

# 粒度神经网络对数字语言的数据融合<sup>\*</sup>

Data Fusion of Numerical-Linguistic Based on Granular Neural Networks

行小帅<sup>1,2</sup> 焦李成<sup>1</sup>

(西安电子科技大学雷达信号处理国家重点实验室 西安710071)<sup>1</sup>

(山西师范大学物理系 临汾041004)<sup>2</sup>

**Abstract** This paper presents a neural networks based knowledge discovery and data mining (KDDM) methodology based on granular computing, neural computing, fuzzy computing, linguistic computing, and pattern recognition. A granular neural network (GNN) is designed to deal with numerical-linguistic data fusion and granular knowledge discovery in numerical-linguistic databases. The GNN is able to learn internal granular relations between numerical-linguistic inputs and outputs, and predict new relation in a database.

**Keywords** Granular neural networks, Data fusion, Data mining, Knowledge discovery

## 1 引言

知识发现和数据挖掘(Knowledge discovery and data Mining, 简称 KDDM)是近几年来随着人工智能和数据库发展起来的一门新兴的数据库技术。其处理对象是海量的日常业务数据,其目的是从大量的数据源中提取人们感兴趣的、有价值的知识和重要的信息<sup>[1]</sup>。由于计算机和通信技术的迅猛发展,人类活动产生的数据日益增加,大量的各种数据库用于政府事务、科学研究、工业生产、商业管理和其它各个方面。数据的爆炸式增长使 KDDM 成了一个日益重要的研究领域。所提取的知识可用于问题求解、生产控制、信息管理、判断决策及其它许多方面,这将会产生巨大的社会效益和经济效益。

由于神经网络分类精确、鲁棒性好,因此能从数据库中发现非线性的、不明显的和潜在的有用信息和知识<sup>[2]</sup>。在数据库中,对给定一组输入-输出数据,普通清楚的神经网络(Crisp Neural Network, CNN)能训练所给的数据组,然后训练的 CNN 能调节清楚的数字权重,学习输入数据和输出数据之间潜在的知识,最后 CNN 能够应用发现的知识为新的输入数据去预测新的输出数据。对于 KDDM 而言, CNN 是一个有用的工具。在大多数的应用中上述输入和输出数据是清楚的数字。但在实际情况中,不仅存在数字数据如 3.1416-1.732,而且也存在语言(或者符号)数据如比较高、很年轻、近似等于 10 等等,这在政府和商业的各种数据库中都存在。一个语言神经网络(Linguistic Neural Network, LNN)能够处理语言(或者符号)输入和输出数据,如何在 KDDM 领域中用神经网络去处理数据库中的数字和语言数据问题变成了一个挑战性的问题。在数据库中对于单纯数字的 KDDM 而言, CNN 和模糊神经网络(Fuzzy network FNN)是一个强大的工具<sup>[3,4]</sup>,而对于语言 KDDM 而言,用文字计算(Computing with Words, CW)是一般语言计算的方法,能够用于发展语言计算的 KDDM 系统<sup>[5]</sup>。对于一个完成数字和语言的 KDDM,动力学结构的模糊神经网络可用于数字-符号的挖掘,并且可用于模糊知识的压缩、扩大和再发现<sup>[7,8]</sup>。然而,存在的主要问题是:在数

据库中如何训练 CNN 和 FNN 用于数字和语言数据;如何从数字-语言数据库中提取 IF-Then 规则;如何在数据库中发现的知识去检查丢失的数字-语言数据并且预测新的数据。针对上述三个问题,提出了一个新颖的粒状神经网络(Granular neural network, GNN)<sup>[3,6,8]</sup>。一般情况下, GNN 有能力处理粒状数据(如数字和语言数据),提取粒状的 IF-Then 规则,融合粒状数据组,压缩粒状数据库,并且能预测新的数据。本文的主要内容是:1)提出了模式识别基本的方法,这个方法能将数字和语言数据转换为数字特性的关系;2)在数据库中,利用 CGNN 和 FGNN 工具处理数字数据和语言文字;3)利用 CGNN 和 FGNN 解决了粒状数据的融合问题,而且 FGNN 也可用于发现粒状模糊知识。

## 2 粒状特性的提取

一个数据库可以包含各种各样的多媒体数据例如:数值、语言文字、肖像、声音、音乐、画像和课文等。最低的数据粒状技术,是从真正的世界环境中直接处理未加工的多媒体数据的收集;而较高的数据粒状技术可以分类未加工的多媒体数据成为较高水平的粒状(即可分类、聚类、种类、群、组等),可简化数据的处理和数据挖掘。在这样的意义下,数据的粒化可叙述为数据的挖掘、数据聚类、信息融合和知识发现。实际上,人脑有很强的能力去处理多媒体的粒化,计算语言文字并从多媒体数据库中发现有用的信息。在外部真正的世界中未加工的多媒体数据和人脑内部生物神经网络之间存在相互关系。类似地,在数据库中多媒体粒化和人工神经网络的输入和输出之间将建立相互关系。在大多数情况下,一般的神经网络不能直接处理多媒体数据,所以如何转换多媒体数据成为对应的数字特性,这对于 KDDM 特性系统使用中的神经网络至关重要。由于这个问题的复杂性,因此下面仅讨论关于数字和语言文字的粒化,并且提供解决实际问题的方法;研究运用神经网络有效地处理多媒体数据粒状特性的提取方法和模糊语言数据特性的提取方法。

