FACULDADE METROPOLITANA DO ESTADO DE SÃO PAULO - FAMEESP

PÓS-GRADUAÇÃO EM CIÊNCIA DE DADOS E BIG DATA

RELATÓRIO TÉCNICAS UTILIZADAS EM CIÊNCIA DE DADOS E BIG DATA.

ALUNA: Rosilene Lima Justo

Brasília – DF
28/10/2024

RELATÓRIO TÉCNICAS UTILIZADAS EM CIÊNCIA DE DADOS E BIG DATA.

Data de entrega do relatório: 28/10/2024

Responsável:
Relatório: Rosilene Lima Justo – Cientista de dados

Versão 1.0

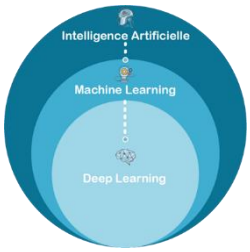
1. INTRODUÇÃO

Segundo as pesquisas realizadas, a ciência de dados faz parte de um contexto abrangente, cujo processo é composto pelas fases de coleta, armazenamento, análise, descarte, processamento e visualização. Existem técnicas específicas em ciência de dados que permitem o tratamento e a extração de informações. Nesse sentido, o aprendizado de máquina (*Machine Learning*) se refere a um contexto inteligente, capaz de obter, de forma automática, conhecimento a partir de dados. O intuito dessa técnica é simular o processo de aprendizagem humana e ter os mesmos *insights* que um humano teria, porém, trata-se de um conhecimento adquirido de forma artificial, por uma máquina.

A ciência de dados é composta por várias outras ciências, modelos, tecnologias, processos e procedimentos relacionados aos dados. Aprendizagem de máquina se diferencia da estatística tradicional, conhecida como "Estatística de Inferência", por ter como foco a predição e a busca por modelos práticos que podem ser utilizados para a forma de decisão. O aprendizado de máquina se realiza com base em um conjunto de dados usando a indução.

O aprendizado indutivo ocorre tomando-se por base o raciocínio e inferências sobre exemplos fornecidos ao sistema de aprendizado. Ele pode ser dividido em **supervisionado, não supervisionado e por reforço.**

Dentro do contexto do aprendizado de máquina, estão as "redes neurais" (*Deep Learning*), que se baseiam em neurônios artificiais, em referência aos biológicos. Técnicas em ascensão no contexto de ciência de dados e inteligência artificial estão ligadas ao uso do aprendizado de máquina e das redes neurais. São baseadas em conhecimento a partir de um conjunto de dados, que pode ser imagens, textos, arquivos, entre outros. Também pode ser entendido como modelos matemáticos que descreve o comportamento dos neurônios.



2. OBJETIVO GERAL

Criar uma rede neural que possa ser embarcado em um capacete de um astronauta, que conterá uma câmera inteligente para a identificação em tempo real e o envio de informações para a central em Bogotá na Colômbia.

3. OBJETIVOS ESPECÍFICOS

3.1. Identificar qual tipo de arquitetura de rede neural deve ser usada;

3.2. Definir como será dividido os dados para treinamento e teste;

3.3. Analisar se o treinamento de máquina será supervisionado e/ou não supervisionado.

4. OBJETIVOS SECUNDÁRIOS (NÃO OBRIGATÓRIO)

4.1. Conceitos básicos de aprendizado de máquina profundo e redes neurais convolucionais;

4.2. Conceitos de visão computacional, processamento de imagens e reconhecimento de padrões;

4.3. Conhecimento em desenvolvimento e implementação de sistema para identificação biométrica de indivíduos em tempo real, com ênfase no reconhecimento facial;

4.4. Conhecimento em alocação do sistema para funcionamento em uma arquitetura *IoT* (*Internet of Things*) usando o *Raspberry Pi;*

4.5. Conhecimento em treinamento e teste de acurácia.

5. FUNDAMENTAÇÃO TEÓRICA

5.1. Aprendizagem de máquina

É o processo pelo qual os parâmetros (os pesos das conexões entre os neurônios) e hiperparâmetros (taxas, topologia, etc...) de uma rede neural são ajustados através de uma forma continuada de estímulo pelo ambiente no qual a rede está operando.

5.2. Regras de aprendizagem

- Aprendizado por correção de erro;
- Aprendizado baseado em memória;
- Aprendizado Hebbiano;
- Aprendizado competitivo;
- Aprendizado Boltzman.

