

TP555 – Inteligência Artificial e Machine Learning

Josino Villela da Silva Neto – Matrícula: 854 – Curso: Mestrado em Engenharia de Telecomunicações

Lista de Exercícios #1

1. a) Inteligência: É a capacidade de raciocinar e discernir diante de diferentes situações.

b) Inteligência artificial: É a capacidade de máquinas raciocinarem, com base em dados de entrada, diante de diferentes situações. Corresponde a imitação do comportamento do cérebro humano pelas máquinas.

c) Aprendizado de máquina: É a capacidade de máquinas aprenderem com a experiência.

2.

Do enunciado:

E: Experiência de aprendizado; T: Tarefa a ser desempenhada; D: Medida de desempenho.

Resposta:

E: Dados climáticos históricos; T: Prever o tempo; D: A probabilidade de prever corretamente o tempo de uma data futura.

3.

Do enunciado:

E: Experiência de aprendizado; T: Tarefa a ser desempenhada; D: Medida de desempenho.

Resposta:

T: Prever o tempo; E: Dados climáticos históricos.

4. Tarefa: prever o tempo nos dias seguintes; Dados de entrada: Ensolarado, nublado e chuvoso.

A tarefa de prever o clima seria tratada como classificação. Optou-se por classificação, pois de acordo com o enunciado pode-se fazer apenas 3 tipos de previsões para cada dia, ou seja, há finitas opções de previsões.

5. Tarefa: prever o preço de determinada ação no mercado financeiro; Dados de entrada: preços de determinada ação nos dias anteriores.

A tarefa de prever o preço de determinada ação “amanhã”, é uma tarefa de regressão. Optou-se por regressão, pois os preços das ações podem assumir infinitos valores.

6. Para permitir que o robô aprenda a andar em diferentes ambientes desconhecidos, é utilizado o algoritmo de aprendizado por reforço.

7. Para seguir clientes de uma grande empresa de e-commerce em vários grupos, podem ser utilizados algoritmos de aprendizado supervisionado (Os grupos de clientes já são definidos previamente, ao passo que novos clientes são adicionados a estes grupos), não supervisionado (a máquina é que irá criar e definir os grupos, com base nos dados fornecidos dos clientes), e por fim semi-supervisionado (mix das opções apresentadas anteriormente).

8. a) Alocação de recursos em redes móveis.

Método utilizado: Q-Learning.

Artigo lido: Q-Learning Algorithm for VoLTE Closed Loop Power Control in Indoor Small Cells Faris B. Mismar and Brian L. Evans.

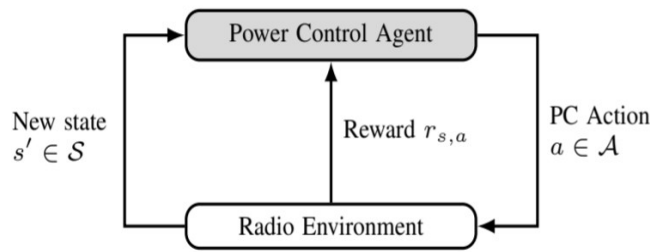
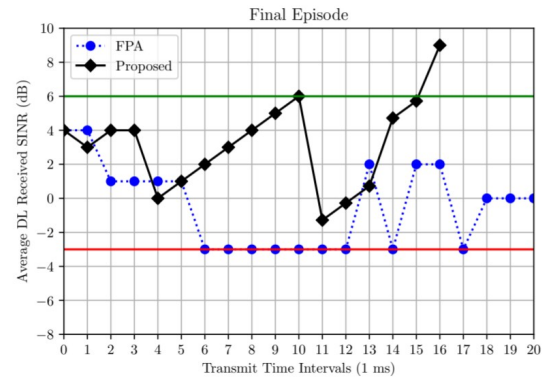


Fig. 3. Reinforcement learning elements.



O controle de potência (PC), é feito considerando os estados de entrada (S : estado atual e s' : novo estado), que são atualizados de acordo com o ambiente de rádio, que gera uma recompensa ($r_{s,a}$). O resultado ao empregar tal método pode ser observado na figura da direita, acima.

b) Mitigação de colisões em redes sem fio e móveis

Método utilizado: K-means

Foram implantados diversos sensores, que cobrem a faixa de frequência de 70MHz a 6GHz. O gerenciamento e armazenamento dos dados gerados pelos sensores é feito na nuvem. Os sensores implantados, realizam a medição do nível de energia para as respectivas bandas. O nível de energia medido, se acima de um limiar pré-determinado, indica um canal ocupado. Os dados coletados, são então armazenados e processados em conjunto com uma base legada, de forma que seja fornecida a taxa de ocupação para determinado canal por determinado período (1 hora no experimento). Em seguida, a base de dados é caracterizada/rotulada, e são indicados os canais candidatos e os de treinamento. Os canais candidatos e de treinamento, formam o conjunto de treinamento (rotulado). Um preditor que utiliza K-Means, é que realiza a atribuição de canal, bem como a de suas informações de compartilhamento. Por fim, os canais candidatos são ranqueados, para futura atribuição.

