



APRENDIZADO EM TEMPO REAL PARA PREVISÃO DE RISCO DE *FLASHOVER* EM CADEIAS DE ISOLADORES

Trabalho de Conclusão de Curso

Engenharia da Computação

Felipe Rocha Bruce

Orientador: Prof. Dr. Sérgio Campello Oliveira

Coorientador: Prof. Dr. Byron Leite Dantas Bezerra



UNIVERSIDADE
DE PERNAMBUCO

**Universidade de Pernambuco
Escola Politécnica de Pernambuco
Graduação em Engenharia de Computação**

FELIPE ROCHA BRUCE

**APRENDIZADO EM TEMPO REAL
PARA PREVISÃO DE RISCO DE
FLASHOVER EM CADEIAS DE
ISOLADORES**

Monografia apresentada como requisito parcial para obtenção do diploma de Bacharel em Engenharia de Computação pela Escola Politécnica de Pernambuco – Universidade de Pernambuco.

Recife, Maio de 2022.

Bruce, Felipe Rocha

APRENDIZADO EM TEMPO REAL PARA PREVISÃO DE RISCO DE FLASHOVER EM CADEIAS DE ISOLADORES / Felipe Rocha Bruce. - Recife - PE, 2022.

xv, 49 f. : 31 il. ; 29 cm.

Trabalho de Conclusão de Curso (Graduação em Engenharia de Computação) Universidade de Pernambuco, Escola Politécnica de Pernambuco, Recife, 2022.

Orientador: Prof. Dr. Sérgio Campello Oliveira.

Coorientador: Prof. Dr. Byron Leite Dantas Bezerra.

Inclui referências.

1. Previsão de risco de *flashover*. 2. Aprendizagem de máquina. 3. Treinamento incremental. I. APRENDIZADO EM TEMPO REAL PARA PREVISÃO DE RISCO DE FLASHOVER EM CADEIAS DE ISOLADORES. II. Oliveira, Sérgio Campello. III. Universidade de Pernambuco.

MONOGRAFIA DE FINAL DE CURSO

Avaliação Final (para o presidente da banca)*

No dia 24/5/2022, às 11h00min, reuniu-se para deliberar sobre a defesa da monografia de conclusão de curso do(a) discente **FELIPE ROCHA BRUCE**, orientado(a) pelo(a) professor(a) **SÉRGIO CAMPELLO OLIVEIRA**, sob título Aprendizado em tempo real para previsão de risco de flashover em cadeias de isoladores, a banca composta pelos professores:

BRUNO JOSÉ TORRES FERNANDES (PRESIDENTE)

SÉRGIO CAMPELLO OLIVEIRA (ORIENTADOR)

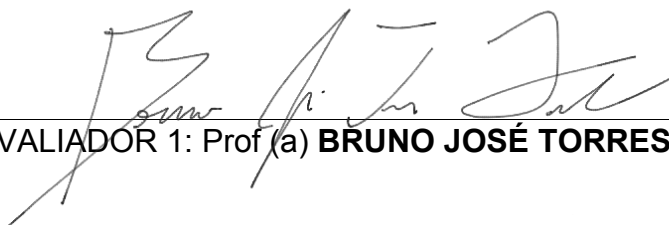
Após a apresentação da monografia e discussão entre os membros da Banca, a mesma foi considerada:

Aprovada Aprovada com Restrições* Reprovada

e foi-lhe atribuída nota: 9,0 (nove)

*(Obrigatório o preenchimento do campo abaixo com comentários para o autor)

O(A) discente terá 07 dias para entrega da versão final da monografia a contar da data deste documento.


AVALIADOR 1: Prof (a) **BRUNO JOSÉ TORRES FERNANDES**


AVALIADOR 2: Prof (a) **SÉRGIO CAMPELLO OLIVEIRA**

AVALIADOR 3: Prof (a)

* Este documento deverá ser encadernado juntamente com a monografia em versão final.

AGRADECIMENTOS

Agradeço aos meus pais, Ricardo e Silvana, à família e à namorada, Ariane, por sempre me apoiarem e serem compreensíveis em relação a preocupações que eu tinha e ausências devido a objetivos acadêmicos.

Gostaria de agradecer também aos meus amigos que me deram suporte ao longo da trajetória universitária.

Além disso, agradeço aos professores que tive a oportunidade de aprender com seus conhecimentos transmitidos.

Em especial, quero agradecer ao meu orientador, Professor Sérgio Campello, e ao meu coorientador, Professor Byron Leite, não só pela confiança depositada em mim, mas pela cooperação, incentivo e por sempre serem solícitos mediante às minhas dúvidas e necessidades.

Por fim, deixo registrado o meu agradecimento à Instituição, por me acolher e ter oferecido uma educação de qualidade.

Autorização de publicação de PFC

Eu, **Felipe Rocha Bruce** autor(a) do projeto de final de curso intitulado: **Aprendizado em tempo real para previsão de risco de flashover em cadeias de isoladores**; autorizo a publicação de seu conteúdo na internet nos portais da Escola Politécnica de Pernambuco e Universidade de Pernambuco.

O conteúdo do projeto de final de curso é de responsabilidade do autor.



Felipe Rocha Bruce



Orientador(a): **Sérgio Campello Oliveira**



Coorientador(a): **Byron Leite Dantas Bezerra**



Prof, de TCC: **Daniel Augusto Ribeiro Chaves**

Data: 24/5/2022

RESUMO

As cadeias de isoladores presentes em linhas de transmissão de alta tensão estão propensas a poluição e intempéries. Essa combinação cria uma camada condutiva, e propicia o surgimento do fenômeno *flashover*, que consiste em uma descarga parcial nos isoladores. Neste sentido, o presente trabalho teve como objetivo utilizar uma aprendizagem supervisionada incremental para prever o risco de *flashover* ao longo do tempo. Foram utilizadas redes neurais artificiais para realizar a tarefa de previsão, onde cenários de execuções foram definidos de modo a avaliar o desempenho do modelo. Foi possível obter resultados mais satisfatórios utilizando a arquitetura LSTM, ao realizar transferência de aprendizagem e utilizar informações relacionadas ao risco, temperatura, umidade relativa do ar e níveis de contagem de pulsos de 5 dias consecutivos para prever o risco 1 dia à frente. Após o treinamento incremental de cerca de 300 dias, utilizando uma base de dados localizada em Angelim, o modelo conseguiu prever de maneira aceitável o risco de ocorrência de *flashover* em tempo real.

Palavras-chave: *flashover*; previsão; risco; isoladores; aprendizado incremental; transferência de aprendizagem.

ABSTRACT

The chains of insulator strings present in transmission lines and high voltage are prone to pollution and inclement weather. This combination creates a conductive layer, and provides the appearance of the flashover phenomenon, which consists of a partial discharge on the insulators. In this sense, the present work aimed to use an incremental supervised learning to predict the risk of flashover over time. Artificial neural networks were used to perform the prediction task, where execution scenarios were defined in order to evaluate the performance of the model. It was possible to obtain more satisfactory results using the LSTM architecture, when performing transfer of learning and using information related to risk, temperature, relative humidity and pulse count levels of 5 consecutive days to predict the risk 1 day ahead. After incremental training of about 300 days, using a database located in Angelim, the model was able to acceptably predict the risk of flashover in real time.

Keywords: flashover; forecast; risk; insulators; incremental learning; transfer learning.

LISTA DE FIGURAS

Figura 1 - <i>Flashover</i> em isolador polimérico.....	14
Figura 2 - Arquitetura da comunicação via satélite.....	15
Figura 3 - Localização das torres com sistema de monitoramento de corrente de fuga.....	16
Figura 4 - Arquitetura de um <i>perceptron</i>	22
Figura 5 - Arquitetura MLP.....	22
Figura 6 - Célula LSTM.....	23
Figura 7 - Arquitetura do Sistema de Inferência Fuzzy.....	25
Figura 8 - Risco calculado pelo Sistema de Inferência <i>Fuzzy</i>	26
Figura 9 - Resultado da previsão do risco de <i>flashover</i>	27
Figura 10 - Janelas deslizantes.....	30
Figura 11 - Momento de retreinamento do modelo.....	32
Figura 12.a - Resultado da configuração 1 (MLP) para Angelim.....	35
Figura 12.b - Resultado da configuração 1 (MLP) para Angelim.....	35
Figura 13.a - Resultado da configuração 2 (MLP) para Angelim.....	36
Figura 13.b - Resultado da configuração 2 (MLP) para Angelim.....	36
Figura 14 - Resultado da configuração 3 (MLP) para Angelim.....	36
Figura 15.a - Resultado da configuração 1 (LSTM) para Angelim.....	37
Figura 15.b - Resultado da configuração 1 (LSTM) para Angelim.....	38
Figura 16.a - Resultado da configuração 2 (LSTM) para Angelim.....	38
Figura 16.b - Resultado da configuração 2 (LSTM) para Angelim.....	38
Figura 17 - Resultados sem treinamento incremental para o cenário 2 (MLP).....	39
Figura 18 - Resultados com treinamento incremental para o cenário 2 (MLP).....	40
Figura 19 - Resultados sem treinamento incremental para o cenário 2 (LSTM).....	41

Figura 20.a - Resultados com treinamento incremental para o cenário 2 (LSTM)...	41
Figura 20.b - Resultados com treinamento incremental para o cenário 2 (LSTM)...	41
Figura 21 - Resultados sem treinamento incremental para o cenário 3 (MLP).....	42
Figura 22.a - Resultados com treinamento incremental para o cenário 3 (MLP).....	43
Figura 22.b - Resultados com treinamento incremental para o cenário 3 (MLP).....	43
Figura 23 - Resultados sem treinamento incremental para o cenário 3 (LSTM).....	44
Figura 24.a - Resultados sem treinamento incremental para o cenário 3 (LSTM)...	44
Figura 24.b - Resultados com treinamento incremental para o cenário 3 (LSTM)...	44

LISTA DE TABELAS

Tabela 1 - Configurações do cenário 1 (MLP e LSTM).....	34
Tabela 2 - Métricas de avaliação do cenário 1 (MLP).....	34
Tabela 3 - Métricas de avaliação do cenário 1 (LSTM).....	37
Tabela 4 - Métricas de avaliação do cenário 2 (MLP).....	39
Tabela 5 - Métricas de avaliação do cenário 2 (LSTM).....	40
Tabela 6 - Métricas de avaliação do cenário 3 (LSTM).....	42