5.3. Aprendizado de máquina supervisionado

É um aprendizado indutivo e tem como característica o fornecimento ao algoritmo de aprendizado (ou indutor) um conjunto de dados que já possuem os seus rótulos de saída. Seu objetivo é prever uma variável dependente por meio de variáveis independentes. São indicados por **problemas de classificação ou regressão.**

- **Problemas de classificação**: As variáveis dependentes do problema são discretas. Ocorre quando precisamos prever o resultado em uma saída discreta.
- **Problemas de regressão**: As variáveis dependentes são contínuas. Ocorre quando precisamos prever o resultado de uma saída contínua.

Existem alguns algoritmos bastante utilizados para os problemas de classificação e regressão, são eles:

- Regressão linear – É uma análise que tem o intuito de gerar uma função linear para descrever a relação entre os dados, de forma que se possa estimar uma variável numérica por meio da função gerada.
- Regressão logística – É semelhante à linear, mas a variável estimada será categórica.
- SVM (*Support Vector Machine)* - É um vetor que utiliza o conceito de planos de decisão em um espaço multidimensional utilizando uma função *Kernel,* que é ajustada de forma a generalizar o modelo.
- KNN (*K-Nearest Neighbors)* – É um algoritmo simples que gera um modelo baseado nos dados e nos seus vizinhos.
- Árvores de decisão – São conjuntos de raízes e de nós que se organizam como um fluxograma de deliberações, a fim de se consolidar um modelo. Pode-se ter inúmeras árvores para um conjunto de dados.
- *Naive Bayes* – É um algoritmo de classificação que gera uma tabela de probabilidades a partir de uma técnica de classificação de dados.

5.4. Aprendizado de máquina não supervisionado

É um aprendizado indutivo e tem como característica um conjunto de dados não rotulados, e, portanto, o indutor não tem conhecimento inicial das classes envolvidas. Ou seja, é dado um conjunto de dados, não é previsto qual é a saída. Um tipo de abordagem utilizado é o de clusterização (agrupamento). Onde ocorre o agrupamento dos dados conforme características em comum.

5.5. Aprendizado de máquina por reforço e as cadeias de Markov

É um aprendizado indutivo e tem como característica um conjunto de dados, na qual a máquina aprende tomando decisões circunstanciais e aprendendo com elas.

- Se tomar uma decisão correta, recebe uma pontuação positiva (recompensa);
- Se tomar uma decisão errônea, recebe uma pontuação negativa (punição). Entendendo que aquela decisão foi errônea naquela circunstância.

A teoria matemática das cadeias de *Markov*, é comparado com o aprendizado por reforço. Essa teoria afirma que um caso particular do processo estocástico é caracterizado pelo fato de que o seu estado futuro depende apenas do seu estado atual, e não dos fatos passados.

5.6. Redes Neurais Artificiais (RNA) ou *Deep Learning*

É o subtópico dentro do aprendizado de máquina. *Deep learning* é uma ciência empírica. As redes neurais artificiais é um algoritmo e é pensado para imitar o cérebro humano. Os neurônios se propagam direcional, sempre para mesma direção. Ele passa no neurônio e é chamado de estímulo. Se o estímulo não for forte o suficiente para ativar o outro neurônio o sinal morre. Para os estudos é importante os seguintes segmentos:

- A intensidade do estímulo;
- O limiar (se o estímulo for maior tem-se uma ativação);
- Estado inativo (período refratário – o neurônio está descansando).

Integrando o segmento acima chega na função *sigmóide*, que em redes neurais artificiais é chamado de função de ativação. A preocupação é a intensidade do estímulo. De acordo com o estímulo que está retratado no eixo "x". A implementação do código é o valor de entrada. A partir desse valor de entrada, vai ter um valor de saída, que está representado no eixo vertical. Pode-se ter uma função degrau. Quando é feita ativação aplicado um peso nos valores que estão entrando o peso é multiplicativo. O funcionamento das RNAs é o seguinte:

- A informação ou solicitação é recebida na camada de entrada.
- A camada de entrada transforma a informação em um formato numérico que a máquina compreende.
- Os dados são transmitidos aos neurônios das camadas ocultas e processados.
- Os neurônios produzem um único valor, que depende dos coeficientes desenvolvidos durante o treinamento.
- Na última camada de saída, a rede neural tira uma conclusão e termina de processar a solicitação.