### 2.1 多媒体数据特性的提取

<sup>\*</sup> 本课题得到国家自然科学基金项目(60133010)和863高技术研究发展计划项目的资助。行小帅 副教授,访问学者,主要研究领域包括:人工神经网络、数据挖掘和进化算法等。焦李成 教授,博士生导师,主要研究领域包括非线性理论、人工神经网络、子波理论与应用、进化算法、数据挖掘与多用户检测等。

各种多媒体数据库、数字图书馆和 Web 基本的数据库包含不同的多媒体数据,这需要用神经网络来挖掘有用的信息。但在许多情况下,神经网络使用的是数字数据,而不能直接使用多媒体数据,因此合理地转变多媒体数据是必要的。一般的多媒体数据特性提取系统如图1(a)所示,利用多媒体数据做为系统的输入,多媒体数据特性提取系统利用一些模式识别技术能将多媒体数据转换成对应的数字特性;然后将这些特性提供给一个神经网络。多媒体数据特性系统的输入联接多媒体数据库,其输出联接神经网络的数字输入。然而,它是不同于一般设计的多媒体数据特性提取系统,因为我们必须利用许多不同的模式识别技术去处理不同的多媒体数据。下面仅考虑如何设计一个语言数字特性提取系统。

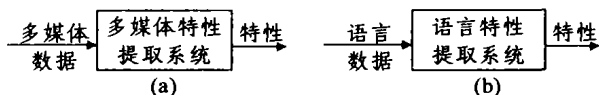


图1 多媒体特性、语言特性提取系统

## 2.2 语言数字特性的提取

基于上述设计多媒体数据特性提取系统的方法,现在考虑一个具体的语言数字特性提取系统如图1(b)所示。一般情况下,语言数字特性系统能够将模糊语言数据(如很高、太老、近似于50)转换成为典型的数字特性。下面给出有关模糊数字特性的四个定义。

**定义1** 一个梯形特性矢量 $(a, b, c, d)$ 是几何表示的梯形模糊集 $\tilde{A}$ 。使用最小、最大算子有更简洁的表述形式,可定义如下:

$$\mu_{\tilde{A}}(x; a, b, c, d) = \max(\min(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}, 0)) \quad (1)$$

参数 $\{a, b, c, d\}$  ( $a < b \leq c < d$ ) 决定了梯形 MF 的四个角的  $x$  坐标值。

**定义2** 高斯特性矢量 $(\alpha, \beta, \delta_-, \delta_+)$ 是几何表示的梯形-高斯模糊集 $\tilde{B}$

$$\mu_{\tilde{B}}(x; \alpha, \beta, \delta_-, \delta_+) = \begin{cases} \exp\left[-\left(\frac{x-\alpha+\beta}{\delta_-}\right)^2\right] & \text{for } (a-\beta) > x \\ 1 & \text{for } (a-\beta) \leq x < (a+\beta) \\ \exp\left[-\left(\frac{x-\alpha-\beta}{\delta_+}\right)^2\right] & \text{for } x > (a+\beta) \end{cases} \quad (2)$$

**定义3** 模糊信息容量的模糊集 $\tilde{A} = \int_A \mu_{\tilde{A}}(x)/x$  在区间 $[x_1, x_2]$ 上定义

$$\rho = \int_{x_1}^{x_2} \mu_{\tilde{A}}(x) dx \quad (3)$$

**定义4** 模糊信息特性矢量 $[a, b, \rho^-, \rho^+]$ 是梯形模式特性表示的模糊集 $\tilde{A}$ ,定义如下:

$$\mu_{\tilde{A}} = \begin{cases} f^+(x) & \text{for } x > (a+b) \\ 1 & \text{for } (a-b) \leq x \leq (a+b) \\ f^-(x) & \text{for } (a-b) > x \end{cases} \quad (4)$$

其中,

$$\rho^- = \int_{-\infty}^{a-b} f^-(x) dx \quad (5)$$

$$\text{和 } \rho^+ = \int_{a+b}^{+\infty} f^+(x) dx \quad (6)$$

式中 $f^-(x)$ 和 $f^+(x)$ 是单调非减和非增函数,分别为 $0 \leq f^-(x) \leq 1$ ,和 $0 \leq f^+(x) \leq 1$ 。