LISTA DE QUADROS

Quadro 1 - Descrição do critério de agrupamento por coluna.....	29
-----------------------------------------------------------------	----

LISTA DE SIGLAS

IA - Inteligência Artificial

LSTM - *Long Short-Term Memory*

MAE - *Mean Absolute Error* (Erro médio absoluto)

ML - *Machine Learning* (Aprendizagem de Máquina)

MLP - *Multi Layer Perceptron*

RMSE - *Root Mean Squared Error* (Raíz do erro médio quadrático)

RNA - Rede Neural Artificial

RNAs - Redes Neurais Artificiais

TL - *Transfer Learning* (Transferência de aprendizagem)

SUMÁRIO

1. INTRODUÇÃO.....	14
1.1 Objetivos.....	17
1.1.1 Objetivos gerais.....	17
1.1.2 Objetivos específicos.....	17
1.2 Contribuições alcançadas.....	18
1.3 Estrutura do trabalho.....	18
2. FUNDAMENTAÇÃO TEÓRICA.....	18
2.1 Lógica Fuzzy.....	19
2.2 Machine Learning.....	20
2.2.1 Aprendizado supervisionado.....	20
2.2 Redes Neurais Artificiais.....	21
2.2.1 <i>Perceptron</i>	21
2.2.2 <i>Multi Layer Perceptron</i>	22
2.2.3 <i>Long short-term memory</i>	23
2.3 Transfer learning.....	23
2.4 Treinamento incremental.....	24
3. TRABALHOS RELACIONADOS.....	25
3.1 Cálculo do risco de ocorrência da descarga parcial.....	25
3.2 Previsão do risco de <i>flashover</i>.....	26
4. MATERIAIS E MÉTODOS.....	28
4.1 Ambiente de execução e ferramentas utilizadas.....	28
4.2 Preparação dos dados.....	28
4.3 Janelas deslizantes.....	29
4.4 Cenários utilizados para o treinamento incremental.....	30
4.4.1 Cenário de execução 1.....	30
4.4.2 Cenário de execução 2.....	31
4.4.3 Cenário de execução 3.....	31
4.5 Atualização do modelo de previsão.....	31
4.6 Configurações e métricas de avaliação dos modelos.....	32
5. RESULTADOS E DISCUSSÕES.....	34
5.1 Resultados do cenário 1.....	34

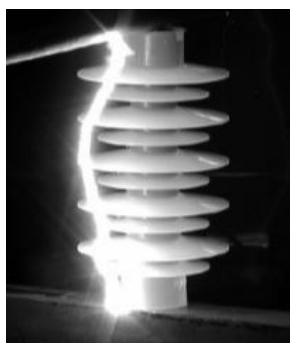
5.2	Resultados do cenário 2.....	39
5.3	Resultados do cenário 3.....	42
5.4	Considerações.....	44
6.	CONCLUSÃO E TRABALHOS FUTUROS.....	46
	REFERÊNCIAS.....	48

1. INTRODUÇÃO

Um isolante elétrico é um material no qual a corrente elétrica não flui livremente e são usados em equipamentos elétricos para suportar e separar condutores elétricos, sem permitir que a corrente passe por eles (BAYLISS; HARDY, 2007). Em se tratando de linhas de transmissão de energia de alta tensão, um isolador é um componente crucial, uma vez que é utilizado para evitar o fluxo indesejado de corrente para o solo a partir de seus pontos de suporte. Estes isoladores estão expostos a poluição e intempéries, e essa combinação de contaminantes, principalmente salinos, somados com uma alta umidade relativa do ar, criam uma camada condutiva, que origina uma corrente de fuga que flui pela superfície desses isoladores.

O aquecimento irregular causado pela corrente de fuga, por sua vez, cria bandas secas, provocando o acúmulo de cargas em suas extremidades, que culminam em quebrar a resistência dielétrica do ar, provocando um pequeno arco elétrico (HAMPTON, 1964). Caso nenhuma providência seja tomada para a remoção dos poluentes, a frequência e intensidade das descargas parciais evolui até a ocorrência de uma descarga completa, conhecida como *flashover* (FONTANA *et al.*, 2012), que inutiliza aquela cadeia, paralisando a linha de transmissão de energia e exigindo sua substituição. A Figura 1 ilustra o evento.

Figura 1 - *Flashover* em isolador polimérico

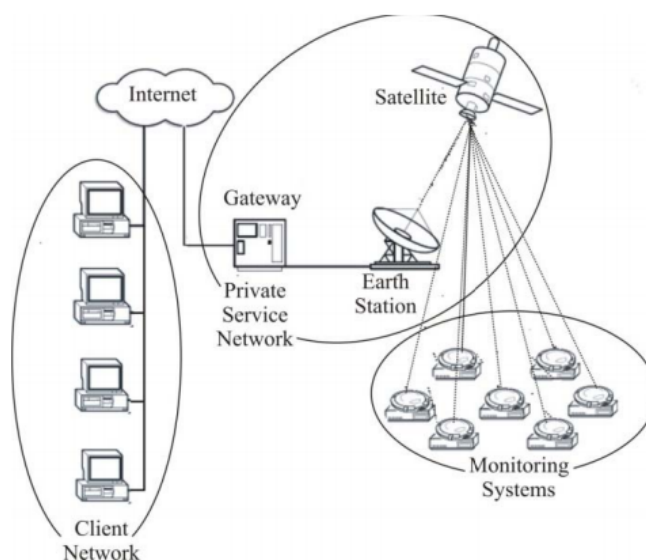


Fonte: *Contamination Flashover for porcelain and polymer insulator*, c2020.

A corrente de fuga contém informações que podem ser utilizadas para subsidiar manutenções corretivas para a remoção dos poluentes. Na Região Nordeste do Brasil, uma rede com 6 pontos de monitoramento em tempo real foi

construída como um experimento piloto para manutenção da rede de transmissão de energia (LIMA; OLIVEIRA; FONTANA, 2011a). A cada hora, sensores ópticos disponibilizavam informações sobre as ocorrências de pulsos curtos no sinal da corrente de fuga, relacionados às descargas parciais ocorridas nas cadeias de isoladores. Tendo operado por aproximadamente dois anos, a rede de sensores transmitiu, via satélite, os dados coletados, o que possibilitou a criação de uma base de dados rica em informações, contendo hora, umidade, temperatura e 4 contagens horárias de pulsos ocorridos com intensidades maiores que: 5, 10, 20 e 40 mA. A arquitetura da comunicação via satélite pode ser vista na Figura 2.

Figura 2 - Arquitetura da comunicação via satélite



Fonte: (FONTANA *et al*, 2012).

A localização dos pontos de monitoramento, citados no parágrafo anterior, pode ser visualizada na Figura 3.

Figura 3 - Localização das torres com sistema de monitoramento de corrente de fuga



Fonte: (LIMA; OLIVEIRA; FONTANA, 2011a).

A partir das informações disponíveis na base de dados, proveniente dos dados transmitidos pelos sensores (LIMA; OLIVEIRA; FONTANA, 2011a), e utilizando-se da Lógica *Fuzzy* (ZADEH, 1965), um classificador *fuzzy* de risco foi desenvolvido (LIMA; OLIVEIRA; FONTANA, 2011b), de modo a mensurar, numericamente, o risco de *flashover*. A informação do risco poderia ser utilizada para a geração automática de alertas de manutenção, dispensando a necessidade de checagem periódica por especialistas (LIMA; OLIVEIRA; FONTANA, 2011b).

Posteriormente ao classificador *fuzzy*, foi implementado um sistema de previsão do risco, utilizando uma rede neural artificial (RNA) (MCCULLOCH, W.; PITTS, W., 1990) de arquitetura *Multi Layer Perceptron* (MLP), que utilizava dados de sensores localizados a 100 km de distância do local alvo. As entradas dessa rede consistiam nas informações de risco, calculadas através do classificador *fuzzy* (LIMA; OLIVEIRA; FONTANA, 2011b), durante dois dias consecutivos. Na contribuição supracitada, foi realizada uma divisão na base dos dados para treino e validação da solução. Embora tenham sido obtidos resultados satisfatórios, a solução proposta não se aplica para ambientes mais afastados, com características climáticas diferentes e que possam exibir diferentes padrões de deposição e remoção natural de poluentes, ou mesmo locais que não dispõem de uma base de dados prévia.

Nesse sentido, uma alternativa seria realizar um treinamento incremental em tempo real da RNA, onde os modelos de previsões continuam a aprender com novos dados na medida em que são disponibilizados, sem a necessidade de executar todo o treinamento baseado no histórico de dados. Tal abordagem permitiria, por exemplo, uma rede independente e genérica, visto que nenhuma base de dados seria pré-requisito para um treinamento prévio. Além disso, não seria necessário atualizar o dispositivo que contém a rede neural treinada, pois a mesma iria se auto ajustar.

1.1 Objetivos

A seguir, serão apresentados os objetivos gerais e específicos.

1.1.1 Objetivos gerais

O presente trabalho busca aplicar um treinamento incremental em uma rede neural artificial para prever o risco de ocorrência de *flashover* em uma linha de transmissão de energia de alta tensão, de modo a mensurar em quanto tempo a rede começa a acertar as previsões.

1.1.2 Objetivos específicos

São definidos os seguintes objetivos secundários a fim de atingir o objetivo anterior:

- Revisar a literatura acerca do tema alvo;
- Realizar o tratamento da base de dados;
- Configurar um ambiente para executar o treinamento incremental da rede neural artificial;

- Utilizar RNAs, com treinamento incremental e transferência de aprendizagem, para prever o risco de *flashover* nas seguintes arquiteturas: MLP e *Long Short-Term Memory* (LSTM);
- Ajustar parâmetros das RNAs para obter melhores resultados.
- Analisar e comparar os resultados obtidos.

1.2 Contribuições alcançadas

Este trabalho atuou na importante investigação da previsão do risco de descargas parciais em linhas de transmissão de alta tensão, e propôs como contribuição inicial uma nova maneira de realizar tal tarefa.

Além disso, foi possível observar uma melhora nas métricas utilizadas para avaliar os modelos de previsões, quando comparado com o uso de modelos realizando previsões sem e com o treinamento incremental em tempo real para a previsão do risco de *flashover*.

