



Implementação de uma Rede Neural para o problema do Trem de Michalsky

Bruno Nascimento Maciel, Gabriel da Silva Freitas, Gabriel Luciano Nunes, Guilherme Silveira Duarte, João Pedro Abensur

¹Instituto de Computação – Universidade Federal do Amazonas (UFAM)
Av. Gen. Rodrigo Octávio 6200, Coroado – 69080-900 – Manaus – AM – Brasil

{gabriel.freitas,gabriel.luciano,guilherme.silveira}@icomp.ufam

1. Introdução

O objetivo deste trabalho é construir uma rede neural do zero a partir dos conceitos de obtidos na disciplina de Inteligência Artificial. Para isso o problema do Trem de Michalsky foi utilizado como escopo a ser resolvido da nossa rede neural. Primeiramente é abordado na seção 1 um referencial teórico a respeito dos Sistemas Neuro-Simbólicos, em seguida na seção 2 explicamos o problema do trem e em seguida discutimos sobre a rede neural implementada e os resultados obtidos.

2. Referencial Teórico: Sistemas neuro-simbólicos

Segundo [Kononenko 2010] os algoritmos de Aprendizado de máquina podem ser classificados em três grandes grupos: Métodos de Padrões Estatísticos; Redes Neurais Artificiais e Aprendizado indutivo de regras Simbólicas. Para nós interessa os dois últimos modelos de aprendizado de máquina apresentados

Os sistemas neuro-simbólicos combinam pontos fortes de fracos de dois ramos da inteligência Artificial. O primeiro é a IA simbólica que utiliza conceitos de indução, dedução e predicados de lógica para resolver problemas de aprendizado de máquina. Ela tenta representar explicitamente o conhecimento humano por meio de símbolos legíveis por humanos e regras que permitem a manipulação desses símbolos

A segunda parte abrange redes neurais que é uma abordagem mais complexa no qual o sistema de IA "simula", mesmo que de forma simples, os neurônios de um cérebro humano. Uma rede neural consiste em transformar dados de entrada para n camadas de neurônios e transformar esses dados através de variáveis (denominadas pesos e bias), aplicar uma função de ativação com funções matemáticas de linearização para gerar um padrão de saída esperado. A figura 1 apresenta uma visão geral do que são os neurônios, camada de neurônios, entradas e saídas.

A abordagem simbólica conexionista se torna uma ponte que liga os dois campos e preenche a lacuna de cada um destes com seus respectivos pontos positivos. O primeiro

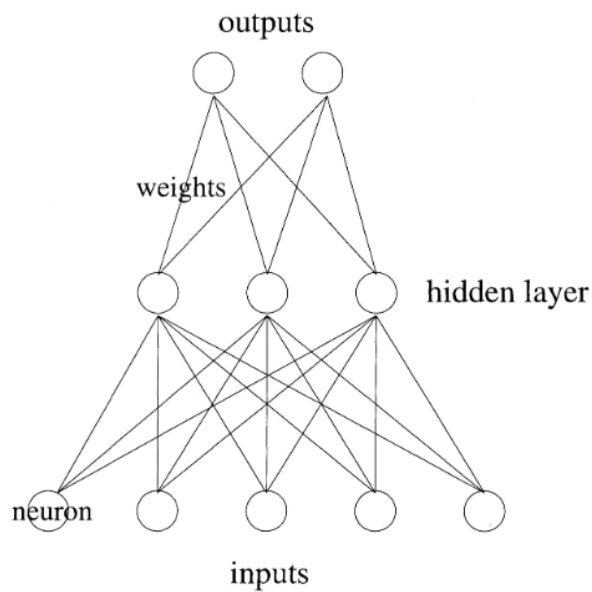


Figura 1. Representação de uma Rede Neural

trabalho que utilizou este método foi desenvolvido por McCulloch e Pitts e forneceu insights fundamentais sobre como a lógica proposicional pode ser processada usando redes neurais artificiais simples. Segundo [McCulloch et al. 1943]:

Os métodos neuro-simbólicos são utilizados em situações em que há conhecimento parcial sobre um domínio de aplicação antes do treinamento, seria desejável ter meios para orientar algoritmos de aprendizagem conexionistas usando esse conhecimento. Isso ocorre especialmente em tarefas de aprendizagem que tradicionalmente se enquadram no âmbito da inteligência artificial simbólica e que são caracterizadas por interdependências complexas e frequentemente recursivas entre peças de conhecimento representadas simbolicamente. Esta é uma área que está em constante evolução atualmente.