根据上述定义,模糊语言数据特性提取系统能把通常使用的模糊集转换成为数字特性。对于数字-语言的 KDDM 而

言,这些特性既可以用于 CNNs 也可以用于 FNNs。

## 3. 粒度神经网络

通常 GNN 有能力处理各种粒状数据,像数字分类、一像素、一组概念、一个目标、一种范畴的数据分类,这些粒状数据是 GNNs 的输入和输出。因此,在处理多媒体粒状数据方面,与普通的神经网络相比,GNNs 不仅有用,而且有效。

### 3.1 CGNN 的设计

CNNs 和语言数字特性提取系统是 CGNN 基本的组成部分。为了方便,设计的输入-输出 CGNN 典型的结构如图2所示。 $X$  和  $Y$  是混合数字-语言数据的两个数据变量。用4个特性模糊集表示模糊语言数据。左边的语言特性提取系统可将  $X$  转换成为一个特性组 $(a1, b1, c1, d1)$ ,并且右边的语言特性提取系统可将  $Y$  转换成为另一个特性组 $(a2, b2, c2, d2)$ 。注意到数字的值表示为一个清楚的特性 $(a, 0, 0, 0)$ ,因为它是清楚的而不是模糊的。

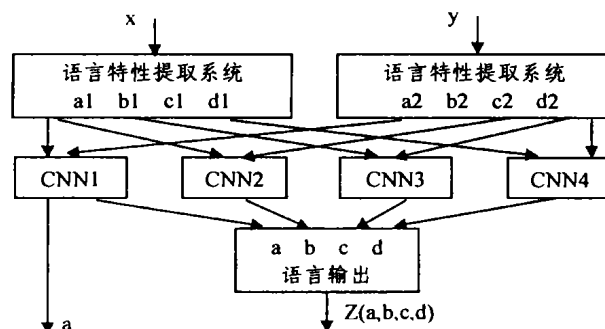


图2 CGNN 一般结构

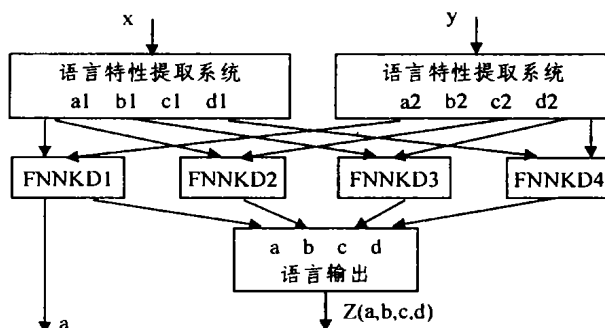


图3 FGNN 一般结构

BPA 用于训练 CNNs 的输入和输出特性集。训练之后,CGNN 能够产生一个数字输出  $a$  和一组模糊语言输出。因此 CGNN 具有数字和语言数据融合的能力。CGNN 可用分布式的数字权重解释信息和知识,但是它存在两个主要的缺点:①学习算法非常慢。由于权重在神经网络中有不明确的物理意义,因此在刚开始时其效率很低。②提取语言知识是很困难的。所以下面设计了一个 FGNN 来处理各种粒状数据,其分类好、鲁棒性强。

### 3.2 FGNN 的设计

FNNKD 是 FGNN 的基本结构模块,FNNKD 详细的五层结构参阅文[9]。FGNN 的结构如图3所示,有2个输入1个输出,由三层组成,不同的层次有不同的作用。现描述如下:

第一层是语言特性提取层:这一层可将数字语言  $X$  和  $Y$  转换成相应的特性矢量 $(a1, b1, c1, d1)$ 和 $(a2, b2, c2, d2)$ 。

第二层是多层 FNNKD 层:这一层由四个 FNNKDs 组成,即: FNNKD1, FNNKD2, FNNKD3, FNNKD4. FNNKD1 是  $2 \times K_1 \times 1$  模糊神经网络,  $a$  表示输出模糊组的中心,  $K_1$  表示模糊规则; FNNKD2 是  $2 \times K_2 \times 1$  模糊神经网络,  $b$  表示输出模糊组的宽度,  $K_2$  为模糊规则; FNNKD3 是  $2 \times K_3 \times 1$  模糊神经网络, 产生模糊输出  $c$ ,  $K_3$  为模糊规则; FNNKD4 是  $2 \times K_4 \times 1$  模糊神经网络, 产生模糊输出  $d$ ,  $K_4$  为模糊规则。

第三层是输出层,有两种情况:第一种情况是模糊语言输出表示矢量  $(a, b, c, d)$ ;第二种情况,  $a$  是清楚的数字输出。

### 3.3 探索知识库学习算法 (Heuristic Knowledge-Based Learning Algorithm, HLA)

为了尽可能地完善上述每个 FNNKD, 所以发展了 HLA<sup>[9]</sup>。为了有效的利用和方便,简化的学习算法用一个常数  $\gamma (\gamma \in [0, 1])$  描述如下:

假设  $n$  维输入数据矢量  $x^p$  (即  $x^p = (x_1^p, x_2^p, \dots, x_n^p)$ ) 和一维输出数据矢量  $y^p$ , 对于  $p=1, 2, \dots, N$ , 能量函数定义如下:

$$E^p = \frac{1}{2} [f(x_1^p, \dots, x_n^p) - y^p]^2 \quad (7)$$

为了简化,令  $E$  和  $f^p$  分别表示  $E^p$  和  $f(x_1^p, \dots, x_n^p)$ 。

学习算法如下:

#### step1 探索知识库初始化参数

I) 种类  $x_i^p$  对  $p=1, 2, \dots, N$  在不减小指令的情况下,而改变指令的一致性  $x_1^p, \dots, x_n^p$  和  $y^p$ 。

II) 划分  $x_1^p, x_2^p, \dots, x_n^p$  和  $y^p$  对于  $p=1, 2, \dots, N$ , 成为  $m$  数字段 (第  $k$  段数据  $K_{th}$  用于模糊规则的开始参数), 然后计算探索参数的初始值, 即  $a_i^p, a_n^p, b^p, b^n, \sigma_i^p, \sigma_n^p, \eta^p, \eta^n$  等。

III) 建立数值  $wLeft_i^p$  和  $wRight_i^p$ , 随机数  $i=1, 2, \dots, n; k=1, 2, \dots, m$ 。

#### Step2 根据梯度的计算方法,训练下列数据

$$I) \text{ 训练: } b^k(t+1) = b^k(t) - \lambda \frac{\partial E^p}{\partial b^k} \quad (8)$$

$$II) \text{ 训练: } \eta^k(t+1) = \eta^k(t) - \lambda \frac{\partial E^p}{\partial \eta^k} \quad (9)$$

$$III) \text{ 训练: } a_i^k(t+1) = a_i^k(t) - \lambda \frac{\partial E^p}{\partial a_i^k} \quad (10)$$

$$IV) \text{ 训练: } \sigma_i^k(t+1) = \sigma_i^k(t) - \lambda \frac{\partial E^p}{\partial \sigma_i^k} \quad (11)$$

V) 训练  $wLeft_i^k$  或  $wRight_i^k$

$$wRight_i^k(t+1) = wRight_i^k(t) - \lambda \frac{\partial E}{\partial wRight_i^k} \quad (12)$$

$$wLeft_i^k(t+1) = wLeft_i^k(t) - \lambda \frac{\partial E}{\partial wLeft_i^k} \quad (13)$$

#### Step3 发现模糊知识 (FUZZY IF-Then rules)

对于 FNNKD 所有的参数可作调整,不断完善。作为结果,从训练的数据中可以发现所有  $m$  的模糊规则。训练的 FNNKD 能够对新输入的数据产生新的价值。

#### Step4 结束。

上述在数据库中,用数字语言特性提取系统可产生典型的特性,而后用 CGNN 和 FGNN 处理数字-语言数据,这在 KDDM 中存在许多潜在的应用。下面讨论 CGNN 和 FGNN 在 KDDM 中有意义的应用。

## 4. 粒状数据的融合

现在讨论数字-语言数据的融合问题,也可用于理解预言性数据挖掘问题。

### 4.1 数字-语言数据预测乘法器

已知  $4 \times 4$  语言乘法器如表 1 所示,表中有 8 个不同的模糊

语言数据。around  $i$  用  $i$  表示,对于  $i=1, 2, 3, 4, 6, 8, 12, 16$ 。语言 around  $i$  意味着大约  $i$ , 即不确切的  $i$ , 对于每一个  $i$  是一个梯形模糊集,可用一个梯形特性矢量  $(i, i/10, i/4, i/2)$  表示。语言乘法器是  $Z=XY$ , 这里  $X, Y \in \tilde{1}, \tilde{2}, \tilde{3}, \tilde{4}$ , 并且  $Z \in \tilde{1}, \tilde{2}, \tilde{3}, \tilde{4}, \tilde{6}, \tilde{8}, \tilde{9}, \tilde{12}, \tilde{16}$ 。在表 1 中给出了 9 规则语言乘法知识,详细的 16 语言乘法规则储存在有关系的 4-领域数据库中如表 2 所示。

现在根据已知的 16-规则乘法数据库,如何发现一个较大的  $7 \times 7$  语言乘法数据库,即如何预测 33 个未知?的规则如表 3 所示。它表示  $X \in \tilde{1}, 1.5, \tilde{2}, 2.5, \tilde{3}, 3.5, \tilde{4}$  和  $Y \in \tilde{1}, 1.5, \tilde{2}, 2.5, \tilde{3}, 3.5, \tilde{4}$ 。X 和 Y 是混合数字-语言数据。下面利用 CNN 和 FGNN 分别解决数字-语言数据的融合问题。