Por fim, houve o enriquecimento da literatura acerca da temática, visto que foi utilizado um treinamento incremental para realizar a previsão, diferentemente de trabalhos anteriores.

1.3 Estrutura do trabalho

O presente documento está dividido em 6 capítulos no total, e os próximos capítulos estão elencados abaixo:

- **Capítulo 2 - Fundamentação teórica;**
- **Capítulo 3 - Trabalhos relacionados;**
- **Capítulo 4 - Materiais e Métodos;**
- **Capítulo 5 - Resultados e discussões;**
- **Capítulo 6 - Conclusão e trabalhos futuros.**

2. FUNDAMENTAÇÃO TEÓRICA

Este capítulo tem como objetivo apresentar conceitos importantes para a construção desta pesquisa, com a definição do que é Lógica *Fuzzy*, *Machine Learning*, Redes Neurais Artificiais (RNAs), *Transfer Learning* (TL) e o que vem a ser treinamento incremental.

2.1 Lógica *Fuzzy*

Criada por Lofti Zadeh (ZADEH, 1965), a lógica *fuzzy*, também conhecida como lógica nebulosa, trata-se de uma extensão da lógica booleana clássica, e é capaz de tratar incertezas da linguagem natural.

De acordo com Lima (2003), esse tipo de lógica permite lidar melhor com a representação das subjetividades humanas:

Um aspecto interessante da teoria nebulosa é a possibilidade de se incluir em um modelo matemático conceitos intuitivos que na maioria das vezes são altamente imprecisos e conseqüentemente de difícil tratamento. A capacidade de capturar com clareza e concisão as várias nuances dos conceitos psicológicos utilizados pelos seres humanos em seu raciocínio usual, sem necessidade de enquadrá-lo em estados nítidos torna a lógica nebulosa uma importante ferramenta na modelagem de sistemas imprecisos (LIMA, 2003, p. 83).

Por exemplo, uma expressão como “pouco nublado” pode ser convertida para um valor numérico e, posteriormente, ser facilmente manipulada por sistemas computacionais.

Klir G. J. (2015) exemplifica ao afirmar que o tempo está “ensolarado”, o qual não afirma a inexistência de nuvens, da mesma forma em que um tempo nublado pode haver sol. Como ensolarado, geralmente é aceito um céu que possui entre 20% a 30% de nuvens. Entretanto, os termos “ensolarado” e “nublado” trazem

consigo informações vagas e a representação destas informações se dá pelo uso de conjuntos *fuzzy*.

2.2 Machine Learning

Antes de definir propriamente o que é ML, é importante entender de que se trata de um subconjunto da Inteligência Artificial (IA), que é a ciência e engenharia que se preocupa com a compreensão computacional da inteligência, do comportamento e, portanto, da criação de máquinas inteligentes (RUSSELL; NORVIG, 2010).

Tom Mitchell define da seguinte maneira o que é ML:

Um programa de computador é dito para aprender com uma experiência E com uma relação de uma classe de tarefas T e medida de desempenho P, se o seu desempenho em tarefas em T, medida pelo P, melhora com a experiência E. (MITCHELL, 1997).

Nesse âmbito, existem alguns tipos de paradigmas de aprendizagem de máquina, tais quais: aprendizado supervisionado, aprendizado não supervisionado e aprendizado por reforço.

Porém, para atingir a compreensão de termos que virão a surgir no decorrer do trabalho, será necessário introduzir apenas o conceito de aprendizado supervisionado.

2.2.1 Aprendizado supervisionado

Aprendizado supervisionado consiste em um modelo que é apresentado para um conjunto de dados do tipo $D = \{x, y\}_{n=1}^N$, onde o par x e y correspondem a entrada do modelo e rótulo, respectivamente (KLIR e YUAN, 2015).

O valor de y pode assumir diferentes tipos, a depender do tipo de problema a ser resolvido. Por exemplo, caso seja um problema relacionado a classificação, geralmente y é uma representação escalar de um rótulo. Caso se trate de um problema de regressão, y pode assumir um vetor de valores contínuos (KLIR e YUAN, 2015). Desta maneira, o paradigma de aprendizagem supervisionada se fará presente no trabalho.

2.2 Redes Neurais Artificiais

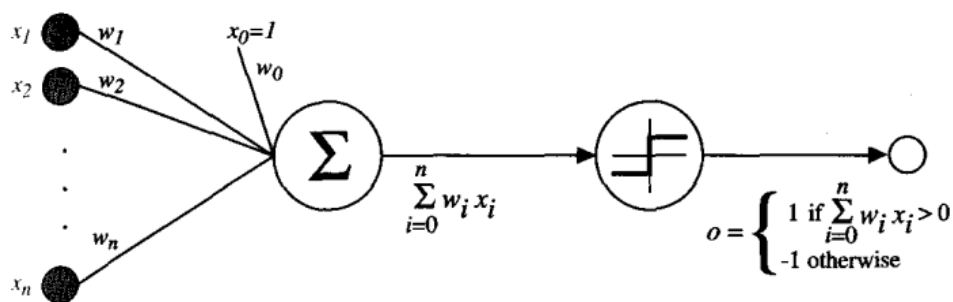
As pesquisas em RNAs têm sido influenciadas pela observação acerca dos sistemas de aprendizagem biológica, que são compostos por redes muito complexas de neurônios interconectados. Uma rede neural artificial consiste em um conjunto de unidades simples fortemente interconectadas, onde cada uma obtém uma entrada de valor real (que pode ser a saída de outras unidades) e produz uma única saída de valor real (que pode ser uma entrada para muitas outras unidades) (MITCHELL, 1997).

Os métodos de aprendizado de redes neurais fornecem uma abordagem robusta para aproximar funções de destino com valor real, valor discreto e valor vetorial. Para certos tipos de problemas, como aprender a interpretar dados complexos de sensores do mundo real, as redes neurais estão entre os métodos de aprendizado mais eficazes atualmente conhecidos (MITCHELL, 1997).

2.2.1 Perceptron

O *perceptron* é o tipo mais simples de uma RNA. Ela consiste em um vetor de valores reais, que atuam como entradas da rede e são utilizadas para o cálculo de uma combinação linear delas, resultando na ativação, ou não, do neurônio. A Figura 4 ilustra esse tipo de RNA.

Figura 4 - Arquitetura de um *perceptron*



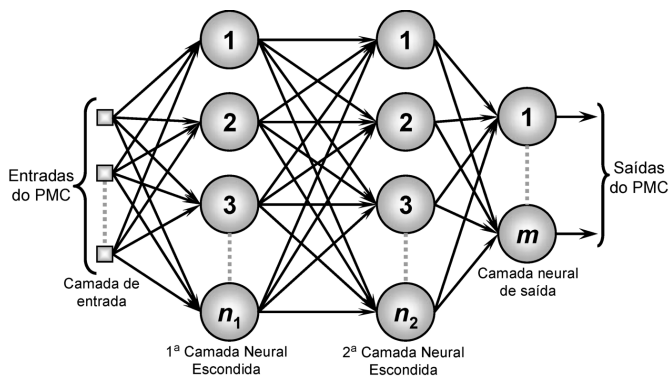
Fonte: (MITCHELL, 1997).

2.2.2 Multi Layer Perceptron

O *Perceptron* Multicamadas, do Inglês *Multi Layer Perceptron*, consiste em um sistema de neurônios simples interconectados (ou nós), que é um modelo que representa um mapeamento não linear entre um vetor de entrada e um vetor de saída. Os nós são conectados por pesos e sinais de saída, que são uma função da soma das entradas para o nó modificado por uma simples função de transferência não linear de ativação.

A superposição de muitas funções de transferência não lineares simples permite que a arquitetura MLP aproxime funções extremamente não lineares (MITCHELL, 1997). A Figura 5 representa uma arquitetura de uma RNA utilizando MLP.

Figura 5 - Arquitetura MLP



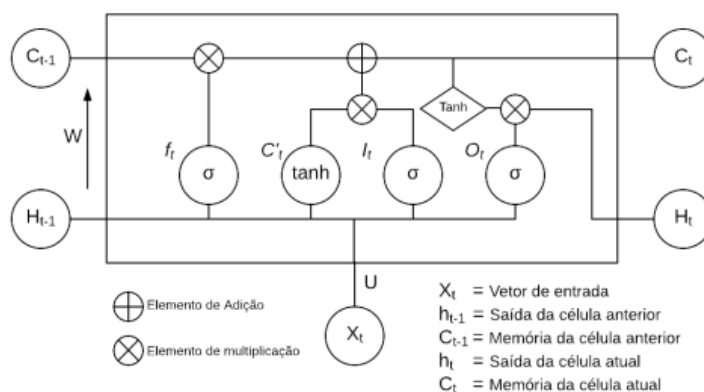
Fonte: (MOREIRA, 2018).

2.2.3 Long short-term memory

Antes de entender o que vem a ser LSTM, é necessário definir o que é uma Rede Neural Recorrente, pois LSTM consiste em uma rede recorrente. Segundo Yu *et al* (2019), “A característica típica da arquitetura Rede Neural Recorrente é uma conexão cíclica, que permite que a mesma possua a capacidade de atualizar o estado atual com base em estados passados e dados de entrada atuais”. Dessa maneira, uma rede recorrente possibilita a persistência de informações.

LSTM é uma arquitetura composta por unidades, conhecidas como “célula”, e tem se mostrado um tipo de arquitetura adequada para classificação, processamento e previsão de séries temporais com intervalos de tempo de duração não conhecidos. A arquitetura de uma célula LSTM pode ser visualizada na Figura 6.

Figura 6 - Célula LSTM



Fonte: (MELO, 2018).

Na Figura 6, W e U constituem vetores de pesos utilizados em f_t , C'_t , I_t e O_t . Por realizar um controle de fluxo de informações, f_t , C'_t , I_t e O_t são chamadas de válvulas. Por fim, σ e \tanh são funções de ativação, onde σ é a *sigmoid*, com valores que vão de 0 a 1 e \tanh , ou tangente hiperbólica, possui valores de -1 a 1.