c) Otimização de esquema de moduação e codificação

Métodos utilizados: Redes Neurais Convolutivas e Redes Neurais Recorrentes

Artigo lido: Deep Learning Aided Method for Automatic Modulation Recognition CHENG YANG¹ , ZHIMIN HE² , YANG PENG² , YU WANG² , and JIE YANG²

Estrutura do sistema utilizado:

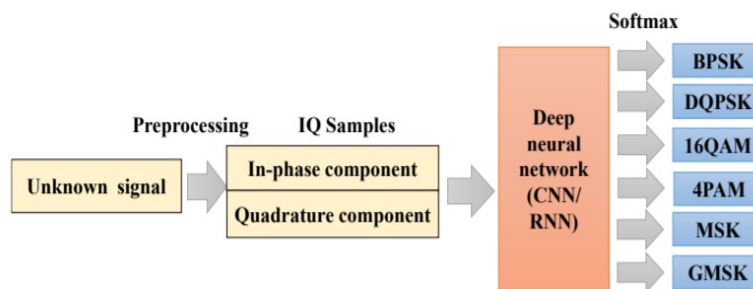


FIGURE 2. The structure of system model.

De um sinal desconhecido, são coletadas amostras de fase e quadratura. Estas amostras, então, são fornecidas as redes neurais, primeiro a uma rede neural convolutiva (CNN) e posteriormente a uma rede neural recorrente (RNN). Por fim, utilizando a ferramenta computacional Softmax, é realizada a estimativa da modulação utilizada.

Estrutura da rede neural convolutiva (CNN):

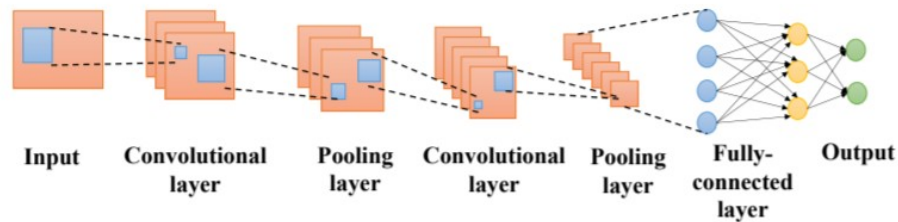


FIGURE 3. The structure of CNN.

Convolutional Layer: realiza a extração das características dos sinais de entrada (quadratura e fase), através da utilização de um núcleo convolucional.

Pooling Layer: realiza a compressão e simplificação das características recebidas da convolutional layer. Esta camada tem o objetivo de reduzir a complexidade computacional.

Fully connected layer: é a camada que realiza a combinação de todas as características de entrada.

Estrutura da rede neural recorrente (RNN):

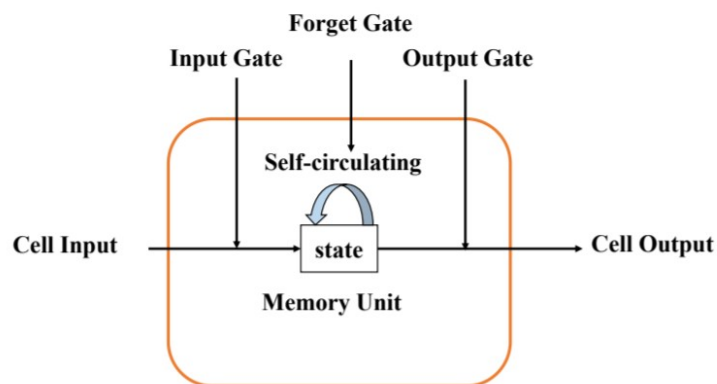


FIGURE 4. The structure of LSTM memory unit.

No artigo estudado, foi utilizada a Long-Short Term Memory (LSTM), que é uma das variações de RNN. Importante mencionar que a RNN, é uma rede neural com maior foco em “feedback”. Os elementos presentes na figura acima, são descritos abaixo.

Input gate: define quais as novas informações que podem entrar na unidade de memória.

Forget gate: define quando a informação do estado anterior, deve ser mantida ou apagada da memória.

