structure. In described parallel processes relations among these cells can be recognised and inscribed in graphs of *neuronal interrelations*. Some of used graphs may have special meaning and can even form solution of *structural pattern analysis* problem.

To present an example, perspectives in the field of my special interest, data segmentation, will be analysed. Considering two dimensional segmentation problem (as image segmentation is) let neuron reference points lie in space $N \times N$ (as pixel coordinates do). It is quite natural, that neuron distribution in pixel coordinates space together with neuronal groups and labellings reflecting output labels forms a solution. However, additional somatic parameters and/or interrelation information may be needed during network learning.

Willing to provide argument proving potential of the concept presented it this field a straightforward image segmentation or contour inference algorithm can be defined in two simple ways.

The first uses concept of group activity and possibility to define them so as to get model equivalent *potential method*. In conjunction with meaningful definition of *energy* function, a *Potential Active Contours* are obtained. The other, better suited in neural network backbone of the method presented is known as *Kohonen Snakes*.

VI. Conclusions and further work

The article presents *Neuronal Group* concept accompanied with based on graphs *neuron interrelation description* as a method of enriching neural network inference models. If properly defined *Neuronal Group Learning* can also lead to solution of *structural pattern analysis* tasks. Group descriptions and interrelations between neurons can explicitly be defined by a researcher. However, the widest perspectives are given by automatic, supervised or unsupervised interrelation inference.

Apart from extending existing methods, the one presented in this article can be used in advanced *structural pattern analysis* problems. Recent development and new tasks of *Information Retrieval* and *Text/Image Understanding* fields created a huge demand on contextual methods that would be flexible in utilisation of any kind of human knowledge about problem domain. What is more, automated knowledge inference from data is of increasing importance in all *Artificial Intelligence* domains. Future work on presented concepts will focus on these demanding task, especially that they proves to be much more difficult and requiring more sophisticated computational and analysis models than many currently considered.

Acknowledgments

The author is grateful to Professor Piotr S. Szczepaniak from the Technical University of Lodz, Poland, for fruitful discussions.

References

[1] C. J. C. Burges, “A tutorial on support vector machines for pattern recognition,” *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[2] Y. Freund and R. E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” in *Computational Learning Theory, Second European Conference, EuroCOLT ’95, Barcelona, Spain, March 13-15, 1995, Proceedings*, ser. LNCS, vol. 904. Springer, 1995, pp. 23–37.

[3] A. Esuli, T. Fagni, and F. Sebastiani, “MP-Boost: A multiple-pivot boosting algorithm and its application to text categorization,” in *Proceedings of the 13th International Symposium on String Processing and Information Retrieval (SPIRE’06), Glasgow, UK*, ser. LNCS, vol. 4209. Berlin / Heidelberg: Springer, 2006, pp. 1–12. [Online]. Available: http://www.isti.cnr.it/People/F.Sebastiani/Publications/SPIRE06a.pdf

[4] A. Tomczyk and P. S. Szczepaniak, “Contribution of active contour approach to image understanding,” in *Proceedings of IEEE International Workshop on Imaging Systems and Techniques—IST 2007, May 4–5, 2007, Krakow, Poland*, 2007.

[5] J. Wiebe, T. Wilson, and C. Cardie, “Annotating expressions of opinions and emotions in language,” *Language Resources and Evaluation*, vol. 39, no. 2, pp. 165–210, 2005.

[6] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, “Conditional random fields: Probabilistic models for segmenting and labeling sequence data.” in *Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, 28 June – 1 July, 2001*, C. E. Brodley and A. P. Danyluk, Eds. Morgan Kaufmann, 2001, pp. 282–289.

[7] E. Breck, Y. Choi, and C. Cardie, “Identifying expressions of opinion in context.” in *Proceedings of the 20th International Joint Conference on Artificial Intelligence, 6-12 January 2007, Hyderabad, India,*, M. M. Veloso, Ed., AAAI Press, Menlo Park, California, 2007, pp. 2683–2688.

[8] A. Tomczyk and P. S. Szczepaniak, “Adaptive potential active hypercontours,” in *Proceedings of 8th International Conference on Artificial Intelligence and Soft Computing (ICAISC). Zakopane. Poland*. Berlin. Heidelberg: Springer-Verlag, 2006, pp. 692–701.

[9] P. S. Szczepaniak, A. Tomczyk, and M. Pryczek, “Supervised web document classification using discrete transforms, active hypercontours and expert knowledge,” in *Proceedings of Web Intelligence Meets Brain Informatics, First WICI International Workshop, WImBI 2006, Beijing, China, December 15-16, 2006, Revised Selected and Invited Papers*, ser. LNCS, N. Zhong, J. Liu, Y. Yao, J.-L. Wu, S. Lu, and K. Li, Eds., vol. 4845. Springer, 2006, pp. 305–323.

[10] K. Ciesielski, “Adaptacyjne metody grupowania w mapowej wizualizacji kolekcji dokumentów tekstowych (eng: Adaptive grouping methods in map visualisation of document collection),” Ph.D. dissertation, Institute of Computer Science, Polish Academy od Sciences, Warszawa, 2007, (in polish).

[11] S. Haykin, *Neural Networks: A Comprechensive Foundation*. New York: Macmillan College Publishing Company, 1994.

[12] G. M. Edelman, *Neural Darwinism: The Theory of Neuronal Group Selection*. New York: Basic Books, 1987.

[13] M. Hadders-Algra, “The neuronal group selection theory: a framework to explain variation in normal motor development,” *Developmental Medicine & Child Neurology*, vol. 42, pp. 566–572, 2000.

[14] M. Pryczek, “Supervised object classification using adaptive active hypercontours with growing neural gas representation,” *Journal of Applied Computer Science*, 2008, (submited).