As redes neurais contêm quantidades de camadas e neurônios superior ao contexto de *Machine Learning.* A rede neural consegue aprender com o erro de saída da função de ativação, ajustando os pesos para que esse erro seja minimizado. A função de custo é a responsável por minimizar o erro da saída da rede neural. Processos importantes das redes neurais:

- Os dados de entradas x_1 e x_2 (teoria da *Perceptron)*;
- Os pesos w_1 e w_2;
- O parâmetro bias;
- A função de ativação $f(x)$; e
- A minimização do erro por uma função de custo.

Uma *Perceptron* pode ser conceituada em um modelo matemático que recebe várias entradas que serão ponderadas por pesos. É o início de uma rede neural. Os pesos ponderam esses dados de entrada de determinada forma. O parâmetro bias é o aprendizado inicial, isto é, o ponto de onde partimos. A função de ativação se dá quando tem a soma das entradas e são ponderadas por pesos, demonstrando se o neurônio será ativado ou não.

Ao observarmos a *perceptron* como o início de uma rede neural, temos os dados de entrada como um movimento de ida (*forward)* e um de volta (*backward)*, sobre os quais desejamos que a rede aprenda algo. Os pesos ponderam esses dados de entrada de determinada forma, os pesos são ajustados pela função de custo para que os erros sejam minimizados. O parâmetro bias inicia o aprendizado. Os dados são multiplicados pelos pesos, e os resultados serão somados. Há uma função de ativação, que dita se aquele resultado ativa ou não ativa o neurônio.

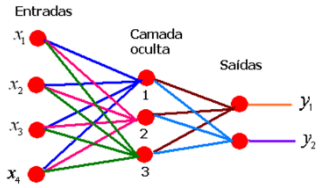
Assim, com base nela temos a saída da *perceptron.* Usando as redes neurais profundas a quantidade de camadas e neurônios são maiores possibilitando uma aprendizagem mais efetiva, de forma que a rede neural consiga aprender mais características e padrões daquele conjunto de dados. A primeira camada é a entrada, e a segunda é a saída, todas as camadas entre as duas são camadas ocultas. Entre cada camada, o sinal é propagado por meio de ativação. Esse método é pioneiro em diversas soluções das áreas de reconhecimento de fala, visão, computacional, carros autônomos e diversas outras. Costumeiramente, é feito a divisão dos dados de entrada para a aprendizagem, treino, teste e validação. Ao realizar o treinamento, o modelo pode ter um:

- Sobreajuste (*overfitting)* – o modelo memoriza os dados do conjunto e, portanto, não consegue predizer nada sobre novos dados que entrarem na rede.
- Subajuste (*underfitting)* – o modelo criado pela rede neural fica aquém do esperado e aprende muito pouco sobre os dados.
- Equilibrado (*balanced)* – Com base na etapa de treinamento, o modelo aprende sobre os dados e consegue inferir sobre os dados novos que possam ser apresentados à rede neural.

Os hiperparâmetros são parâmetros de redes neurais que devem ser definidos antes de treinar o modelo. São variáveis de configuração externas usadas para gerenciar o treinamento dos modelos de *Machine Learning.* Os hiperparâmetros controlam diretamente a estrutura, a função e a performance do modelo. Tem papel fundamental no treinamento de uma rede neural e na sua acurácia (nível de exatidão) para determinado problema. O ajuste pode ser manual ou automatizado. É interativo.

É pode ser feita diferentes combinações de parâmetros e valores. Começa definindo uma variável de destino como precisão, como á métrica principal e pretende maximizar ou minimizar essa variável. É recomendado usar técnicas de validação cruzada, para que seu modelo não fique centrado em uma única parte de seus dados. Existem dois tipos de redes neurais:

- Redes neurais rasas ou simples: costumam ter apenas uma camada oculta.
- Redes neurais profundas: têm várias camadas ocultas.