2.3 Transfer learning

Aprendizagem por transferência, do Inglês *transfer learning*, consiste no processo de realizar o aprimoramento de um modelo para realizar uma determinada tarefa, a partir da transferência de conhecimento adquirido previamente em uma tarefa similar (MOHANASUNDARAM *et al.*, 2019). Esse processo tem como objetivo

reduzir o tempo gasto com o treino de modelos criados do zero (do princípio), de modo que tenta suprir a falta de grandes conjuntos de dados para o treinamento dos modelos posteriores, ao realizar o reaproveitamento do conhecimento de modelos previamente treinados (MENEGOLA *et al.*, 2017).

Dessa forma, TL também possibilita o cenário de comparação entre modelos treinados previamente e ajustados em uma nova base de dados ou treinandos integralmente utilizando apenas conjunto de dados alvo.

2.4 Treinamento incremental

O aprendizado incremental é um paradigma de ML, em que o processo de aprendizado ocorre quando novos exemplos surgem, ajustando o que foi aprendido de acordo com os novos exemplos. A diferença entre o aprendizado incremental do aprendizado de máquina tradicional é que ele não pressupõe um conjunto de dados de treinamento suficiente antes do processo de aprendizado, mas os exemplos de treinamento surgem ao longo do tempo (R.R; P. R, 2013).

Dessa maneira, em ambientes onde dados são recebidos constantemente, modelos previamente treinados com treinamento tradicional podem não estar preparados para lidar com mudanças repentinas, relacionadas à tarefa que lhe foi atribuída originalmente. Em contrapartida, realizar um treinamento incremental faz com que o modelo se torne mais flexível e apto a lidar com novas variações dos dados, assim como é uma técnica que resulta em menos intervenção humana no processo de atualização do modelo, já que ele está em treinamento constante. Além disso, como o treino ocorre à proporção que novas informações são disponibilizadas para o modelo, não é necessário ter uma base de dados prévia disponível.

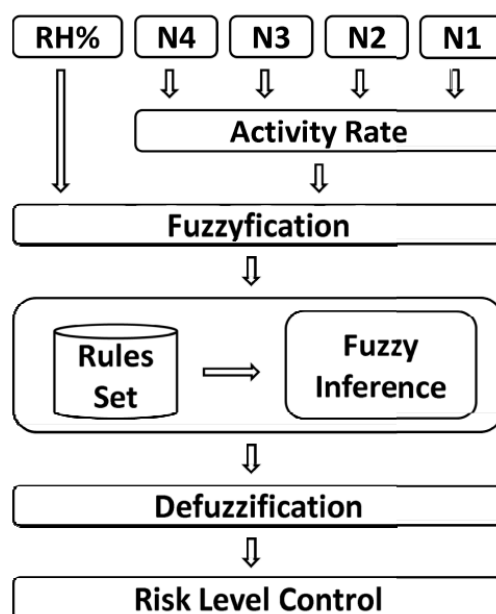
3. TRABALHOS RELACIONADOS

3.1 Cálculo do risco de ocorrência da descarga parcial

Lima, Oliveira e Fontana, (2011b), desenvolveram um Sistema de Inferência *Fuzzy* para realizar o cálculo do risco de ocorrência da descarga parcial em linhas de alta tensão. Como mencionado na Introdução, os sistemas de sensores classificam as descargas parciais em quatro faixas, de N1 a N4. Tais faixas são combinadas de modo a gerar a taxa de atividade.

A Figura 7 consiste no diagrama esquemático do Sistema de Inferência *Fuzzy* desenvolvido por Lima, Oliveira e Fontana, (2011b). O sistema espera como entrada a combinação das contagens de N1 a N4, juntamente com a umidade relativa. A saída do sistema trata-se de um valor numérico que incorpora todas as regras e todos os testes declarados como expressões de linguagem natural.

Figura 7 - Arquitetura do Sistema de Inferência *Fuzzy*



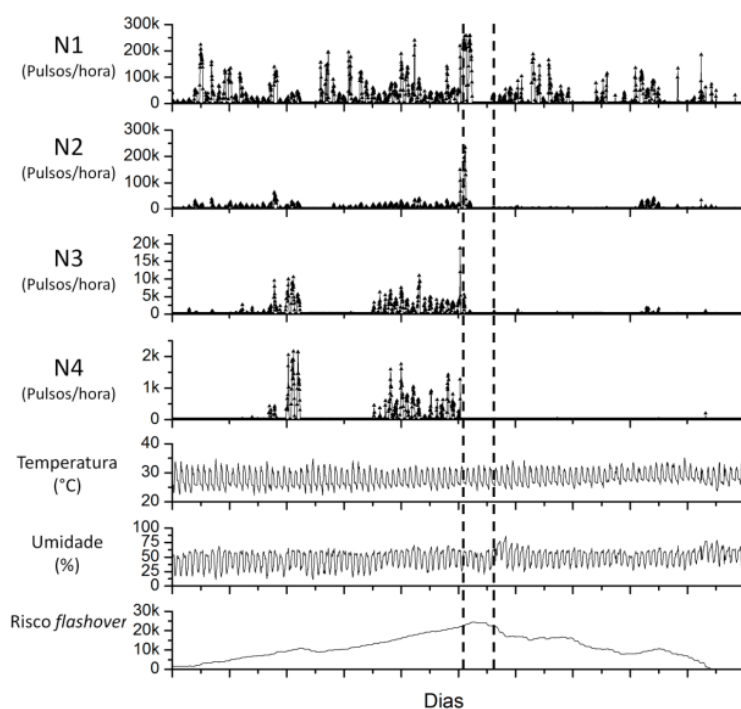
Fonte: (LIMA; OLIVEIRA; FONTANA., 2011b).

A saída gerada pelo sistema é então utilizada para o cálculo do risco, que consiste na fórmula a seguir:

$$R_{k+1} = R_k + R_{k-1} \quad (1)$$

Em (1), R_{k+1} é o próximo nível de risco, R_k é o nível de risco atual, R_{k-1} é o último nível e R_0 é zero. De acordo com Lima, Oliveira e Fontana, (2011b), o risco consiste em função incremental e R_k pode assumir um valor positivo ou mesmo negativo, a depender da combinação obtida entre o nível de atividade e umidade relativa. A Figura 8 contém os gráficos relacionados aos níveis de atividade, temperatura e umidade relativa registrados, com o respectivo risco calculado.

Figura 8 - Risco calculado pelo Sistema de Inferência Fuzzy



Fonte: Adaptado (SILVA, 2013).

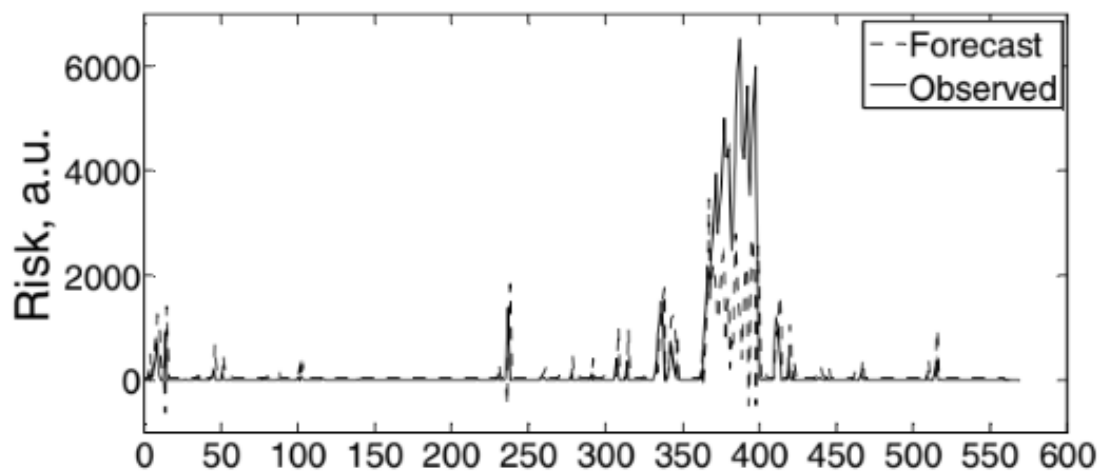
3.2 Previsão do risco de *flashover*

Uma rede neural MLP foi desenvolvida por Lima, Oliveira e Fontana, (2011a), com o objetivo de prever o risco de *flashover* em isoladores de redes de transmissão de alta tensão. Foi realizado o treinamento da MLP com dados referentes à uma base de treinamento localizada a 100 Km de distância da base de dados alvo.

A informação do risco foi obtida utilizando o Sistema de Inferência Fuzzy citado anteriormente, e utilizando a informação de dois dias seguidos para prever o risco dez dias à frente, não foram observados resultados falso-positivos nas

previsões, o que evitaria mobilizações desnecessárias para manutenção corretiva do isolador. A Figura 9 representa o gráfico do resultado obtido pela contribuição.

Figura 9 - Resultado da previsão do risco de *flashover*



Fonte: (LIMA; OLIVEIRA; FONTANA, 2011a)

4. MATERIAIS E MÉTODOS

O presente capítulo contém a descrição da metodologia utilizada na etapa de preparação e execução do trabalho. Em posse dos dados gerados pelos sensores de descarga parcial e do risco gerado pelo sistema de inferência *fuzzy* (LIMA; OLIVEIRA; FONTANA., 2011b), o problema consiste em utilizar tais dados como entrada de uma RNA para realizar a previsão do risco, utilizando o treinamento incremental.

4.1 Ambiente de execução e ferramentas utilizadas

Neste trabalho, foi utilizada a linguagem de programação Python, juntamente com o auxílio de bibliotecas voltadas para ML, como a biblioteca Keras (<https://keras.io/>). Para realizar a plotagem dos gráficos, foi utilizada a biblioteca Plotly (<https://plotly.com/>).

Com o intuito de montar um ambiente de execução que independe das configurações da máquina local e já possuir bibliotecas Python pré-instaladas, foi escolhida a plataforma em nuvem *Google Colaboratory* para realizar o tratamento dos dados, desenvolvimento e treinamento dos modelos de previsões.

4.2 Preparação dos dados

Inicialmente, foram selecionadas as bases de dados das seguintes regiões: Angelim, Coruripe, Fortaleza e São Miguel dos Campos, que se encontravam em planilhas no formato CSV. Os dados estão distribuídos por hora, e consistem nos valores registrados pelos sensores bem como o resultado do risco calculado pelo sistema de inferência *fuzzy*.