表 1  $4 \times 4$  语言乘法表

|             |             |             |              |              |
|-------------|-------------|-------------|--------------|--------------|
|             | $\tilde{1}$ | $\tilde{2}$ | $\tilde{3}$  | $\tilde{4}$  |
| $\tilde{1}$ | $\tilde{1}$ | $\tilde{2}$ | $\tilde{3}$  | $\tilde{4}$  |
| $\tilde{2}$ | $\tilde{2}$ | $\tilde{4}$ | $\tilde{6}$  | $\tilde{8}$  |
| $\tilde{3}$ | $\tilde{3}$ | $\tilde{6}$ | $\tilde{9}$  | $\tilde{12}$ |
| $\tilde{4}$ | $\tilde{4}$ | $\tilde{8}$ | $\tilde{12}$ | $\tilde{16}$ |

表 2 有 4 个关系特性的语言乘法数据库

| $X(a1, b1, c1, d1)$            | $Y(a2, b2, c2, d2)$            | $Z=XY(a, b, c, d)$              |
|--------------------------------|--------------------------------|---------------------------------|
| $\tilde{1}(1, 0.1, 0.25, 0.5)$ | $\tilde{1}(1, 0.1, 0.25, 0.5)$ | $\tilde{1}(1, 0.1, 0.25, 0.5)$  |
| $\tilde{1}(1, 0.1, 0.25, 0.5)$ | $\tilde{2}(2, 0.2, 0.5, 1.0)$  | $\tilde{2}(2, 0.2, 0.5, 1.0)$   |
| $\tilde{1}(1, 0.1, 0.25, 0.5)$ | $\tilde{3}(3, 0.3, 0.75, 1.5)$ | $\tilde{3}(3, 0.3, 0.75, 1.5)$  |
| $\tilde{1}(1, 0.1, 0.25, 0.5)$ | $\tilde{4}(4, 0.4, 1.0, 2.0)$  | $\tilde{4}(4, 0.4, 1.0, 2.0)$   |
| $\tilde{2}(2, 0.2, 0.5, 1.0)$  | $\tilde{1}(1, 0.1, 0.25, 0.5)$ | $\tilde{2}(2, 0.2, 0.5, 1.0)$   |
| $\tilde{2}(2, 0.2, 0.5, 1.0)$  | $\tilde{2}(2, 0.2, 0.5, 1.0)$  | $\tilde{4}(4, 0.4, 1.0, 2.0)$   |
| $\tilde{2}(2, 0.2, 0.5, 1.0)$  | $\tilde{3}(3, 0.3, 0.75, 1.5)$ | $\tilde{6}(6, 0.6, 1.5, 3.0)$   |
| $\tilde{2}(2, 0.2, 0.5, 1.0)$  | $\tilde{4}(4, 0.4, 1.0, 2.0)$  | $\tilde{8}(8, 0.8, 2.0, 4.0)$   |
| $\tilde{3}(3, 0.3, 0.75, 1.5)$ | $\tilde{1}(1, 0.1, 0.25, 0.5)$ | $\tilde{3}(3, 0.3, 0.75, 1.5)$  |
| $\tilde{3}(3, 0.3, 0.75, 1.5)$ | $\tilde{2}(2, 0.2, 0.5, 1.0)$  | $\tilde{6}(6, 0.6, 1.5, 3.0)$   |
| $\tilde{3}(3, 0.3, 0.75, 1.5)$ | $\tilde{3}(3, 0.3, 0.75, 1.5)$ | $\tilde{9}(9, 0.9, 2.25, 4.5)$  |
| $\tilde{3}(3, 0.3, 0.75, 1.5)$ | $\tilde{4}(4, 0.4, 1.0, 2.0)$  | $\tilde{12}(12, 1.2, 3.0, 6.0)$ |
| $\tilde{4}(4, 0.4, 1.0, 2.0)$  | $\tilde{1}(1, 0.1, 0.25, 0.5)$ | $\tilde{4}(4, 0.4, 1.0, 2.0)$   |
| $\tilde{4}(4, 0.4, 1.0, 2.0)$  | $\tilde{2}(2, 0.2, 0.5, 1.0)$  | $\tilde{8}(8, 0.8, 2.0, 4.0)$   |
| $\tilde{4}(4, 0.4, 1.0, 2.0)$  | $\tilde{3}(3, 0.3, 0.75, 1.5)$ | $\tilde{12}(12, 1.2, 3.0, 6.0)$ |
| $\tilde{4}(4, 0.4, 1.0, 2.0)$  | $\tilde{4}(4, 0.4, 1.0, 2.0)$  | $\tilde{16}(16, 1.6, 4.0, 8.0)$ |