5.7. Tipos de Arquiteturas de Redes Neurais Artificiais (RNA)

5.7.1. Rede Multilayer Perceptron – MLP

O *Perceptron* é uma arquitetura simples. É a representação de um neurônio biológico em um modelo matemático que permite várias entradas. É um algoritmo simples destinado a realizar a saída em classificação binária, isto é, prevê se a entrada pertence a uma determinada categoria de interesse ou não; é um classificador linear, isto é, classifica a entrada separando duas categorias com uma linha reta. Não possui múltiplas camadas. A entrada geralmente é um vetor. É uma rede neural artificial composta por mais de um *Perceptron*. Esses são compostos por uma camada de entrada para receber o sinal, uma camada de saída que torna uma decisão ou previsão sobre a entrada, e entre esses dois, um número arbitrário de camadas ocultas que são o verdadeiro mecanismo computacional da MLP.

A MLP com uma camada oculta são capazes de aproximar qualquer função contínua. Aplicados a problemas supervisionados: treinam em um conjunto de pares entrada-saída e aprendem a modelar a correlação (ou dependência) entre essas entradas e saídas. O treinamento envolve o ajuste dos parâmetros, ou os pesos e bias, do modelo para minimizar o erro.

O *backpropagation* é usado para fazer os ajustes dos pesos e bias em relação ao erro, e o próprio erro pode ser medido de várias maneiras, inclusive pelo erro quadrático médio (*Mean Squared Erros* - MSE). É um treinamento supervisionado. Define a forma com que a rede é treinada.

5.7.2. Redes Neurais Convolucionais ou Convolutional Neural Networks - CNN

Reconhecer para dígitos manuscritos chamado *LeNet.* Redes *ConvNets* ou Redes CNNs são redes neurais profundas. As redes Convolucionais realizam o reconhecimento óptico de caracteres (OCR) para digitalizar texto e tornar possível o processamento de linguagem natural (LN) em documentos analógicos e manuscritos, onde as imagens são símbolos a serem transcritos. É um algoritmo de detecção facial.

CNNs podem ser aplicados a arquivos de áudio, quando estes são representados visualmente como um espectrograma. Tem eficácia nos avanços em visão computacional. Tem aplicações em:

- Carros autônomos;
- Robótica;
- Drones;
- Segurança;
- Diagnósticos médicos e tratamentos para deficientes visuais.

As redes convolucionais ingerem e processam imagens como tensores. Essas redes percebem imagens como volumes. Por meio das camadas convolucionais são extraídas todas as características de uma imagem.

A função de ativação *Rectied Linear Unit* – ReLU, possui a função de criar um limite para cada entrada após a camada de convolução.

A camada de *pooling* possui o objetivo de reduzir a dimensionalidade da entrada. Como consequência da redução, há uma minimização do tempo de treinamento da rede. Há dois tipos

de funções que podem ser realizadas na camada de *pooling,* a função max *pooling* (máximo) e *average pooling* (média).

A função max *pooling* utiliza o valor máximo do *kernel* de h x l dimensões e isso proporciona uma minimização dos ruídos contidos na imagem. Dessa forma, o valor máximo torna-se entrada da próxima camada. Já a função *average pooling* utiliza o valor médio para atribuir como entrada à próxima camada.

A camada de classificação ou também chamada de camada totalmente conectada, possui o objetivo de realizar a classificação da imagem com base nos processamentos realizados nas camadas anteriores. A classificação se dá com um novo dado que não está na base de conhecimento. É preciso buscar na memória os exemplos mais parecidos com o exemplo dado, a fim de rotular os novos dados. Os dados são classificados segundo os exemplos contidos na vizinhança que já está rotulados. Pode ser usado a regra do vizinho mais próximo.

5.7.3. Redes Neurais Recorrentes

São redes neurais artificiais para o processamento de dados sequenciais, como som, dados de séries temporais ou Linguagem Natural (LN). A rede recorrente que efetivamente associa memorias e entradas remota no tempo é chamada de memória de longo prazo (*Redes Long Shart Turn Memory* - LSTM).

As redes LSTM possuem aplicações práticas, incluindo processamento de LN, geração automática de texto e análise de séries temporais.

5.7.4. Redes feedforwards

Em uma rede feedforward, cada camada se conecta à próxima camada, porém não há caminho de volta.

5.7.5. Deep Belief Networks – DBNs

São usadas para reconhecer, agrupar e gerar imagens, sequências de vídeos e dados de captura de movimento. Também no processamento de LN.

Apresenta também as redes neurais *Deep Auto Encoder*; e as *Generative Adversarial Network* (GANs).