Em seguida, foi realizada uma análise dos dados para direcionar o pré-processamento deles, que resultou em uma observação de dados que não compunham o total de um mês contínuo, e sim a fração de alguns dias do mês analisado. Além disso, foi percebido uma grande discrepância entre os valores de cada coluna, que consistiam em: N1, N2, N3, N4, Umidade, Temperatura e Risco.

Após a fase da análise, o pré-processamento se inicia, com a remoção de colunas não relevantes e o agrupamento dos dados de acordo com a data em que foram registrados, de modo a ser gerado uma base com valores diários. O Quadro 1 ilustra as colunas que serão utilizadas, bem como o critério utilizado por cada agrupamento.

Quadro 1 - Descrição do critério de agrupamento por coluna

Nome da coluna	Descrição	Critério de agrupamento
N1	Contagem de pulsos ocorridos com intensidades maiores 5 mA	Soma
N2	Contagem de pulsos ocorridos com intensidades maiores 10 mA	Soma
N3	Contagem de pulsos ocorridos com intensidades maiores 20 mA	Soma
N4	Contagem de pulsos ocorridos com intensidades maiores 40 mA	Soma
Umidade	Valor da umidade relativa	Média aritmética
Temperatura	Valor da temperatura	Média aritmética
Risco	Valor do risco calculado pelo sistema de inferência	Média aritmética

Fonte: De autoria própria.

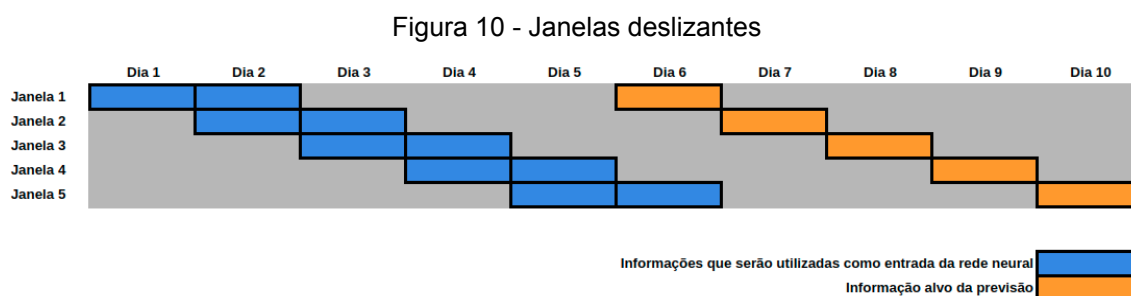
Com o objetivo de possuir uma base de dados com registros completos por mês, foram removidos os registros iniciais das bases de Angelim e São Miguel dos Campos. Por fim, os dados foram normalizados utilizando a normalização min-max.

4.3 Janelas deslizantes

Após o pré-processamento, os dados foram separados de maneira que fossem obtidas as entradas da RNA e o valor esperado como resultado da previsão. Para isso, foi desenvolvido um algoritmo de janelas deslizantes, onde se espera o número de dias para servir de entradas da rede e quantos dias à frente seria o alvo da previsão.

Por exemplo, se as entradas do modelo forem utilizar informações contidas em dois dias e o objetivo é realizar a previsão de 4 dias à frente, a janela terá um

tamanho igual a 6, e será deslizada desconsiderando o dia anterior. A Figura 10 ilustra tal configuração.



Fonte: De autoria própria.

4.4 Cenários utilizados para o treinamento incremental

A realização do treinamento incremental e posterior avaliação da previsão levou em consideração três cenários de execução.

4.4.1 Cenário de execução 1

Este cenário consistiu em um treinamento incremental de uma RNA não treinada previamente, apresentada para cada uma das bases de dados disponíveis. Foram utilizadas três configurações, com o critério de utilizar informações de uma quantidade maior de dias para prever o risco de um dia próximo, assim como utilizar uma quantidade menor de dias para prever o risco muitos dias à frente. As configurações encontram-se elencadas a seguir:

- Configuração 1 - Utilização de todas as informações presentes em 5 dias seguidos para prever o risco do dia seguinte;
- Configuração 2 - Utilização somente da informação do risco de 2 dias seguidos para prever 10 dias à frente;
- Configuração 3 - Utilização de todas as informações presentes em 2 dias consecutivos para prever 10 dias à frente;

4.4.2 Cenário de execução 2

Esse cenário consistiu em utilizar TL, com uma RNA treinada previamente em uma base de dados distante da base alvo. Nesse caso, foi utilizada a base de dados de Fortaleza como treino, e um posterior treinamento incremental nas bases de Angelim e São Miguel dos Campos. Todas as informações disponíveis foram utilizadas para realizar a previsão.

4.4.3 Cenário de execução 3

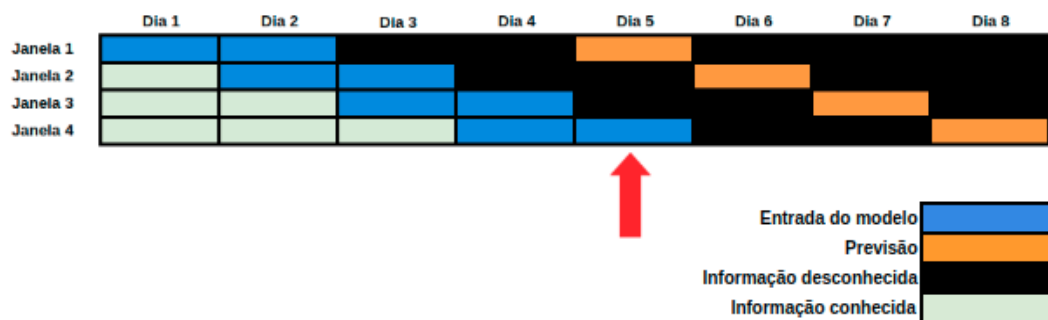
Por fim, o objetivo desse cenário foi utilizar TL com um modelo treinado previamente em uma base de dados localizada próximo a base de dados alvo. Assim, o modelo foi treinado previamente utilizando a base de São Miguel dos Campos e, posteriormente, foi realizado o treinamento incremental do modelo utilizando os dados de Angelim. Novamente, foram utilizadas todas as informações disponíveis para realizar a previsão.

4.5 Atualização do modelo de previsão

Como a abordagem do treinamento será realizada de maneira incremental, o treino do modelo, a partir da chegada de novos dados, só poderá ser realizado uma vez que a quantidade de dias passados seja igual ao tamanho da janela deslizante. Pois, em um cenário real, o dia previsto só será conhecido com o passar do tempo e assim, será possível fornecer entradas e saída esperada para o treinamento da RNA.

Considere o seguinte cenário: é desejado realizar a previsão fornecendo 2 dias como entrada de um modelo e obter a informação 3 dias à frente. A Figura 11 representa essa configuração.

Figura 11 - Momento de retreinamento do modelo



Fonte: De autoria própria.

As três primeiras janelas só poderão realizar previsão, uma vez que para realizar o treinamento, é necessário fornecer o valor correto da saída, mediante ao conjunto de entradas associado. A seta vermelha da Figura 11 indica quando o modelo estará apto a realizar o treinamento com um novo conjunto de informações. Neste caso, apenas a partir da quarta janela, quando a informação utilizada no primeiro treinamento se torna conhecida, e o aprendizado supervisionado pode então ser realizado. A partir daí, a cada dia subsequente, será possível treinar o modelo antes de realizar uma nova previsão.

4.6 Configurações e métricas de avaliação dos modelos

Objetivando realizar a comparação com configurações semelhantes, os modelos utilizados possuíram as mesmas configurações, com a exceção do modelo LSTM, que possuiu uma camada escondida extra com 16 células.

As RNAs utilizadas possuíram apenas uma camada escondida e utilizaram a função de ativação ReLU, do Inglês *rectified linear unit*. Além disso, foi utilizado o otimizador Adam.

No Capítulo 5 serão apresentadas, previamente, tabelas contendo informações numéricas sobre os parâmetros das RNAs, tais como: quantidade de dados na camada de entrada, quantidade de neurônios/células na camada escondida, taxa de aprendizagem e métricas que avaliam as taxas de erros, que

utilizam os valores previstos com os valores reais. Nesse caso, as métricas utilizadas serão o *Root Mean Squared Error* (RMSE) e *Mean Absolute Error* (MAE).

5. RESULTADOS E DISCUSSÕES

Este capítulo contém os resultados e suas discussões. Para os gráficos encontrados ao decorrer do capítulo, as linhas azuis representam os valores reais, enquanto que as linhas vermelhas representam os valores previstos pela RNA. Além disso, a plotagem encontra-se baseada nos valores normalizados do risco.

5.1 Resultados do cenário 1

A seguir, a Tabela 1 apresenta os parâmetros das três configurações citadas anteriormente para o cenário 1, sem uso de TL.

Tabela 1 - Configurações do cenário 1 (MLP e LSTM)

Configuração	Colunas utilizadas	Neurônios da camada escondida	Épocas	Taxa de aprendizagem	Dias utilizados na entrada	Dias à frente a ser previsto
1	Todas	32	8	0,01	5	1
2	Risco	64	8	0,01	2	10
3	Todas	64	8	0,01	2	10

Fonte: De autoria própria.

A Tabela 2 possui os resultados mais relevantes obtidos com a RNA MLP.

