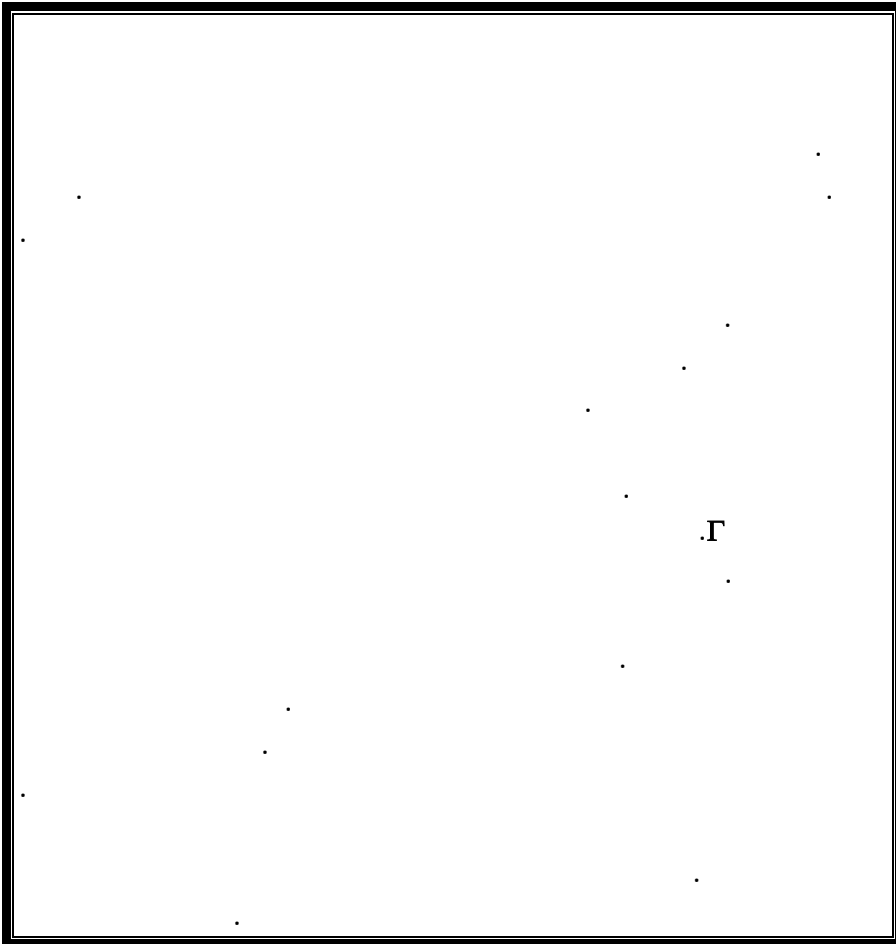


1



1

[1]

.[6]

.[5]

[10]

.[11]

[7]

.[13]

.[2]

.[8] [16]

.[12]

[3]

.[4]

[15]

. [17]

-2

- 1.2

[9] Visual Prolog 6.1
Rule Based Systems
() Control Structure

Backward Chaining

Forward Chaining

Hybrid Control Structures

Indexing of Predicate Expressions

. [14]

-2.2

- 3.2

-4.2

W :

W

(1) = [W1, W2, W3, W4, W5, W6, W7, W8, ...]

(1, 0, 0, 0, ...) (0)

... (0, 0, 1, 0, ...) (0, 1, 0, 0, ...)

-5.2

()

(21×23)

() 1

.() 0

W3= W2= W1= :

W8= W7= W6= W5= W4=

W12=2 W11=1 W10=1 W9=1

W16=1 W15=2 W14= W13=2

W20= W19=3 W18=3 W17=2

.W21=

(78×43)

1

) 0 ()

.(

W3= W2 = W1 = :
 W8= W7= W6= W5= W4=
 W12=1 W11= W10=1 W9=1
 W15=1 W14= W13=2
 W20=2 W19=1 W18= W17= W16=
 W25=2 W24=2 W23=2 W22=1 W21=3
 W29=3 W28=3 W27=2 W26=
 W33=3 W32=3 W31=4 W30=1
 W37= W36=5 W35= W34=4
 W41= W40=6 W39=2 W38=5
 .W43=4 W42=

-6.2

" " " " :
 " " " " " "
 " " + " " "
 " " " "

-7.2

field_v_1()
field_v_2()
field_v_3()
field_v_4()
field_v_5()
.....

field_v_2 2*8 field_v_1
3*8

$$\text{field_v_1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\text{field_v_2} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

()

field_n_1()
field_n_2()
field_n_3()
field_n_4()
field_n_5()
field_n_6()
field_n_7()
field_n_8()
field_n_9()
.....