表 3 不完全的  $7 \times 7$  语言乘法表

|             |             |       |             |       |              |       |              |
|-------------|-------------|-------|-------------|-------|--------------|-------|--------------|
|             | $\tilde{1}$ | $1.5$ | $\tilde{2}$ | $2.5$ | $\tilde{3}$  | $3.5$ | $\tilde{4}$  |
| $\tilde{1}$ | $\tilde{1}$ | ?     | $\tilde{2}$ | ?     | $\tilde{3}$  | ?     | $\tilde{4}$  |
| $1.5$       | ?           | ?     | ?           | ?     | ?            | ?     | ?            |
| $\tilde{2}$ | $\tilde{2}$ | ?     | $\tilde{4}$ | ?     | $\tilde{6}$  | ?     | $\tilde{8}$  |
| $2.5$       | ?           | ?     | ?           | ?     | ?            | ?     | ?            |
| $\tilde{3}$ | $\tilde{3}$ | ?     | $\tilde{6}$ | ?     | $\tilde{9}$  | ?     | $\tilde{12}$ |
| $3.5$       | ?           | ?     | ?           | ?     | ?            | ?     | ?            |
| $\tilde{4}$ | $\tilde{4}$ | ?     | $\tilde{8}$ | ?     | $\tilde{12}$ | ?     | $\tilde{16}$ |

## 4.2 CGNN-Based 数据融合

2 输入-1 输出的 CGNN 结构如图 2 所示。基于表 2 中的语言数据,用 BPA 训练。实际上在图 2 中 4 个工作 CNNs, 用表 2 中对应的数据分别训练。在表 2 中,输入训练的数据是  $X(a1, b1, c1, d1)$  和  $Y(a2, b2, c2, d2)$ , 并且输出  $Z(a, b, c, d)$ 。CNN1 输入是  $a1$  和  $a2$  并且其输出是  $a$ , 训练 CNN1 是  $2 \times 81 \times 1$  CNN, 绝对平均训练误差是 0.0003; CNN2 的输入是  $b1$  和  $b2$  并且其输出是  $b$ , 训练 CNN2 是  $2 \times 81 \times 1$  CNN, 绝对平均训练误差是

0.027; CNN3的输入是c1和c2,训练 CNN3是 $2 \times 81 \times 1$  CNN,绝对平均误差是0.008; CNN4的输入是d1和d2,并且其输出是d,训练 CNN4是 $2 \times 81 \times 1$  CNN,绝对平均误差0.006.4个CNN训练之后,训练的CGNN能够产生33个新的乘法结果,

新的数据如表4所示,并且49-规则库如表5所示.显然,CGNN能够处理数字和语言数据如表4所示.33个乘积结果的平均预测绝对误差是0.325.因此CGNN能够从数字-语言数据库中学习,然后产生新的知识.