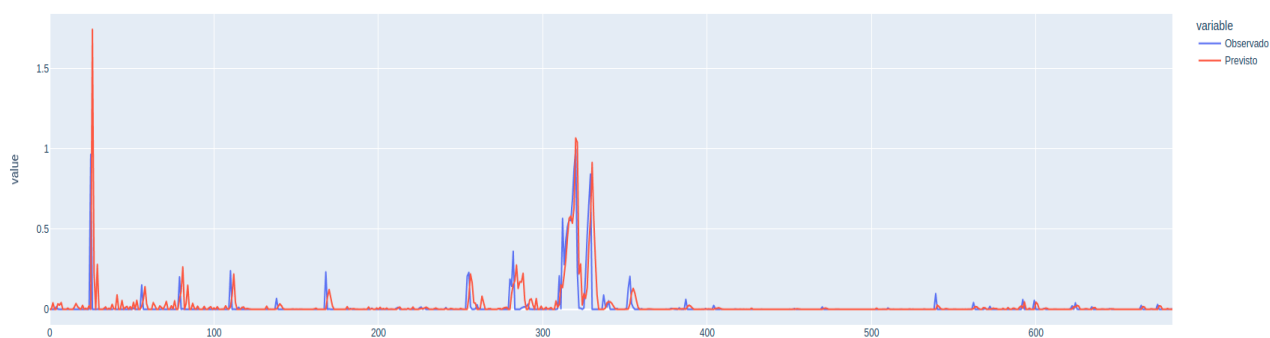
Tabela 2 - Métricas de avaliação do cenário 1 (MLP)

Configuração	Base de dados	RMSE	MAE
1	Angelim	300,37	73,03
1	São Miguel dos Campos	336,48	91,90
2	Angelim	341,10	93,70
2	Coruripe	331	161,11
3	Angelim	336,20	89,85
3	Coruripe	335,65	165,51

Fonte: De autoria própria.

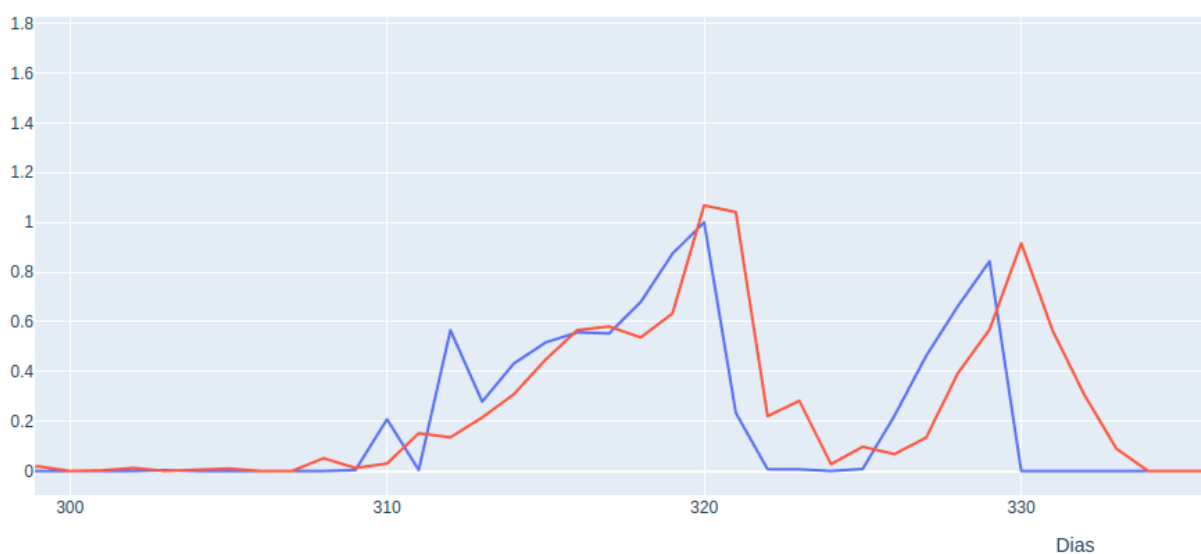
A Figura 12.a ilustra a configuração 1, utilizando Angelim como base. Pode ser observado que inicialmente, a rede oscilou bastante, mas passados 300 dias, o comportamento da previsão se mostrou mais estável, como mostrado na Figura 12.b. É possível observar que existe uma defasagem entre o valor previsto e o observado.

Figura 12.a - Resultado da configuração 1 (MLP) para Angelim



Fonte: De autoria própria.

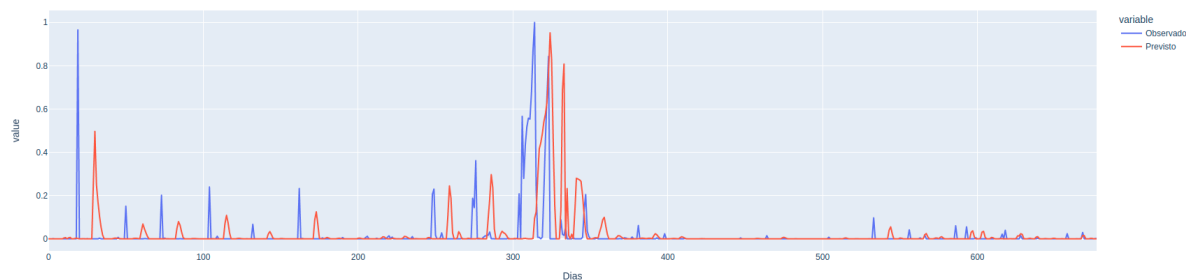
Figura 12.b - Resultado da configuração 1 (MLP) para Angelim



Fonte: De autoria própria.

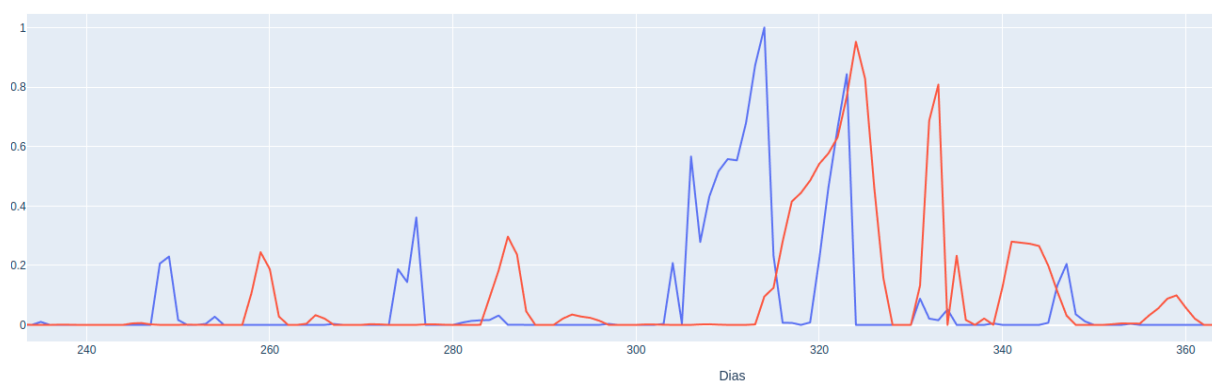
Já na Figura 13.a, utilizando a configuração 2 para Angelim, foi observado uma defasagem maior em relação ao risco observado, embora o comportamento tenha sido replicado, como pode ser visualizado na Figura 13.b.

Figura 13.a - Resultado da configuração 2 (MLP) para Angelim



Fonte: De autoria própria.

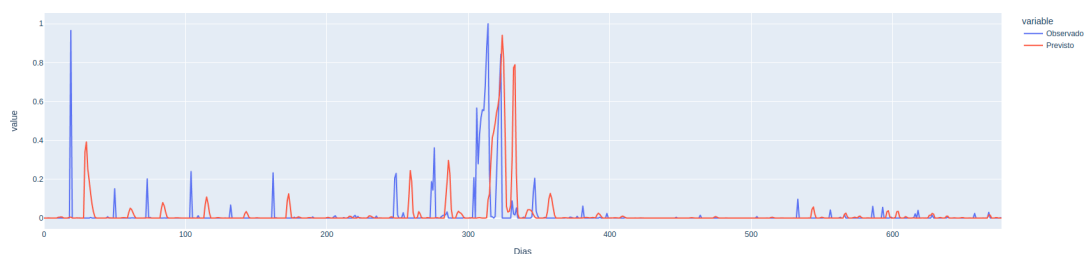
Figura 13.b - Resultado da configuração 2 (MLP) para Angelim



Fonte: De autoria própria.

Já na configuração 3, utilizando todas as colunas, houve uma pequena redução nas métricas utilizadas para avaliação da RNA quando comparado com os resultados da configuração 2. Mesmo assim, a defasagem ainda se faz presente, como é possível observar na Figura 14.

Figura 14 - Resultado da configuração 3 (MLP) para Angelim



Fonte: De autoria própria.

Utilizando a arquitetura LSTM, os resultados podem ser visualizados na Tabela 3.

Tabela 3 - Métricas de avaliação do cenário 1 (LSTM)

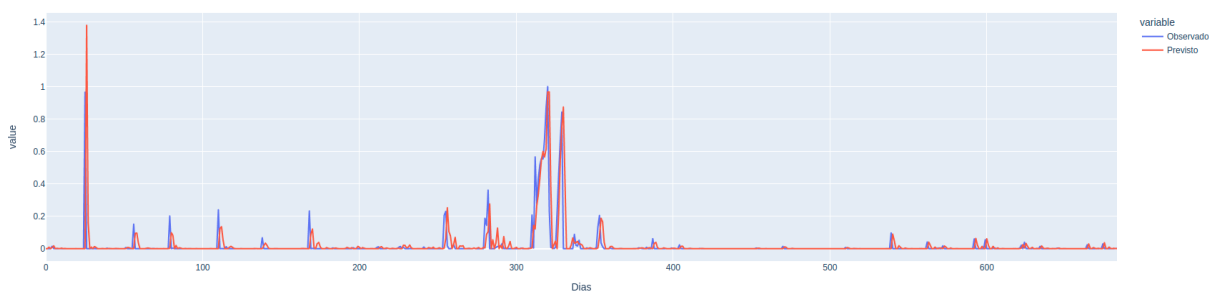
Configuração	Base de dados	RMSE	MAE
1	Angelim	258,48	57,72
1	São Miguel dos Campos	292,75	71,11
2	Angelim	332,75	86,56
2	Coruripe	332,49	157,90
3	Angelim	345,40	89,33
3	Coruripe	330,58	156,22

Fonte: De autoria própria.

Em comparação com a arquitetura MLP, houve uma redução, de modo geral, nas métricas escolhidas para avaliação, levando em consideração que a mesma configuração foi utilizada para os dois tipos de rede.