:

"

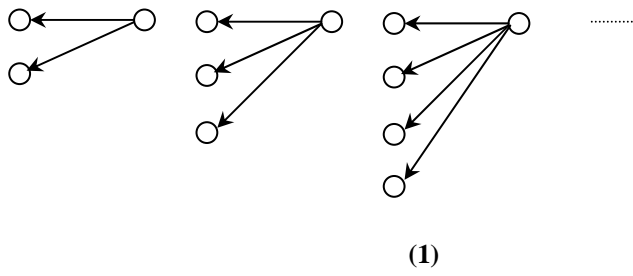
"

-8.2

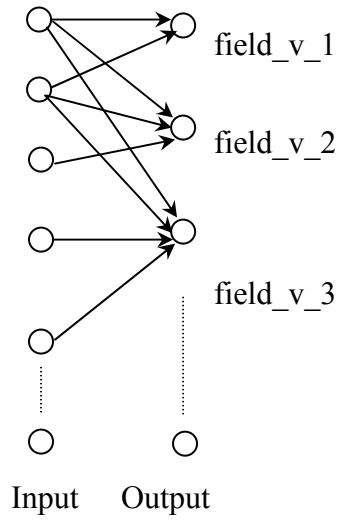
()

field_v_1()

" " " " " field_v_2
(1) " "



(2)

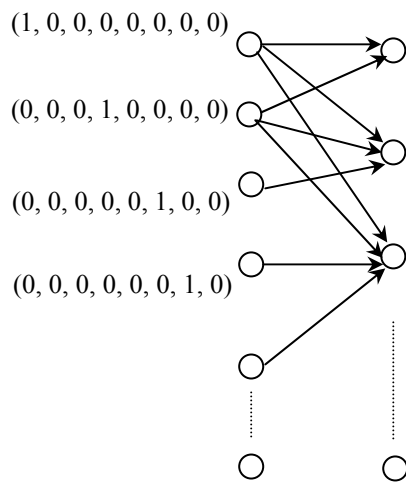


(2)

(1,
(0, 0,

.
0, 0, 0, 0, 0, 0, 0)
.0, 1, 0, 0, 0, 0)

(3)



$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

(3)

-9.2

-1.9.2

()

()

```

      :
clauses
onControlOpen(_CtrlID, _CtrlType, _CtrlWin, _CtrlInfo) = handled(0):-
FileName = vpiCommonDialogs::getFileName (
"* . Txt", ["Txt files (*.txt)", "*.txt", "All files " , " *.*" ]
"Load gram Data Structures"
[] ". _) !
reconsult(FileName)
stdIO::writef("\nData Structures % loaded\n", FileName) .
onControlOpen(_CtrlID, _CtrlType, _CtrlWin, _CtrlInfo) = handled(0).
clauses
reconsult(FileName):-
retractAll(_ , gramDB)
file::consult(FileName, gramDB).

```

-2.9.2

```

      :
      :
      .
      .
      .
      :
      :
      .

```

-
-
-
-

" " •
•
•
:

{ } = F1=
{ } = F2 =
" " = F3 =
{ } = F4=
{ } = F5 =
} = F6 =
{
{F71 ,F72 , F73} = F7 =
{ } =F71 =
{ } = F72 =
} =F7 =
{
} = F8 =
{

} = F9 =

{ } = F10 =

} = F11 =

{

{ } = F12 =

{ } = F13 =

{ } = F14 =

" " " ()

-3.9.2

:

.()

:

" " " "

-4.9.2

:

:

:

:

:

-5.9.2

:

" " " "

:

:

" " :
 " " "
 .
-6.9.2

" " "
 " " "
 " "

-3.

-1.3

$n_i =$
 $43 + 21 =$ $\max(\text{component}_i)$
 $n_o = \sum \text{field}_i$ 64
 $78 + 23 = 101$
 $.n_i * n_o$

...

-2.3

(21)

(43)

(2.5)

$$\frac{dw_i}{dt} = lr(I_i - w_i) \quad (1)$$

i w_i i I_i $0 < lr \leq 1$

MATLAB

I W n
 Γ .OUT
 :

$$\text{Active_Node(out)} = \text{Filter}(I0\Gamma0W) \quad (2)$$

$$\begin{aligned} \text{out}(n) = & b(1) I(n) + b(2) I(n - 1) + \dots + b(n_i + 1) I(n - n_i) \\ & - a(2) \text{out}(n - 1) - \dots - a(n_o + 1) \text{out}(n - n_o) \end{aligned} \quad (3)$$

$$\text{out}(z) = \frac{b(1) + b(2)z^{-1} + \dots + b(n_i + 1)z^{-n_i}}{1 + a(2)z^{-1} + \dots + a(n_o + 1)z^{-n_o}} I(z) \quad (4)$$

-3.3

(2)

```
      :  
clauses  
onFileNew(_MenuTag) = handled(0):-  
  Dialog = myDialog::new(),  
  Dialog:show(thisWin),  
  assert (mydialog_fact(Dialog)).
```

-4.3

)

(

(/)

-5.3

-4.