5.7.6. Haar Cascade

É um framework e tem como base três contribuições-chave introduzidas pelos autores (Viola & Jones, 2001). É uma representação de imagem criada para realizar cálculos eficientes. É um algoritmo de aprendizado baseado em *AdaBoost,* que seleciona um pequeno grupo de recursos críticos de um grupo maior. É um método de combinar classificadores cada vez mais complexos numa estrutura de cascata, aumentando a velocidade do detector focando somente nas regiões promissoras da imagem.

O processo de análise de um objeto utilizando o framework ocorre da seguinte forma: o algoritmo recebe uma imagem integral. Em seguida, serão aplicados retângulos (recursos) em uma ordem já previamente definida que, a partir da diferença de luminosidade, indicarão se aquele fragmento da imagem possui chances de ser uma face ou não. Conforme os recursos indicarem que há chance de o fragmento da imagem possuir um rosto presente, serão aplicados recursos cada vez mais complexos e específicos, aumentando a probabilidade de acerto. É um algoritmo de detecção facial.

5.7.7. Histograma de Gradientes Orientados – HGO ou Histogram of Oriented Gradient – HOG.

Desenvolvido por Dalal e Triggs, é um descritor de recursos cujo diferencial está em sua arquitetura, que foi criada para a identificação de pessoas. O seu funcionamento se dá através da distribuição (histograma) de gradientes orientados de uma imagem. É capaz de extrair bordas de um ou mais objetos, para depois enviá-las a um algoritmo classificador responsável por definir se na imagem analisada há ou não uma face. Quando os pixels de uma região apresentam uma mudança abrupta de cor, significa que aquele trecho analisado é a borda de um objeto.

Em termos matemáticos o gradiente é obtido através da derivada de uma função multivariável. Com esse gradiente foi criado um dos mais famosos descritores de recursos: o SIFT (Scale Invariant Feature Descriptor). Esse algoritmo utiliza gradientes orientados para extrair os pontos são chamados de pontos chave (Keypoints). É dividido os pixels de uma imagem em quadros e aplicaram um histograma de gradientes orientados em cada célula, distinguindo HGO dos demais descritores. É um algoritmo de detecção facial.

5.7.8. You Only Look Once – YOLO

É uma rede neural profunda, classificada como detectora de objetos. Sua principal função não é apenas reconhecer qual objeto está presente na imagem (como ocorre em algoritmos classificadores), mas também identificar em qual posição ele está localizando. Ele é capaz de responder duas perguntas: quais objetos estão na imagem? Onde eles estão posicionados?

A detecção é formada por dois processos: a classificação e localização.

Algoritmos classificadores são capazes de atribuir um rótulo a um conjunto de recursos extraídos da imagem. Já os localizadores distinguem o objeto do plano de fundo, e identificam as relações espaciais entre os objetos na imagem. Esses dois processos juntos permitem identificar vários objetos diferente em uma imagem. É um detector de objetos em tempo real. É um CNN open source, **chamada *Darknet.*** Suas primeiras camadas são responsáveis por extrair os recursos da imagem, enquanto as camadas totalmente conectadas preveem a probabilidade das classes e as coordenadas de saída.

5.7.9. DeepFaces

É um método utilizado para detecção e reconhecimento facial, que utiliza de uma tecnologia totalmente inovadora para fazer o reconhecimento facial em imagens, criado pela empresa facebook, o DeepFace utiliza redes neurais profundas para fazer as análises e alcançou uma precisão de ponta quando foi utilizado em um famoso teste de desempenho humano pela primeira vez.

5.7.10. Multitask Cascated Convolutional Networks – MTCNN

Um dos grandes desafios da identificação biométrica de indivíduos com uso de CNNs é a detecção de faces, a extração de características apropriadas e o reconhecimento da face, de forma robusta e invariante às alterações de pontos de vista, iluminação expressões faciais e obstruções. Contudo, por meio das redes neurais convolucionais em Cascata multitarefas é possível encontrar uma solução para a identificação biométrica facial de indivíduos de maneira eficaz. É organizada em 3 etapas:

- Primeira, utiliza com arquitetura a P-NET e possui o objetivo de prever posições da face e suas delimitações.

- Segunda, são utilizadas imagens para realizar as primeiras classificações, com isso há um refinamento dos resultados. A arquitetura que poderá ser usada é da R-NET, com o objetivo de remover os chamados não candidatos de faces.
- Terceira, é última, há um maior refinamento dos resultados e as marcações faciais. É utilizada a arquitetura O-NET, com o objetivo de marcar cinco posições faciais.