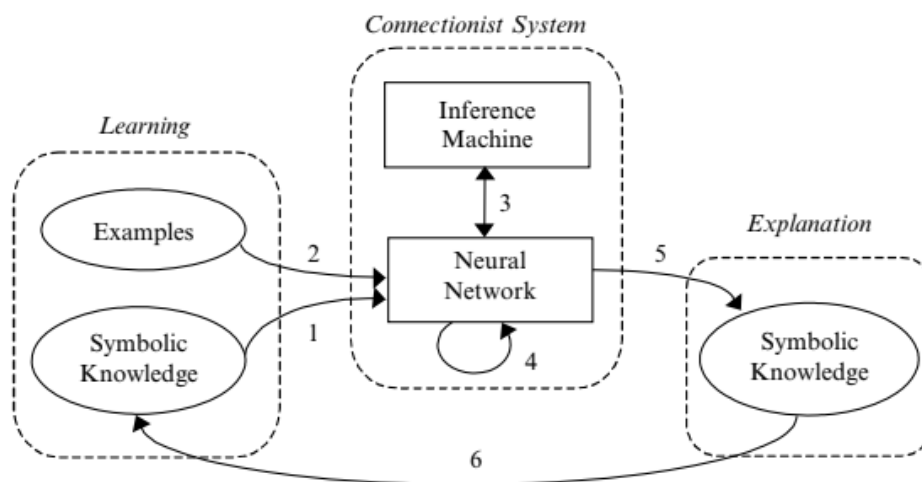


Figura 2. Ciclo de uma rede neural-simbólica

3. Trem de Michaelsky

Desenvolvido por [Michalky 1980] para métodos de inferência sobre objetos, o problema do trem de Michaelsky procura determinar se um trem está indo para o oeste ou para o leste. Essa classificação é determinada através de características dos trens observadas na figura 6, a partir disso demanda-se os atributos base de entradas. Os dados a serem analisados e posteriormente classificados são:

1. quantidade_vagoes: valores de (3 a 5);
2. quantidade_cargas_diferentes valores de (1 a 4);
3. eixo_vagao: valores de (0, 2 a 3 cada)
4. comp_vagao: valores de (curto ou longo -> 0, 1, 2) —————> 0 = sem vagão, 1 = curto, 2 longo
5. formato_vagao: valores de (0 a 9) —————> 0 = sem vagao, 1 a 9 representa formatos
6. quant_cargas: valores de (0 a 3) —————> 0 = sem vagao, >0 = quantidade
7. forma_carga: valores de (0, (sem vagao), circulo (1), hexagono (2), retangulo (3), triangulo (4))
8. circulo_adjacente_circulo: valor booleano
9. circulo_adjacente_retangulo: valor booleano
10. circulo_adjacente_triangulo: valor booleano
11. hexagono_adjacente_retangulo: valor booleano
12. hexagono_adjacente_triangulo: valor booleano
13. hexagono_adjacente_hexagono: valor booleano
14. retangulo_adjacente_retangulo: valor booleano
15. retangulo_adjacente_triangulo: valor booleano
16. triangulo_adjacente_triangulo: valor booleano

Para a treinamento de nossa rede na questão 1: 10 trens foram avaliados como entrada, dentre esses, 5 estão indo para o oeste e outros 5 para o leste. Vale ressaltar que os vagões foram observados da esquerda para a direita Para o treinamento na questão 2: 20 trens foram avaliados, em que 10 vão para oeste e 10 vão para leste