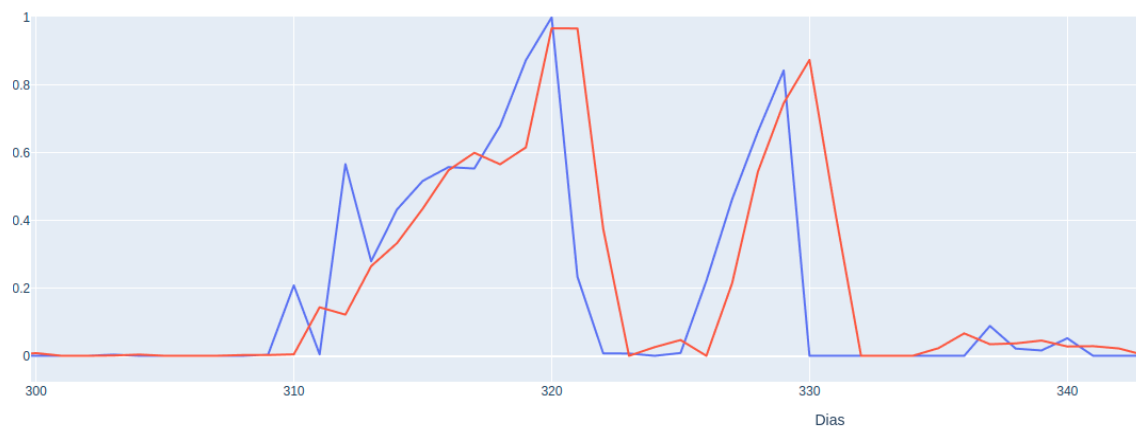
A Figura 15.a mostra que para Angelim, após 300 dias, a rede passou a oscilar menos nas previsões realizadas. A Figura 15.b consiste na ampliação dessa observação.

Figura 15.a - Resultado da configuração 1 (LSTM) para Angelim



Fonte: De autoria própria.

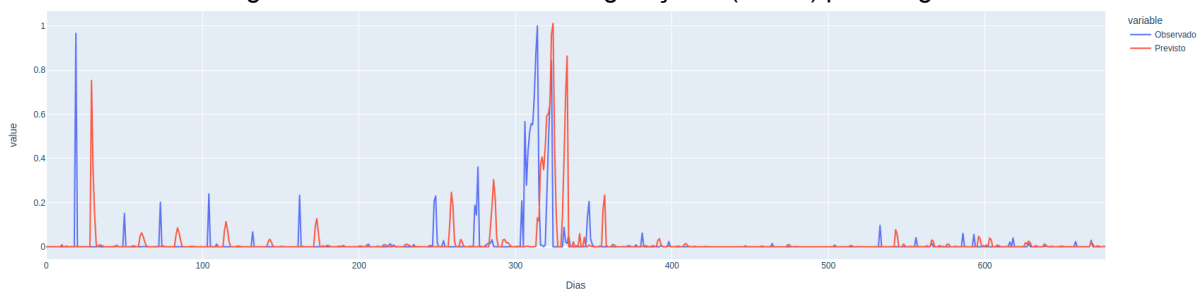
Figura 15.b - Resultado da configuração 1 (LSTM) para Angelim



Fonte: De autoria própria.

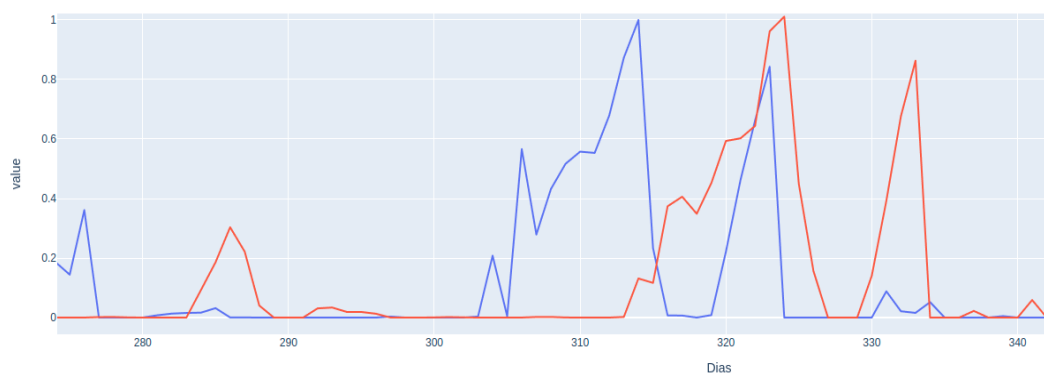
Novamente, quando utilizado um valor maior em relação a quantos dias à frente será realizada a previsão, a Figura 16.a e Figura 16.b mostra que a defasagem se manteve para a LSTM também.

Figura 16.a - Resultado da configuração 2 (LSTM) para Angelim



Fonte: De autoria própria.

Figura 16.b - Resultado da configuração 2 (LSTM) para Angelim



Fonte: De autoria própria.

No caso da arquitetura LSTM, a configuração 3 apresentou uma piora nos resultados, como pode ser verificado na Tabela 3.

5.2 Resultados do cenário 2

A Tabela 4 possui informações relacionadas ao cenário 2, contendo informações antes e depois do treinamento incremental, onde foi realizado TL, utilizando o modelo treinado em uma base de dados distante da base de dados alvo.

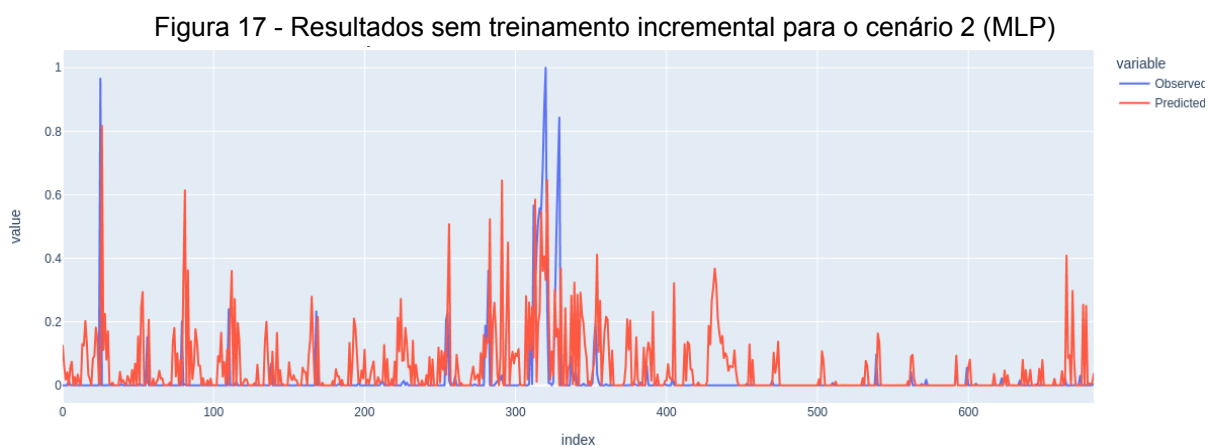
Para ambas as arquiteturas, foram utilizados 32 neurônios na camada escondida, 128 épocas e taxa de aprendizagem igual a 0,00001, utilizando todas as informações presentes em 5 dias para o risco de 1 dia à frente.

Tabela 4 - Métricas de avaliação do cenário 2 (MLP)

Base de dados alvo	RMSE antes do treinamento incremental	RMSE após o treinamento incremental	MAE antes do treinamento incremental	MAE após o treinamento incremental
Angelim	360,98	318,84	169,13	131,68
São Miguel dos Campos	348,98	352,17	125,84	141,68

Fonte: De autoria própria.

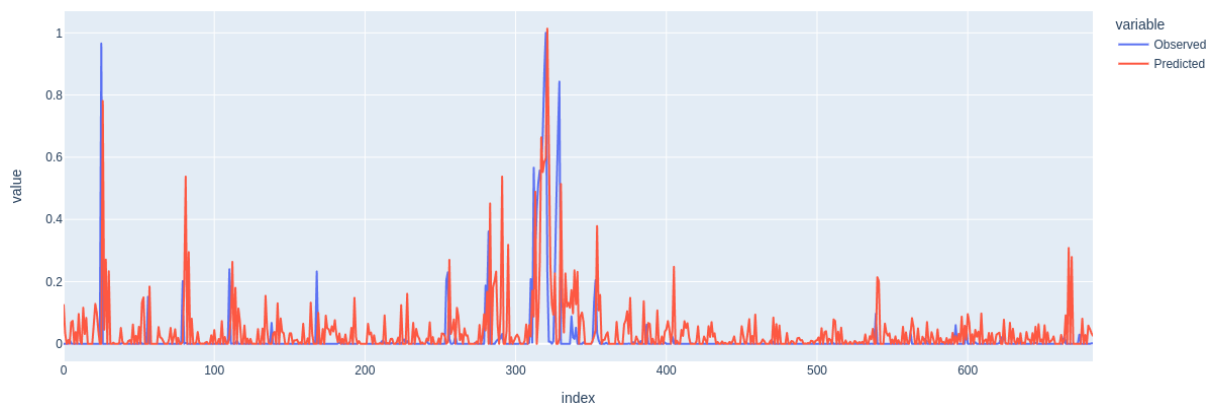
A Figura 17 mostra o gráfico em relação ao risco observado e previsto em relação a Angelim, sem o treinamento incremental.



Fonte: De autoria própria.

Não foram obtidos resultados satisfatórios utilizando essa abordagem com o uso de MLP, como mostrado na Figura 18.

Figura 18 - Resultados com treinamento incremental para o cenário 2 (MLP)



Fonte: De autoria própria.

Utilizando as mesmas configurações que a MLP, foi possível obter melhores resultados com LSTM, como pode ser visto na Tabela 5.

