

TF de IA: relatório sobre o uso de Redes Neurais

Lígia Márcia do Nascimento Teixeira¹, Eriky Rodrigues de Souza², Mateus Mota Nóbrega³, Gabriel César Tavares Ferreira⁴

¹*Instituto de Computação, Universidade Federal do Amazonas (UFAM), Manaus-AM, Brazil*
{lmnt,eriky.rodrigues,mateus.mota,gabriel.cesar}@icomp.ufam.edu.br

1 Introdução

Os modelos de aprendizagem e suas evoluções compõem uma área do Aprendizado de Máquina. Deste modo, o aprendizado por indução simbólica e conexionista determina que o método de aprendizagem pode ser realizado por meio da indução de fatos em sistemas que simulam o funcionamento de redes neurais. Portanto, baseando-se na implementação de modelos que, ao obterem um determinado conhecimento prévio (informações relevantes ao sistema que podem ser utilizados para a criação de hipóteses), podem gerar cláusulas verdadeiras ou falsas (exemplos positivos e negativos, ou EX e NEX), foi possível realizar os experimentos no sistema CILP++.

2 Background

A Programação em Lógica Indutiva (ILP) é uma área de Machine Learning investiga a construção indutiva de teorias de cláusulas de Horn, de primeira-ordem, a partir de exemplos e de um conhecimento preliminar [Muggleton and De Raedt, 1994].

Além do ILP, temos também a Programação em Lógica Indutiva e Conexionista (CILP) que se trata de um sistema que integra um programa lógico proposicional representando o conhecimento prévio e uma Rede Neural Artificial (ANN) [França, 2012].

O CILP++ é um sistema que integra o CILP a um método chamado Proposicionalização de Cláusulas Mais Específicas (BCP), responsável por transformar cada exemplo de primeira ordem em cláusulas mais específicas e então as usa como exemplos proposicionais [França, 2012].

3 Descrição do Problema

A implementação pode ser consultada aqui

Para o problema, foi utilizada a descrição de um conhecimento prévio (background knowledge) para a definição de determinadas relações no conjunto. Essas relações são `parent(X,Y)`, `male(X)` e `female(X)`, e podem ser definidas como "X é pai de Y", "X é homem" e "X é mulher", respectivamente.

1. <code>parent(pam,bob).</code>	6. <code>parent(pat,jim).</code>	11. <code>female(pam).</code>
2. <code>parent(tom,bob).</code>	7. <code>parent(ann,eve).</code>	12. <code>female(liz).</code>
3. <code>parent(tom,liz).</code>	8. <code>male(tom).</code>	13. <code>female(ann).</code>
4. <code>parent(bob,ann).</code>	9. <code>male(bob).</code>	14. <code>female(pat).</code>
5. <code>parent(bob,pat).</code>	10. <code>male(jim).</code>	15. <code>female(eve).</code>

Figure 1: Base de Conhecimento

Tendo a base de conhecimento, também foram definidas quinze cláusulas candidatas para `hd(X)`. A cláusula `hd(X)` significa "hasDaughter", e pode ser definida como "X tem uma filha".

```
hd(X) ← male(X), parent(Y,X).
hd(X) ← male(X), parent(X,Y).
hd(X) ← female(X), parent(Y,X).
hd(X) ← male(X), parent(Y,X).
hd(X) ← female(X), parent(X,Y), female(Y).
hd(X) ← male(X), parent(Y,X), female(Y).
hd(X) ← female(X), parent(X,Y), female(Y).
hd(X) ← female(X), parent(Y,X), male(Y).
hd(X) ← female(X), parent(X,Y), female(Y).
hd(X) ← male(X), parent(Y,X), parent(Y,Z).
hd(X) ← male(X), parent(X,Y), female(Y).
hd(X) ← male(X), parent(X,Y), female(Y), parent(Y,Z), male(Z).
hd(X) ← female(X), parent(Y,X), male(Y), parent(Z,Y), female(Z).
hd(X) ← female(X), parent(Y,X), male(Y), parent(Y,Z), female(Z).
hd(X) ← female(X), parent(Y,X), male(Y), parent(Y,Z), female(Z), parent(Z,W), male(W).
```

Figure 2: Clausulas candidatas

Dentre essas cláusulas, dez geraram exemplos negativos, ou NEX. O motivo dos NEX terem sido gerados foram os mesmos para nove das dez cláusulas: não existe nenhuma relação que defina X como pai de uma mulher. A última cláusula apresentou um mo-

tivo diferente para ser um exemplo negativo: apesar de ser definido X como pai de Y, não existe relação que defina Y como uma mulher. As demais cláusulas não incluídas são exemplos positivos, ou seja, existe uma relação de $hd(X)$ pois X é pai ou mãe de uma mulher.

```

~hd(X) :- male(X), parent(Y,X).
~hd(X) :- female(X), parent(Y,X).
~hd(X) :- male(X), parent(Y,X).
~hd(X) :- male(X), parent(Y,X), female(Y).
~hd(X) :- female(X), parent(Y,X), male(Y).
~hd(X) :- male(X), parent(Y,X), parent(Y,Z).
~hd(X) :- female(X), parent(Y,X), male(Y), parent(Z,Y), female(Z).
~hd(X) :- female(X), parent(Y,X), male(Y), parent(Y,Z), female(Z).
~hd(X) :- female(X), parent(Y,X), male(Y), parent(Y,Z), female(Z), parent(Z,W), male(W).
~hd(X) :- male(X), parent(X,Y).