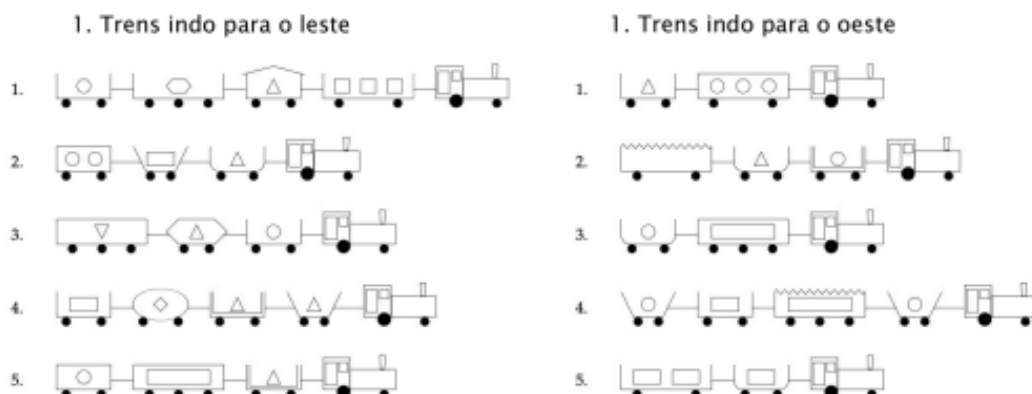


Figura 3. Trens de entrada para o treinamento da rede

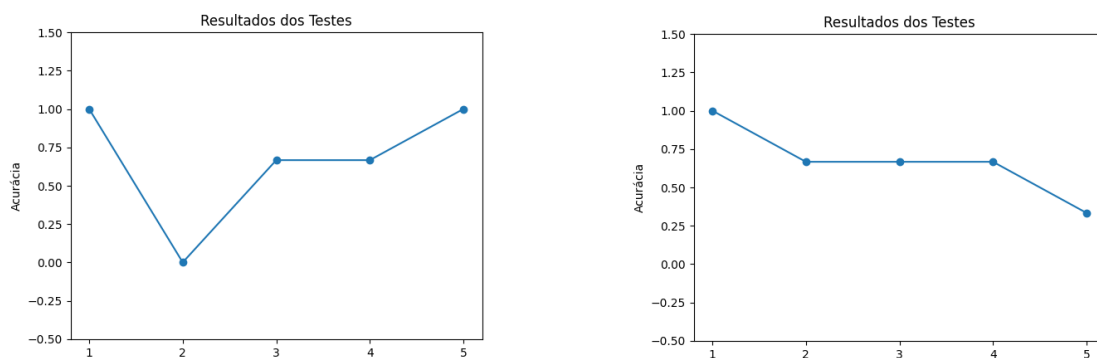


Figura 4. Resultados o Experimento 1 à esquerda e Experimento 2 à direita

4. Resolução da Classificação

4.1. Questão 1

Para essa questão realizamos duas coleções de testes conforme está na solicitado na especificação. A segunda coleção de experimentos foi feita sendo que agora a rede foi treinada retirando dois trens indo para oeste e um trem para leste. Por fim, foi feita a comparação entre os dois experimentos. Os resultados são apresentados na figura 5.



Figura 5. Comparação entre os dois testes especificados

Observamos que no primeiro teste, ambos experimentos tiveram acurácia 1. No segundo, tivemos uma queda para ambos. Na terceira e quarta coleção de testes, a acurácia ficou manteve 7.5. No 5º teste, o Experimento 1 teve a acurácia de 1 enquanto que o Experimento 2 obteve uma acurácia de 0.5.

5. Questão 2

Para essa questão, construímos uma rede com meta-dados para conectar os conceitos das premissas. Para isso, utilizamos as cláusulas apresentadas na especificação da questão.

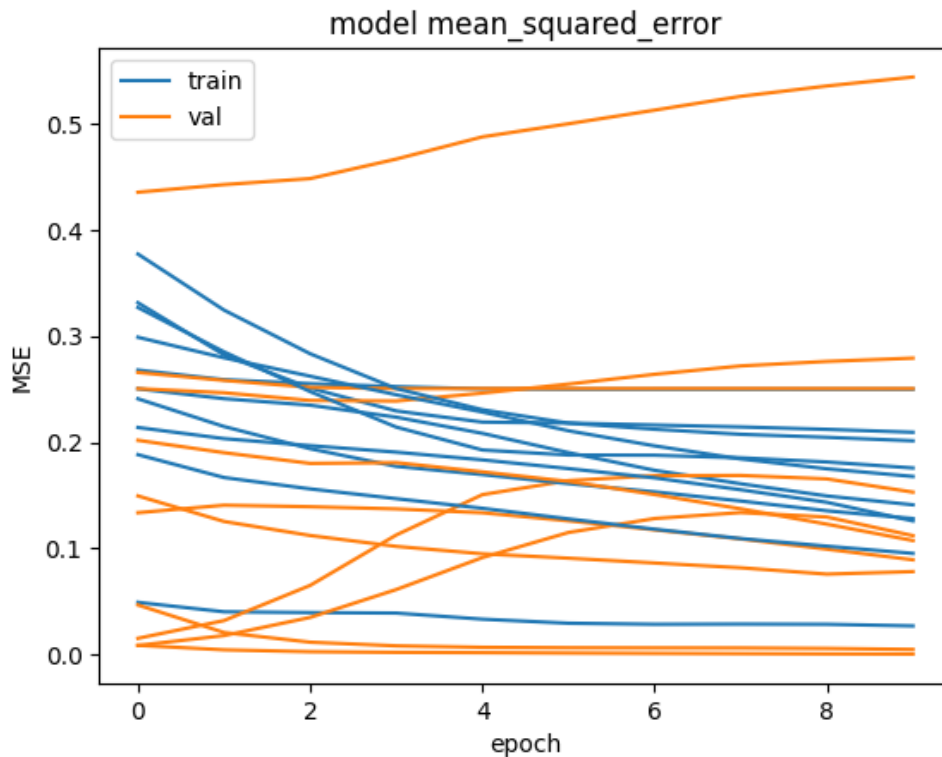


Figura 6. Acurácia do treinamento da questão 2 por 10 épocas

Referências

- Kononenko, I. (2010). *Application of Machine Learning to Medical Diagnosis*. Cengage.
- McCulloch, H., Pitts, J., and Widom, J. (1943). *A logical calculus of the ideas immanent in nervous activity*. *Bulletin of Mathematical Biophysics*. Prentice Hall, 2th edition.
- Michalky, R. S. (1980). *Pattern Recognition as Rule-G uided Inductive Inference*. IEEE.