Output gate: define quais informações podem ser fornecidas como saída.

d) Sensoriamento espectral

Método utilizado: Rede neural convolucional

Artigo lido: A Cognitive Radio Spectrum Sensing Method for an OFDM Signal Based on Deep Learning and Cycle Spectrum
Guangliang Pan , 1 Jun Li , 2 and Fei Lin2

Sistema utilizado no artigo:

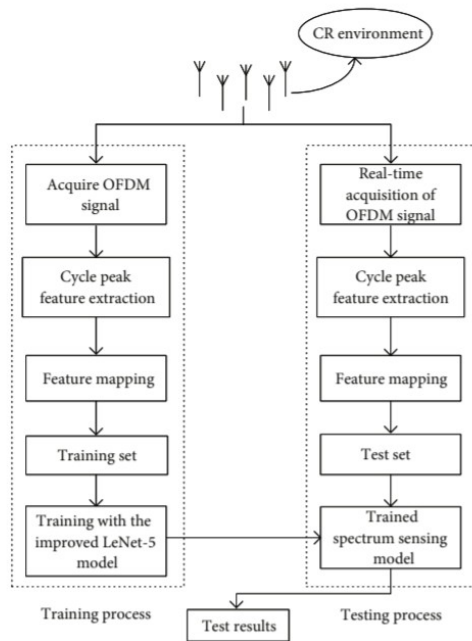
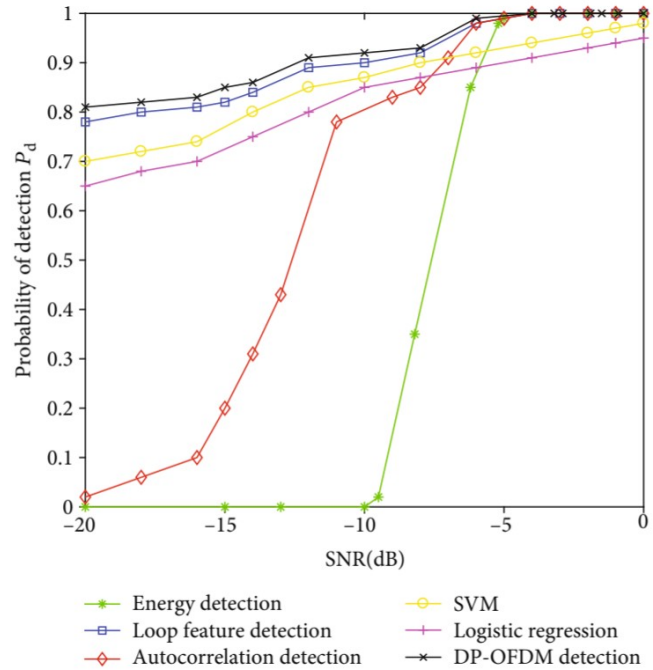


FIGURE 6: The framework of spectrum sensing.



O modelo em questão realiza o sensoriamento espectral, e realiza a conversão das informações colhidas no domínio da frequência, em um problema de imagem, a ser tratado pela rede neural convolutiva (CNN).

Ao tratar o sinal OFDM como uma imagem, houve aumento na precisão na detecção de tais sinais. O resultado pode ser observado na figura da direita, acima.

e) Posicionamento e localização em ambientes indoor

Método utilizado: DeepFi, Deep Learning que faz uso de Channel State Information (CSI), a fim de fornecer o fingerprint do ambiente e localização de User Equipment (UE).

Arquitetura DeepFi:

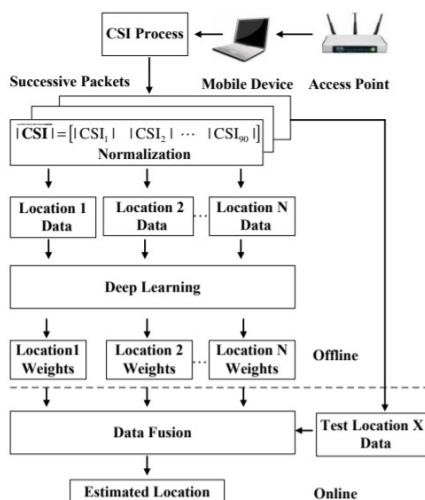
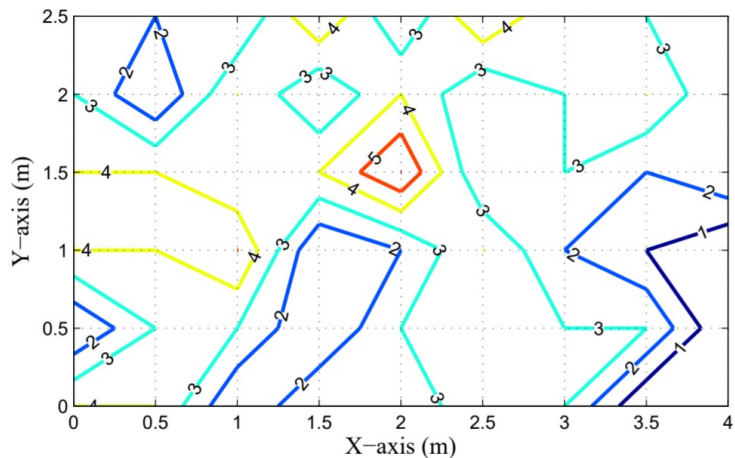


Fig. 5. The DeepFi architecture.