```

Figure 3: Cláusulas negativas

Após isso, foi necessário aplicar o CILP++ p realizar o experimento na base de conhecimento pré estabelecida, e então gerar exemplos positivos e negativos com objetivo de identificar quais cláusulas seriam suficientes para $hd(X)$, ou seja, quais cláusulas teriam podem expressivo suficiente para tornar $hd(X)$ um exemplo positivo.

4 Experimento e análises

Foram utilizadas cláusulas para solucionar o problema proposto descrito acima. Para ser usada nas rede neurais, algumas configurações foram necessárias: uma com pesos positivos e outras com pesos negativos.

Optou-se pela escolha de duas cláusulas que geram resultados positivos pelo Amao. Outras foram alteradas para se obter quatro cláusulas que gerassem resultados negativos, dessa forma, temos em sua composição seis cláusulas com seis literais, como pode ser observado abaixo:

```

hd(X):- female(X), parent(X,Y), female(Y).
hd(X):- male(X), parent(X,Y), female(Y).
~hd(X):- female(X), parent(X,Y), male(Y).
~hd(X):- male(X), parent(X,Y), male(Y).
~hd(X):- female(X), parent(Y,X).
~hd(X):- male(X),parent(Y,X).

```

Para que as cláusulas componham uma rede neural, é necessário definir os valores de inicialização para os pesos e as bias das camadas escondida e de saída. A tabela abaixo mostra seus respectivos valores.

Com isso temos os resultados iniciais da rede como mostra a tabela abaixo. O peso w resultou em um valor muito alto, aplicando a normalização

Cl	K	pl	nl	ul	max	q	m
Claúsulas	Número de literais	Número de literais positivos	Número de literais negativos	Número de cláusulas com o mesmo head que essa	o maior entre kl e ul	número de cláusulas	número total de literais
C1	3	3	0	6	6	6	7
C2	3	3	0	6	6	-	-
C3	3	3	0	6	6	-	-
C4	3	3	0	6	6	-	-
C5	2	2	0	6	6	-	-
C6	2	2	0	6	6	-	-

Table 1: Definição dos valores de inicialização das cláusulas para a Rede Neural

thetaHL	thetaOL	Max	Amin	W	Wn	q	m
bias da camada escondida	bias da camada de saída	Max	Amin	pesos	pesos normalizados	número de cláusulas	número total de literais
1,714285714	4,285714286	6	0,7142857143	3,23E+16	1,39E-17	6	7
1,714285714	-	-	-	-	-	-	-
1,714285714	-	-	-	-	-	-	-
1,714285714	-	-	-	-	-	-	-
0,8571428571	-	-	-	-	-	-	Re -
0,8571428571	-	-	-	-	-	-	-

Table 2: Dados Iniciais da rede

Wn, o resultado foi um valor muito baixo, que foi arredondado para 1 no setup da matriz de peso da camada escondida.

nome	female(X)	male(X)	parent(X,Y)	parent(Y,X)	female(Y)	male(Y)	hd(X)
0 Tom	0	1	1	0	1	0	1
1 Bob	0	1	1	0	1	0	1
2 Ann	1	0	1	0	1	0	1
3 Pam	1	0	1	0	0	1	-1
4 Pat	1	0	1	0	0	1	-1
5 Liz	1	0	0	1	0	0	-1
6 Eve	1	0	0	1	0	0	-1
7 Jim	0	1	0	1	0	0	-1

Figure 4: Base de dados

A base de dados foi baseada na árvore familiar apresentada no problema proposto. Com isso, para cada pessoa X, verificamos que para a cláusula ser verdadeira a pessoa X era um indivíduo que tinha filha(1), caso contrário(-1). Desta forma, pode-se constituir a base de dados conforme os valores de output gerados. A partir do treinamento realizado, também pode ser conferido o ajuste dos pesos que foram continuamente alterados para garantir a aprendizagem por parte do sistema.

Pesos
1.
1.
-1.
-1.
-1.
-1.

Figure 5: Valores de saída dos pesos

5 Conclusão

Com o experimento foi possível colocar em prática grande parte dos conhecimentos adquiridos durante as aulas. Ao iniciar a matriz de pesos e a bias, foi a melhor abordagem para a Rede Neural, portanto concluiu-se que, ao iniciar as matrizes de peso com um valor fixo, é produzido um resultado constante acarretando a estagnação da rede neural, ou seja, ela não evolui. Definiu-se também os pesos dos neurônios para valores positivos ou negativos de acordo com suas respectivas cláusulas, os resultados apresentados para a base determinada foi satisfatória. Mesmo para uma inicialização aleatória, os resultados obtidos foram satisfatórios.

REFERENCES

- França, M. V. M. (2012). *APRENDIZADO RELACIONAL ATRAVÉS DO USO DE CLÁUSULAS MAIS ESPECÍFICAS NO SISTEMA C-IL2P*. PhD thesis, Universidade Federal do Rio de Janeiro.
- Muggleton, S. and De Raedt, L. (1994). Inductive logic programming: Theory and methods. *The Journal of Logic Programming*, 19:629–679.