表4 33个数字-语言乘法规则新的数据库

| X           | Y           | Z=XY                                | X           | Y           | Z=XY                                |
|-------------|-------------|-------------------------------------|-------------|-------------|-------------------------------------|
| $\bar{1}$   | $1.\bar{5}$ | $1.\bar{41}(1.41,0.13,0.35,0.68)$   | $\bar{1}$   | $2.\bar{5}$ | $2.\bar{59}(2.59,0.03,0.50,1.02)$   |
| $\bar{1}$   | $3.\bar{5}$ | $3.\bar{57}(3.57,0.22,0.74,1.47)$   | $1.\bar{5}$ | $\bar{1}$   | $1.\bar{39}(1.39,0.03,0.065,0.13)$  |
| $1.\bar{5}$ | $1.\bar{5}$ | $1.\bar{88}(1.88,0.04,0.10,0.19)$   | $1.\bar{5}$ | $\bar{2}$   | $2.\bar{66}(2.66,0.06,0.14,0.29)$   |
| $1.\bar{5}$ | $2.\bar{5}$ | $3.\bar{48}(3.48,0.01,0.19,0.38)$   | $1.\bar{5}$ | $\bar{3}$   | $4.\bar{12}(4.12,0.13,0.34,0.67)$   |
| $1.\bar{5}$ | $3.\bar{5}$ | $5.\bar{14}(5.14,0.15,0.41,0.84)$   | $1.\bar{5}$ | $\bar{4}$   | $5.\bar{78}(5.78,0.53,1.10,2.31)$   |
| $\bar{2}$   | $1.\bar{5}$ | $2.\bar{74}(2.74,0.29,0.94,1.88)$   | $\bar{2}$   | $2.\bar{5}$ | $5.\bar{21}(5.21,0.07,0.99,1.97)$   |
| $\bar{2}$   | $3.\bar{5}$ | $7.\bar{32}(7.32,0.15,0.59,1.19)$   | $2.\bar{5}$ | $\bar{1}$   | $5.\bar{57}(5.57,0.23,0.64,1.27)$   |
| $2.\bar{5}$ | $1.\bar{5}$ | $3.\bar{54}(3.54,0.37,1.18,2.34)$   | $2.\bar{5}$ | $\bar{2}$   | $5.\bar{30}(5.30,0.54,1.69,3.38)$   |
| $2.\bar{5}$ | $2.\bar{5}$ | $7.\bar{00}(7.00,0.10,1.25,2.50)$   | $2.\bar{5}$ | $\bar{3}$   | $7.\bar{97}(7.97,0.76,1.94,3.88)$   |
| $2.\bar{5}$ | $3.\bar{5}$ | $8.\bar{85}(8.85,0.13,1.57,3.16)$   | $2.\bar{5}$ | $\bar{4}$   | $9.\bar{76}(9.76,0.93,2.36,4.76)$   |
| $\bar{3}$   | $1.\bar{5}$ | $4.\bar{01}(4.01,0.44,1.29,2.58)$   | $\bar{3}$   | $2.\bar{5}$ | $7.\bar{96}(7.96,0.14,1.50,3.01)$   |
| $\bar{3}$   | $3.\bar{5}$ | $10.\bar{68}(10.68,1.12,2.31,4.64)$ | $3.\bar{5}$ | $\bar{1}$   | $3.\bar{32}(3.32,0.34,1.06,2.13)$   |
| $3.\bar{5}$ | $1.\bar{5}$ | $5.\bar{15}(5.15,0.75,0.91,1.84)$   | $3.\bar{5}$ | $\bar{2}$   | $6.\bar{96}(6.96,0.66,1.69,3.14)$   |
| $3.\bar{5}$ | $2.\bar{5}$ | $8.\bar{62}(8.62,0.96,2.23,4.56)$   | $3.\bar{5}$ | $\bar{3}$   | $10.\bar{58}(10.58,1.10,2.10,4.54)$ |
| $3.\bar{5}$ | $3.\bar{5}$ | $12.\bar{36}(12.36,1.29,4.12,8.23)$ | $3.\bar{5}$ | $\bar{4}$   | $13.\bar{87}(13.87,1.41,4.35,8.90)$ |
| $\bar{4}$   | $1.\bar{5}$ | $5.\bar{76}(5.76,0.52,1.05,2.25)$   | $\bar{4}$   | $2.\bar{5}$ | $9.\bar{73}(9.73,0.91,2.32,4.74)$   |
| $\bar{4}$   | $3.\bar{5}$ | $13.\bar{82}(13.82,1.35,4.10,8.24)$ |             |             |                                     |

表5 用 CNN 产生完全的 $7 \times 7$ 语言乘法规则

|             | $\bar{1}$    | $1.\bar{5}$  | $\bar{2}$    | $2.\bar{5}$  | $\bar{3}$     | $3.\bar{5}$   | $\bar{4}$     |
|-------------|--------------|--------------|--------------|--------------|---------------|---------------|---------------|
| $\bar{1}$   | $\bar{1}$    | $1.\bar{41}$ | $\bar{2}$    | $2.\bar{59}$ | $\bar{3}$     | $3.\bar{57}$  | $\bar{4}$     |
| $1.\bar{5}$ | $1.\bar{39}$ | $1.\bar{88}$ | $2.\bar{66}$ | $3.\bar{48}$ | $4.\bar{23}$  | $5.\bar{14}$  | $5.\bar{78}$  |
| $\bar{2}$   | $\bar{2}$    | $2.\bar{74}$ | $\bar{4}$    | $5.\bar{21}$ | $\bar{6}$     | $7.\bar{32}$  | $\bar{8}$     |
| $2.\bar{5}$ | $2.\bar{57}$ | $3.\bar{54}$ | $5.\bar{30}$ | $7.\bar{00}$ | $7.\bar{97}$  | $8.\bar{85}$  | $9.\bar{76}$  |
| $\bar{3}$   | $\bar{3}$    | $4.\bar{01}$ | $\bar{6}$    | $7.\bar{96}$ | $\bar{9}$     | $10.\bar{68}$ | $\bar{12}$    |
| $3.\bar{5}$ | $3.\bar{32}$ | $5.\bar{15}$ | $6.\bar{96}$ | $8.\bar{62}$ | $10.\bar{58}$ | $12.\bar{36}$ | $13.\bar{87}$ |
| $\bar{4}$   | $\bar{4}$    | $5.\bar{76}$ | $\bar{8}$    | $9.\bar{73}$ | $\bar{12}$    | $13.\bar{82}$ | $\bar{16}$    |