CSI process: Corresponde ao processo de recebimento das informações de estado do canal.

Normalization: As amplitudes dos valores de CSI são normalizadas, de modo que assumam valores entre 0 e 1.

Location data: Corresponde as amplitudes dos valores de CSI normalizadas, para cada uma das N antenas utilizadas (3 antenas, no experimento).

Deep learning: Esta camada desconstrói e reconstrói os dados recebidos. A finalidade deste processo, é a de encontrar os location weights otimizados, a fim de reduzir o erro.

Location weights: São os pesos otimizados. Serão utilizados pelo bloco seguinte.

Data fusion: Os pesos otimizados, são comparados as informações de CSI recebidas em tempo real. Em seguida, os dados são combinados. A partir desta combinação, é fornecida a localização estimada do UE.

f) Roteamento de redes

Método utilizado: Deep Belief Architecture (DBA)

Artigo lido: Machine Learning for Networking: Workflow, Advances and Opportunities, Mowei Wang, Yong Cui, Xin Wang, Shihan Xiao, and Junchen Jiang

Padrões de tráfego são taggeados, a partir das rodas utilizadas pelo protocolo OSPF (treinamento offline). Ao mesmo tempo, o tráfego em tempo real é também analisado (treinamento online). Em seguida, os dados de treinamento são utilizados no aprendizado da DBA, fazendo uso de um algoritmo de back propagation.

Como resultado, é fornecida a otimização de rotas, sendo assim, possível a identificação dos próximos nós da rede.

g) Detecção e estimação de canal em sistemas ópticos

Método utilizado: Support vector machine (SVM)

Artigo lido: A Survey on Machine Learning for Optical Communication [Machine Learning View] M. A. Amirabadi

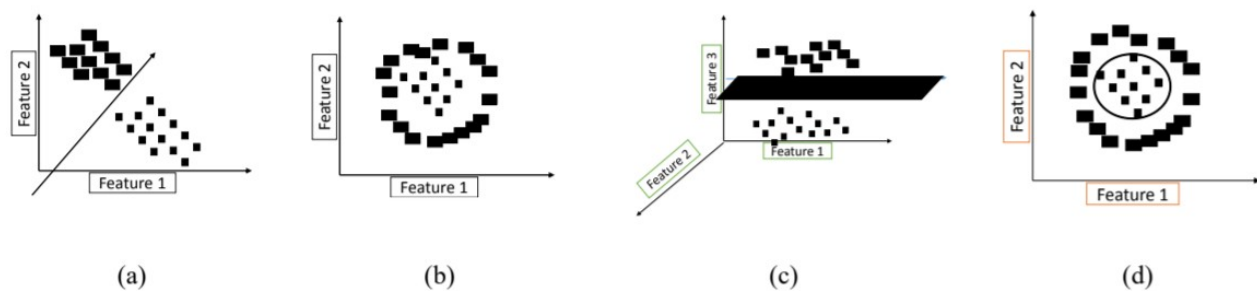


Fig.1.a. SVM, b. nonlinear data structure, c. dimension insertion, d. Kernelized SVM.

Um dos exemplos de aplicação de machine learning (ML) abordados no artigo lido, diz respeito a distinção de bits recebidos, que podem ser 0s ou 1s. Para realizar tal distinção, o algoritmo de ML coleta diversos dados, e aprende a classificá-los/separados, de acordo com as features coletadas dos dados obtidos anteriormente. Este modo de distinção de bits apresenta ganho de desempenho e redução de complexidade computacional, pois não utiliza os dados transmitidos como um todo, mas apenas as features extraídas destes (reiterando). A figura acima, ilustra algumas variações da aplicação de SVM na distinção de tais bits.

i) Segurança e robustez em redes de telecomunicações

Método utilizado: Adversarial Deep Learning, FeedForward Neural Network (FNN)

Artigo lido: IoT Network Security from the Perspective of Adversarial Deep Learning Yalin E. Sagduyu*, Yi Shi*†, and Tugba Erpek*‡

No artigo lido, foi utilizado o método adversarial deep learning, para fazer com que dispositivos IoT pudessem driblar possíveis ataques sofridos através da interface aérea. No artigo em questão, ML foi utilizada para reduzir o impacto de três tipos de ataques: jamming, envenenamento de espectro e violação de prioridades de transmissão.

De forma sucinta, quando o dispositivo IoT está operando com alta confiabilidade, erros propositais são gerados pelo algoritmo de ML, de forma a driblar os ataques sofridos.