+0.001 _0,001

64

$l_r = 0.01$

.101

. (2.9)

-1.4

23×3

"

"

:

List = [" " " " " " ""]

:()

3 =

Γ

" "

:

6

:

- .1 + + =
- .2 + + =
- .3 + + =
- .4 + + =
- .5 + + =
- .6 + + =
- .7 + + =
- .8 + + =
- .9 + + =
- .10 + + =
- .11 + + =
- .12 + + =
- .13 + + =
- .14 + + =

:

•

•

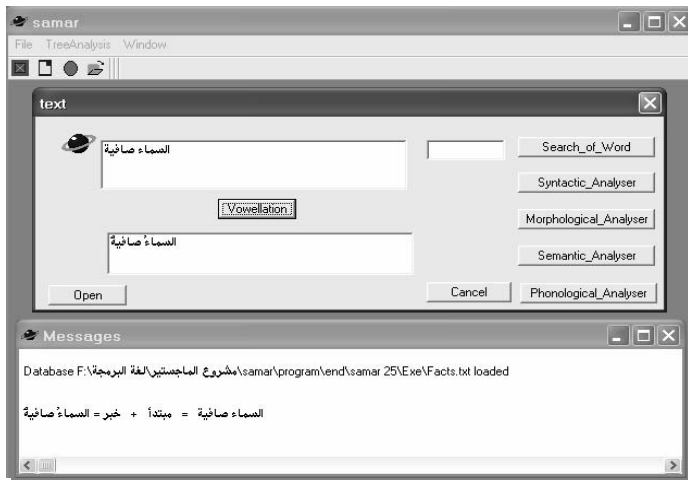
.()

" "

•

:

= + + =



-2.4

78×3

" "

:

List=[" " " ""]

:()

2 =

.((2))

Γ

: " "

" "

:

$$+ = .1$$

$$+ = .2$$

$$+ = .3$$

$$+ = .4$$

$$+ = .5$$

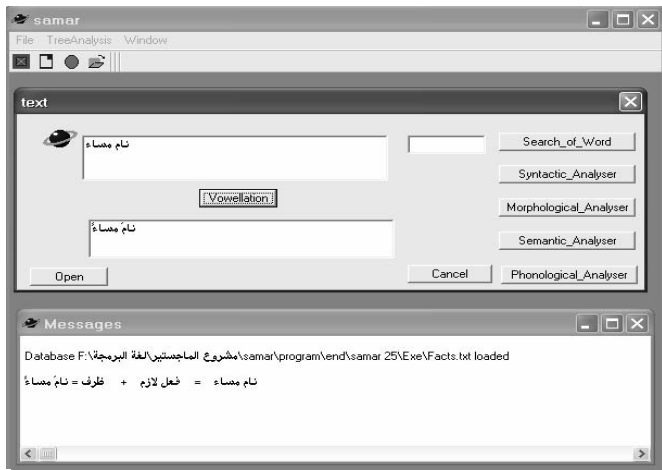
:

3

-
-

:

$$= + =$$



:

List = [" " " " ""]

:

3 =

:

	+		+		=	.1
						.2
						.3
			+	+	=	.4
+		+			=	.5
+		+			=	.6

$$\begin{aligned} & + \quad + \quad = \quad .7 \\ + \quad + \quad = \quad .8 \\ & \quad + \quad + \quad = \quad .9 \\ & \quad \quad + \quad + \quad = \quad .10 \\ & \quad \quad + \quad + \quad = \quad .11 \\ & \quad \quad + \quad + \quad = \quad .12 \end{aligned}$$

:

$$\bullet$$
$$.12 \quad 11 \quad 10$$

11

•

:

$$= \quad + \quad + \quad =$$
$$-5$$

	2003	.1
	2000	.2
2000		.3
()	2001	.4
	()	
	2003	.5
/	2001	.6
	2004	.7

8. Grichnik Anthony, 2003, Artificial Intelligence, Strategy & Technology Manager Caterpillar Inc.

9. Holstvej.H.J., 2003, Visual Prolog version 6.1. Prolog Development Center A\S, Denmark.

10. Kurfess Franz J, 2002, Knowledge-Based Systems, Computer Science Department Cal Poly.

11. Lee Steve, 2003, Internet Communication in Arabic Language, Expert Group Meeting on Promotion of Digital Arabic Content, Economic and Social Commission for Western Asia, Beirut.

12. Meikiu Lo and others, 2003, Natural Language Interfaces to Databases, Kadir Has University.

-
13. Myers Mark and others, 2004, Knowledge-based Systems, Ed Elrod.
 14. Stotts and others, 2003, Logic Programming with Prolog: Resolution, Unification, Backtracking, Comp 144 Programming Language Concepts, The University of North Carolina at Chapel Hill.
 15. Sugai George, 2003, Data-based Decision Making: Basics, OSEP, Center on Positive Behavioral Interventions & Supports.
 16. Tecuci Gheorghe, 2003, CS 580 Introduction to Artificial Intelligence, Learning Agents Laboratory, Department of Computer Science, George Mason University.
 17. Tecuci Gheorghe, 2003, CS 782 Machine Learning, Inductive Learning from Examples: Version Space Learning, Learning Agents, Laboratory Computer Science Department George Mason University, Learning Agents Laboratory.

.2007/2/14