5.7.11. VGG

É uma arquitetura *deep* de uma rede neural convolucional que permite realizar o reconhecimento facial de indivíduos.

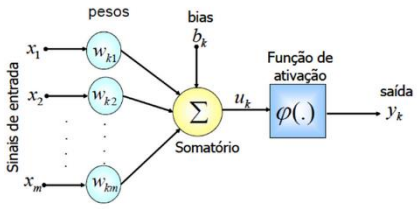
5.7.12. Local Binary Pattern (LBP)

Inicialmente desenvolvido para análise de texturas o LBP tem sido empregado, com sucesso, para extração de características no processo de reconhecimento e classificação de imagens de face. Isso ocorre pelo fato de que as faces podem ser vistas como uma composição de micro padrões que são bem descritos por esta técnica. Neste método a intensidade (v) de cada pixel de uma imagem é substituída por um vetor binário (b), determinado pela comparação entre a intensidade do pixel e as intensidades dos pixels vizinhos. Os valores obtidos para cada vizinho são concatenados e o número binário gerado é convertido na base decimal para substituir o valor central.

5.7.13. Autoencoders e redes adversárias generativas

Autoencoders é uma rede neural artificial de três camadas, isto é, uma rede neural com uma camada de entrada, uma oculta e uma camada de saída. Possui vários tipos: multicamada, convolucional e outras.

Redes adversárias generativas (GAN) é uma arquitetura de aprendizado profundo. Ela treina duas redes neurais para competirem entre si para gerar novos dados mais autênticos a partir de um determinado conjunto de dados de treinamento.



6. FERRAMENTAS

Banco de Dados - Big Data (grandes volumes de dados), Relacionais, NoSQL e outros;

IoT – Internet das coisas é uma rede de objetos e dispositivos que se conectam à internet e trocam dados.

Raspberry Pi – é um microcomputador do tamanho de um cartão de crédito.

Linguagem Python – bibliotecas (*Pandas, Numpy, Scikit-learn* (algoritmo *machine learning), Flask, Tensorflow, Keras, OpenCV*.

Tensorflow - é uma biblioteca de software de código aberto para computação numérica usando grafos computacionais.

Keras - é uma biblioteca de alto nível para deep learning, desenvolvida sobre Theano e Tensorflow. Ele é escrito em Python e fornece uma maneira limpa e conveniente de criar uma

variedade de modelos de deep learning.
Ambientes virtuais (virtualenv) – são ferramentas que permitem separar um projeto e suas dependências, como por exemplo, bibliotecas, gerenciadores de pacotes e versões dos interpretadores python em apenas um único lugar.
Git e Github – O Github é uma "rede social dev" em que é possível armazenar e compartilhar projetos de desenvolvimento de software. O Git é um sistema de controle de versão de arquivos; em outras palavras, é responsável por guardar o histórico de alterações sempre que alguém modificar algum arquivo que está sendo monitorado por ele.

7. DESAFIO PROPOSTO

Os buracos existentes na camada de ozônio podem ter efeitos nocivos em humanos, animais e plantas, pois perde-se a proteção contra os raios ultravioleta irradiados pelo Sol. Assim, a camada é o "protetor solar" de toda a Terra. Vai ser analisado os dados para identificar se há mais buracos menores na camada de ozônio. Para isso, foi encontrado um dataset, oriundo da Universidade de Harvard, que contém 510.566.963 imagens, sendo que 80% estão catalogadas. **Criar uma rede neural que possa ser embarcado em um capacete de um astronauta, que conterá uma câmera inteligente para a identificação em tempo real e o envio de informações para a central em Bogotá na Colômbia**.

- É necessário criar um banco de dados para grandes volumes de dados, indicado um Big Data;
- Deverá ser armazenado imagens rotuladas como positivas e negativas.
- Pré-processamento = Carregar e examinar os dados.
- Tokenização = É de costume utilizar tokenização para converter o texto em valores numéricos, atribuindo id as palavras.
- Modelagem = A parte da modelagem consiste em quatro camadas, entrada, incorporação, recorrentes, saída. A camada de incorporação nos permite capturar relações semânticas de palavras mais precisas. É na camada recorrente que é aplicado um vetor nas palavras atuais.

