Engineering Applications of Artificial Intelligence 23 (2010) 346–356

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Data compactification and computing with words

Witold Pedrycz [a,b,*], Stuart H. Rubin [c]

[a] Department of Electrical & Computer Engineering, University of Alberta, Edmonton, Canada AB T6R 2G7
[b] Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland
[c] Space and Naval Warfare Systems Center, San Diego, code 5634, 53560 Hull Street, San Diego, CA 92152-5001, USA

ARTICLE INFO

Article history:
Received 11 February 2009
Received in revised form
4 September 2009
Accepted 9 September 2009
Available online 11 November 2009

Keywords:
Data compactification
Computing with words
Randomization
Grammars
NP-completeness
Information granules
Information granulation

ABSTRACT

The underlying objective of this study is to show how fuzzy sets (and information granules in general) and grammatical inference play an interdependent role in information granularization and knowledge-based problem characterization. The bottom-up organization of the material starts with a concept and selected techniques of data compactification which involves information granulation and gives rise to higher-order constructs (type-2 fuzzy sets). The detailed algorithmic investigations are provided.

In the sequel, we focus on Computing with Words (CW), which in this context is treated as a general paradigm of processing information granules. We elaborate on a role of randomization and offer a detailed example illustrating the essence of the granular constructs along with the grammatical aspects of the processing.

1. Introduction and problem formulation

Assessing quality of available data, especially in situations where they are significantly scattered and of high dimensionality becomes crucial for their further usage in a variety of reasoning schemes. The nature of data and their distribution implies different levels of quality of results of inference.

The data usually come with some redundancy, which is detrimental to most of the processing in which they are involved. It could be also inconvenient to interpret them considering the size of the data set itself. Taking those factors into consideration, it could be of interest to represent the whole data set $\boldsymbol{D}$ by its selected subset of elements $\boldsymbol{F}$, where $\boldsymbol{F} \subset \boldsymbol{D}$. While there is a wealth of approaches that exist today, most of them are concerned with some form of averaging meaning that at the end we come up with the elements, which have never existed in the original data meaning that they usually may not have any straightforward interpretation. In contrast, if $\boldsymbol{F}$ is a subset of $\boldsymbol{D}$, the interpretability does not cause difficulties. It is also evident that the choice of the elements of $\boldsymbol{F}$, as well as their number, implies the quality of representation of original data $\boldsymbol{D}$. This set being treated as a "condensation" of $\boldsymbol{D}$ can be a result of a certain optimization. The cardinality of $\boldsymbol{F}$, which is far lower that the cardinality of $\boldsymbol{D}$ itself helps alleviate the two problems we identified at the very beginning.

Let us start with a formal presentation of the problem, where we also introduce all required notation. We are provided with a collection of data $\boldsymbol{D}=(\mathbf{x}_k, \mathbf{y}_k)$, $k=1, 2, \ldots, N$ forming an experimental evidence coming from a certain process or phenomenon. We assume that $\mathbf{x}_k$ and $\mathbf{y}_k$ are vectors in $\mathbf{R}^n$ and $\mathbf{R}^m$, respectively. The semantics of $\mathbf{x}_k$ and $\mathbf{y}_k$ depends on the setting of the problem (and will be exemplified through several examples); in general we can regard $\mathbf{y}_k$ to be a certain indicator (output) associated with the given $\mathbf{x}_k$.

Graphically, we can portray the crux of the problem in Fig. 1. The essence of the optimization criterion guiding the construction of $\boldsymbol{F}$ is to represent $\boldsymbol{D}$ by the elements of $\boldsymbol{F}$ to the greatest extent; we will elaborate on the details of the objective function later on. Each element of $\boldsymbol{D}$ is expressed via a certain relationship whose "c" arguments ($\mathbf{x}_{i1}$, $\mathbf{x}_{i2}$,…, $\mathbf{x}_{ic}$) are elements of $\boldsymbol{F}$, see also Fig. 1. More specifically, we can describe it concisely as

$$\hat{\mathbf{y}}_k = \boldsymbol{\Phi}(\mathbf{x}_k;\mathbf{x}_{i1},\mathbf{x}_{i2},\ldots,\mathbf{x}_{ic}) \tag{1}$$

where $k \in \boldsymbol{N}-\boldsymbol{I}$ and we strive for the relationship $\hat{\mathbf{y}}_k = \mathbf{y}_k$, which can be achieved through some optimization of the mapping itself as well as by way in which $\boldsymbol{F}$ has been constructed.

As the form of the mapping stipulates, we are concerned with a certain method for data compactification.

In the study, we use some additional notation: let $\boldsymbol{N}$ stand for the set of indexes, $\boldsymbol{N}=\{1, 2, \ldots, N\}$, while $\boldsymbol{I}$ be a subset of "c"

* Corresponding author at: Department of Electrical & Computer Engineering, University of Alberta, Edmonton, Canada AB T6R 2G7
E-mail address: pedrycz@ee.ualberta.ca (W. Pedrycz).

0952-1976/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.engappai.2009.09.012

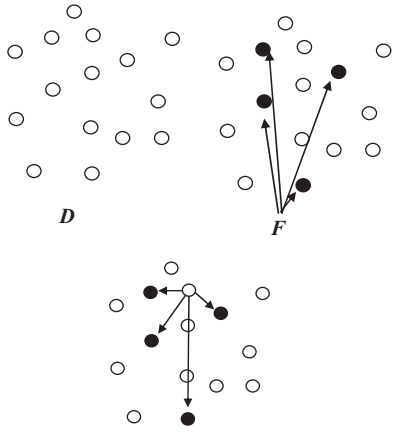

Fig. 1. Compactification of data: from original data ***D*** to its compact representation ***F*** shown is a way in which elements of ***D–F*** are represented by the content of ***F***.

indexes of ***N***, $\boldsymbol{I} \subset \boldsymbol{N}$, $\boldsymbol{I}=\{i_1, i_2, \ldots, i_c\}$ used to denote the elements of ***F***.