表6 用 FGNN 产生完全的 $7 \times 7$ 语言乘法规则

|             | $\bar{1}$    | $1.\bar{5}$  | $\bar{2}$    | $2.\bar{5}$   | $\bar{3}$     | $3.\bar{5}$   | $\bar{4}$     |
|-------------|--------------|--------------|--------------|---------------|---------------|---------------|---------------|
| $\bar{1}$   | $\bar{1}$    | $1.\bar{27}$ | $\bar{2}$    | $2.\bar{73}$  | $\bar{3}$     | $3.\bar{62}$  | $\bar{4}$     |
| $1.\bar{5}$ | $1.\bar{23}$ | $2.\bar{11}$ | $3.\bar{37}$ | $4.\bar{49}$  | $5.\bar{04}$  | $5.\bar{26}$  | $6.\bar{23}$  |
| $\bar{2}$   | $\bar{2}$    | $2.\bar{87}$ | $\bar{4}$    | $5.\bar{13}$  | $\bar{6}$     | $6.\bar{82}$  | $\bar{8}$     |
| $2.\bar{5}$ | $2.\bar{57}$ | $3.\bar{74}$ | $5.\bar{13}$ | $6.\bar{52}$  | $7.\bar{83}$  | $8.\bar{65}$  | $10.\bar{23}$ |
| $\bar{3}$   | $\bar{3}$    | $4.\bar{51}$ | $\bar{6}$    | $7.\bar{52}$  | $\bar{9}$     | $10.\bar{44}$ | $\bar{12}$    |
| $3.\bar{5}$ | $3.\bar{60}$ | $5.\bar{31}$ | $6.\bar{97}$ | $8.\bar{71}$  | $10.\bar{48}$ | $12.\bar{15}$ | $13.\bar{76}$ |
| $\bar{4}$   | $\bar{4}$    | $6.\bar{11}$ | $\bar{8}$    | $10.\bar{18}$ | $\bar{12}$    | $13.\bar{91}$ | $\bar{16}$    |

### 4.3 FGNN-Based 数据融合

2输入-1输出 FGNN 结构如图3所示.同样,基于表2中4个语言数据特性的关系,用 HLA 算法训练.训练之后,在模糊数据中,FGNN 产生33个新的乘法规则,如表6所示.33个新的乘法规则的绝对平均误差是0.236,这比CGNN产生的误差要小,FGNN 比CGNN花费的时间要少.因此,FGNN产生的误差不仅小而且训练速度要快.

**结论** 基于粒状计算、神经计算、模糊计算、语言计算、多媒体特性提取和模式识别提出神经网络基础的 KDDM 方法.特别是从数据库中为数字-语言数据的融合和粒状知识的发现而设计了CGNN和FGNN.在数据库中CGNN和FGNN能够处理数字和语言的数据,能将模糊语言数据转换为相

应的数字特性,完成数字-语言数据的融合,从数字-语言数据库中发现粒状知识,并且预测新的知识.GNN 有很大的压缩能力,能将低水平的粒状数据压缩成为高水平的粒状知识.显然,粒度神经网络在处理多媒体数据,完成多媒体融合和发现粒状多媒体知识等各个领域中将得到广泛的应用.

### 参 考 文 献

- 1 Fayyad U M, Shapiro G P, et al. Advanced in knowledge discovery and data mining. California: AAAI/MIT press, 1996. 1~35
- 2 LU Hongjun, Setiono Rudy, Liu Huan. Effective data mining using neural networks. IEEE Transactions on Knowledge and Data Engineering, 1996, 8(6): 957~961
- 3 Lin T Y. Granular computing on binary relations I: Data mining and neighborhood systems. In Rough sets in Knowledge Discovery, A. Skowron and L. Polkowski, Eds. Berlin. Germany: Springer-Verlag, 1998. 121~140
- 4 Jayashi Y, Nakai M. Automated extraction of fuzzy IF-THEN rules using neural networks. T. IEEE Japan, 1990, 110-C(3): 198~206
- 5 Sugeno M, Yasukawa T. A fuzzy logic based approach to qualitative modeling. IEEE Trans. Fuzzy Syst., 1993, 1: 7~31
- 6 Yao Y Y. Granular computing using neighborhood systems. Engineering Design and Manufacturing, London, U. K. Springer-Verlag 1999. 539~553
- 7 Westphal C, Blaxton T. Data Mining Solutions-Methods and Tools for Solving Real-World Problems. New York: Wiley, 1998
- 8 Yao Y Y, Zhang N. Potential applications of granular computing in knowledge discovery and data mining. In: Proc. World Multiconf. Systemics, Cybernetics and Informatics, Vol. 5, 1999. 573~580
- 9 Zhang Y-Q, Kandel A. Compensatory Genetic Fuzzy Neural Networks and Their applications, Ser. series in machine perception artificial intelligence. Singapore: word scientific, 1998, 30