Responda as seguintes perguntas:

1 – Qual tipo de arquitetura de rede neural deverá ser usada?

Conforme a pesquisa realizada será usada a arquitetura de rede neural profunda. Ela pode analisar conjunto de dados não estruturados. Essa arquitetura possui muitas camadas e um grande números de parâmetros ajustáveis. Têm várias camadas ocultas com milhões de neurônios artificiais interligados. Será usada a rede neural convolucionais ou (CNN). As camadas ocultas nas redes neurais convolucionais executam funções matemáticas específicas, como resumo ou filtragem, chamadas convoluções. São úteis para a classificação das imagens, pois podem extrair recursos relevantes das imagens, são úteis para o reconhecimento e classificação das imagens. Cada camada oculta extrai e processa recursos de imagens diferentes, como bordas, cores e profundidade.

2 – Como deverá ser dividido os dados para treinamento e teste?

Usualmente a prática comum é usar uma proporção de 70% - 30% ou 80% -20% para a divisão dos dados entre treinamento e teste.

Os dados de treinamento serão utilizados para verificar seu desempenho sob condições reais de utilização. Ele treinará quantas vezes for definido (chamadas épocas). É definido

usualmente 10 épocas. O modelo vai aprender e também cometer erros (previsões erradas). Para cada erro que o modelo comete, há um custo e isso é representado no valor da perda *(loss)* para cada época. Deverá ter o mínimo de erro no final da última época. Após o treinamento, o modelo é avaliado com o conjunto de validação para verificar se ele é capaz de generalizar adequadamente para novos dado que não foram vistos durante o treinamento. Se o desempenho do modelo no conjunto de dados de validação não for satisfatório, pode ser necessário ajustar os hiperparâmetros do modelo, sendo os parâmetros que controlam o processo de treinamento. A etapa de treino e validação é um processo iterativo que pode ser repetido várias vezes até que se alcance um modelo satisfatório. É importante lembrar que o objetivo final do modelo de machine learning é ser capaz de generalizar para novos dados e, portanto, a avaliação do seu desempenho em um conjunto de dados de validação é fundamental para verificar se o modelo atende a esse objetivo.

Já os dados de teste é uma das etapas finais do processo de treinamento de um modelo. Nessa etapa, o modelo treinado é avaliado em um conjunto de dados de teste que não foi usado durante o treinamento. O objetivo principal dessa etapa é avaliar a capacidade do modelo de generalizar para novos dados que não foram vistos durante o treinamento. Isso é importante porque o objetivo final do modelo é ser capaz de fazer previsões precisas em dados que nunca viu antes. Durante a etapa de teste, o conjunto de dados de teste é passado pelo modelo treinado e as previsões do modelo são comparadas com as respostas verdadeiras do conjunto de dados de teste. A acurácia é um dos cálculos com base nessas comparações. Se o desempenho do modelo na etapa de teste for satisfatório, ele pode ser considerado para uso em produção. Caso contrário, pode ser necessário ajustar os hiperparâmetros do modelo ou revisar o conjunto de dados de treinamento. É importante destacar que a etapa de teste deve ser realizada em um conjunto de dados completamente separado do conjunto de dados de treinamento e validação, para evitar o overfitting e avaliar de forma justa o desempenho do modelo.

A etapa de inferência é basicamente a etapa de teste, porém em produção. Nesta etapa, o modelo treinado é utilizado para fazer previsões em novos dados e como os dados são novos, os dados devem passar pela etapa de pré-processamento e, se necessário, pela etapa de validação de dados. Essas etapas são necessárias para garantir que os dados estejam na forma esperada pelo modelo. Com os dados formatados conforme o modelo, o modelo é utilizado para fazer previsões e as previsões são enviadas para o usuário final, para um banco de dados ou algum outro local para ser armazenado.

3 – O treinamento do aprendizado de máquina será supervisionado ou não supervisionado?

Será usado o aprendizado supervisionado, por conter algumas imagens catalogadas. O aprendizado supervisionado usa um conjunto de treinamento para ensinar os modelos a produzir o resultado desejado. Esse conjunto de dados de treinamento inclui entradas e saídas corretas, o que permite ao modelo aprender com o tempo. O modelo possui uma referência daquilo que está certo e daquilo que está errado.