The structure of the data as presented above is suitable in a variety of contexts:

- *Decision-making processes*. For instance, in assessing terrorist threats we are provided (on the basis of some previous cases or scenarios), a collection of characterizations of a threat situation ($\mathbf{x}_k$) and the associated actions along with their preference (relevance) $\mathbf{y}_k$, say $\mathbf{y}_k$=[0.8 0.4 0.05] with actions such as "enhance surveillance", "deploy patrol", or "issue warning".
- *Prediction*. Here $\mathbf{x}_k$ is concerned with a vector of variables describing a certain process at a given moment in time, while $\mathbf{y}_k$ is a vector of the same variables with the values assumed in the consecutive moment. The concept can be used in various schemes of learning—including neural networks (Harvey, 1994).
- *Classification*. In this case, $\mathbf{x}_k$ is viewed as a vector of features in the n-dimensional space, while $\mathbf{y}_k$ is a Boolean vector of class allocation; in particular for a two-class problem, $\mathbf{y}_k$ assumes a single Boolean value.

It is worth noting that a well-known scheme for Case-Based Reasoning (CBR) (Duda et al., 2001); (Rubin, 1992) emerges as one of the general alternatives, which takes advantage of the format of the data used here. In general, CBR embraces four major processes: (a) retrieving cases from memory that are relevant to a target problem; (b) mapping the solution from the closest (the most similar) retrieved case to the target problem; (c) possible modification of the solution (its adaptation to the target problem); and (d) retaining the solution as a new case in memory. This study shows that the successive phases of processing can be realized and the reasoning results quantified in terms of information granules.

One of the problems addressed by this paper is not only that of *quantitative* granularization and its attendant mechanics and algorithmic details, but that of *qualitative* granularization and fuzzification (or computing with words as it is more commonly known in the literature). A related problem, addressed herein, has to do with knowledge imbued in specific domains versus techniques for general domains, which may be *NP-hard*. It will be shown that the computer as a device for carrying out massive (and concurrent) searches underpins both and that computing with words can be underpinned by transformational grammars. A specific example relating to the design of a refrigeration device serves to illustrate the point. While 2-level or *w-grammars* (i.e., a pair of CFGs, where one generates the productions used by its companion) are of type-0 generality, the exposition shows that such grammars may transform—not merely write the productions of another grammar in a manner that is similar to the duality between data and program found in common LISP.

The paper is structured in a bottom-up manner. We start with the formulation of the optimization problem (Section 2); here we clearly identify the main phases of the process of optimization by distinguishing between parametric and structural enhancements. The structural aspect of optimization is handled by running one of the techniques of evolutionary optimization, namely Particle Swarm Optimization (PSO). The pertinent discussion is covered in Section 3. Section 4 is concerned with the development of higher-order information granules, which are inherently associated with the essence of the compactification process. We show that, on a conceptual level, the resulting constructs become interval-valued fuzzy sets or type-2 fuzzy sets, in a general setting. Illustrative experiments are reported in Section 6. While those sections are of more detailed nature, in the sequel we build upon these findings and focus on Computing with Words (CW) as a general paradigm of processing information granules. Here we underline the role of randomization as being inherent to the essence of the CW processing. A detailed design example is covered in Section 7.

2. The optimization process

Proceeding with the formulation of the problem, there are two essential design tasks, that is (a) determination of ***F*** and (b) formation of the prediction (estimation) mechanism of the output part associated with $\mathbf{x}_k \in \boldsymbol{F}$. We start in a bottom-up fashion considering (b) and assuming that at this phase the set ***F*** has been already determined.

2.1. Reconstruction and its underlying optimization

In the reconstruction procedure, our intent is to express (predict) the conclusion part associated with $\mathbf{x}_k \in \boldsymbol{F}$ in such a way that this prediction $\mathbf{y}_k$ is made as close as possible to $\mathbf{y}_k$. Intuitively $\mathbf{y}_k$ can be expressed on a basis of what is available to us that is $\mathbf{y}_i \in \boldsymbol{F}$. A general view can be expressed in the form of the following aggregation:

$$\hat{\mathbf{y}}_k = \sum_{i \in \mathbf{I}} \mathrm{u}_i(\mathbf{x}_k)\mathbf{y}_\mathrm{i} \tag{2}$$

where $u_i(\mathbf{x}_k)$ is sought as a level of activation, closeness, proximity, or relevance of $\mathbf{x}_k \in \boldsymbol{D–F}$ and the ith element of ***F***. The closer the two elements are, the higher the value of $u_i(\mathbf{x}_k)$ is. In some sense, $u_i(\mathbf{x}_k)$ can be treated as a receptive field constricted around $\mathbf{x}_i$ capturing the influence $\boldsymbol{x}_i$ has on its neighborhood. The closeness is quantified through some distance and here we may benefit from a variety of ways in which the distance could be expressed. In addition to the commonly encountered distance functions, one can also consider those based on tensor representation of the space, cf. (Dodson and Poston, 1997). The optimization of the receptive field comes from the following formulation of the optimization problem:

$$V = \sum_{i \in \mathbf{I}} \mathrm{u}_i^p(\mathbf{x}_k)\,||\mathbf{x}_k - \mathbf{x}_i||^2$$

$$\text{Min } V \text{ with respect to } \mathbf{x}_i \in \boldsymbol{I} \tag{3}$$

where we assume that $u_i(\mathbf{x}_k) \in [0,1]$ and as usual require that these values sum to 1.