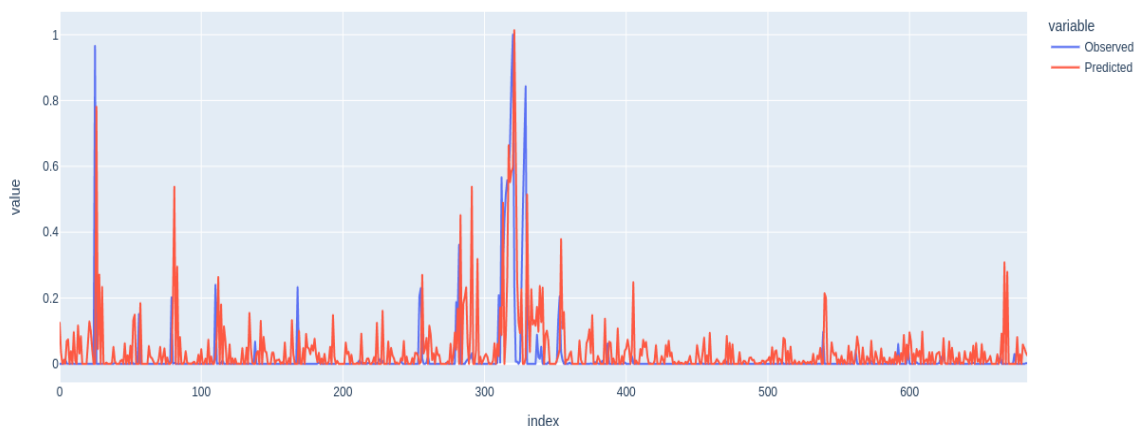
Tabela 5 - Métricas de avaliação do cenário 2 (LSTM)

Base de dados alvo	RMSE antes do treinamento incremental	RMSE após o treinamento incremental	MAE antes do treinamento incremental	MAE após o treinamento incremental
Angelim	301,51	240,93	153,08	75,26
São Miguel dos Campos	364,96	335,35	133,66	107,24

Fonte: De autoria própria.

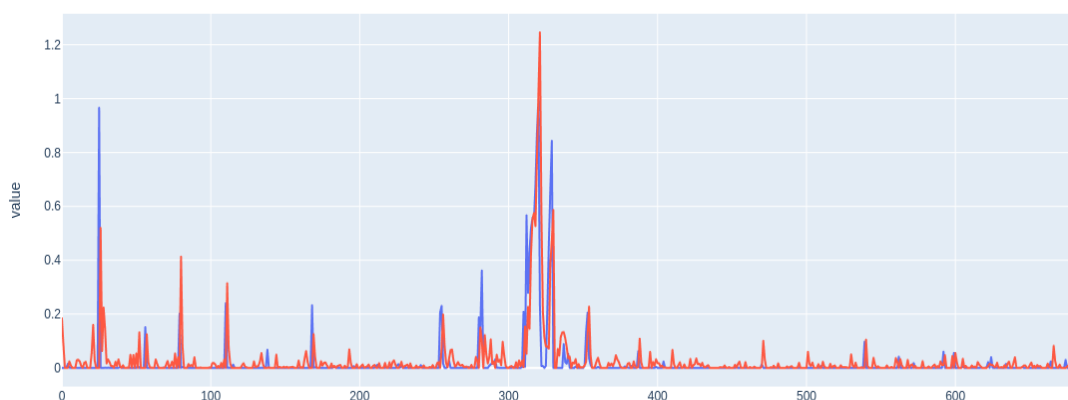
As métricas de avaliação foram melhores utilizando LSTM em comparação a MLP, e isso também é constatado ao analisar a Figura 19 e Figura 20.a, que correspondem a previsão somente com o treinamento de uma base distante e com o posterior treinamento incremental, respectivamente. As figuras são baseadas em Angelim.

Figura 19 - Resultados sem treinamento incremental para o cenário 2 (LSTM)



Fonte: De autoria própria.

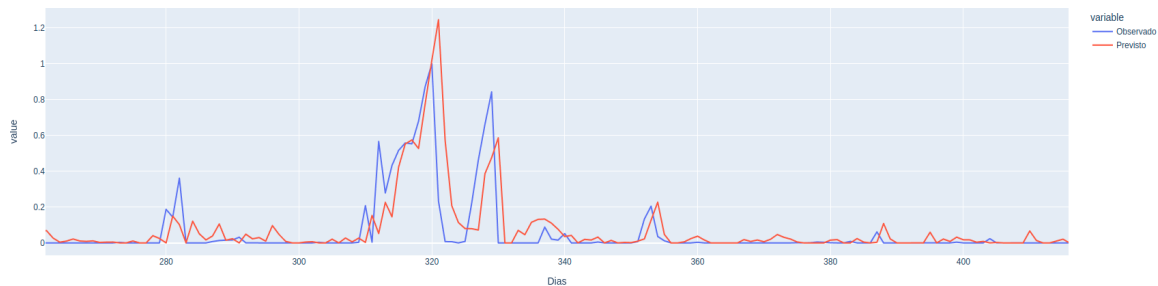
Figura 20.a - Resultados com treinamento incremental para o cenário 2 (LSTM)



Fonte: De autoria própria.

Novamente, após 300 dias de reajustes consecutivos, a rede LSTM foi capaz de prever de maneira aceitável o risco, além de ter reduzido oscilações quando comparado com a MLP, como pode ser observado na Figura 20.b.

Figura 20.b - Resultados com treinamento incremental para o cenário 2 (LSTM)



Fonte: De autoria própria.

5.3 Resultados do cenário 3

Nesse cenário, um modelo foi treinado previamente com a base de São Miguel dos Campos e em seguida, foi apresentado à base de Angelim, com o aprendizado incremental.

As configurações de ambas as arquiteturas foram as seguintes: 64 neurônios na camada escondida, taxa de aprendizagem de 0,01 e 256 épocas, onde a entrada da RNA contém todas as informações de 5 dias consecutivos para prever o risco 1 dia à frente.

A Tabela 6 contém as métricas relacionadas a esse cenário tanto para MLP quanto para LSTM, antes e depois do treinamento incremental.

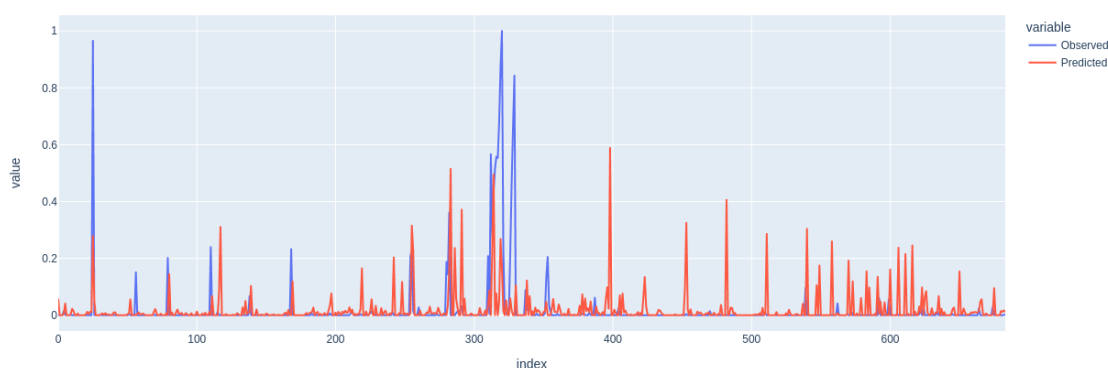
Tabela 6 - Métricas de avaliação do cenário 3 (LSTM)

Arquitetura	RMSE antes do treinamento incremental	RMSE após o treinamento incremental	MAE antes do treinamento incremental	MAE após o treinamento incremental
MLP	286,20	224,70	89,43	64,11
LSTM	245,98	265,53	61,93	55,09

Fonte: De autoria própria.

Embora o RMSE tenha sido ligeiramente maior com o treinamento incremental da LSTM, ambas as arquiteturas apresentaram melhores resultados após o treinamento incremental. A Figura 21 mostra o resultado gráfico antes do treinamento incremental ser utilizado, para o caso da MLP.

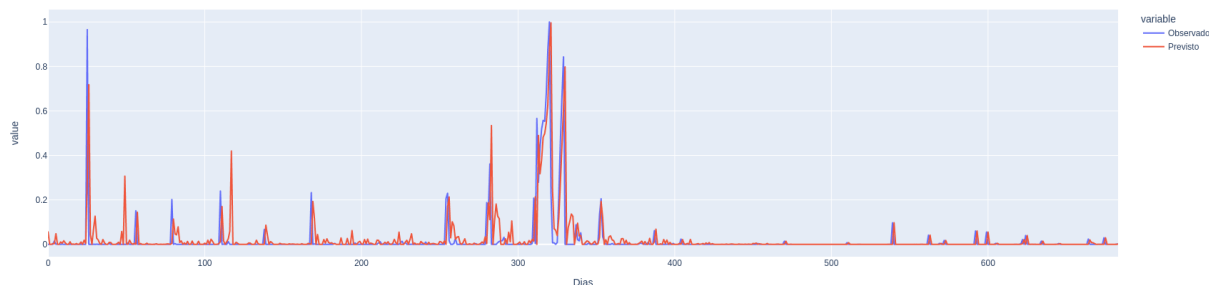
Figura 21 - Resultados sem treinamento incremental para o cenário 3 (MLP)



Fonte: De autoria própria.

Ao utilizar o treino incremental, foi encontrado o resultado gráfico mostrado na Figura 22.a.

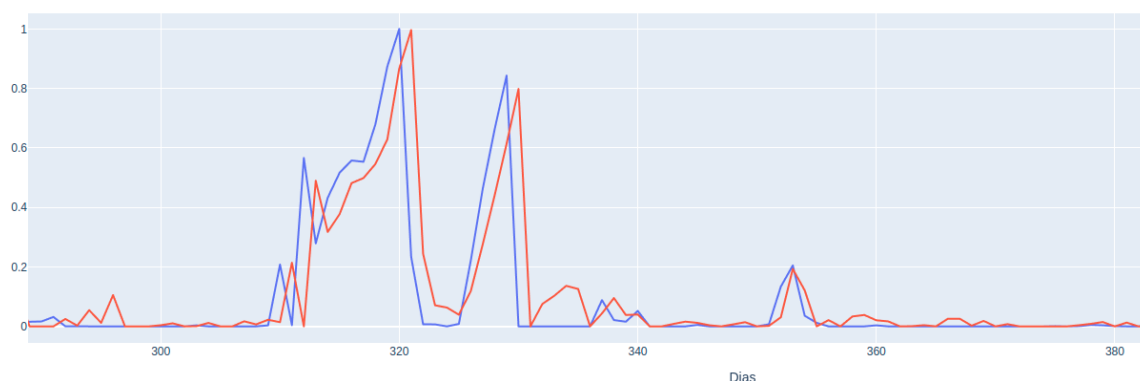
Figura 22.a - Resultados com treinamento incremental para o cenário 3 (MLP)



Fonte: De autoria própria.

Após cerca de 300 dias, a rede MLP apresentou menos oscilações, como mostrado na Figura 22.b.

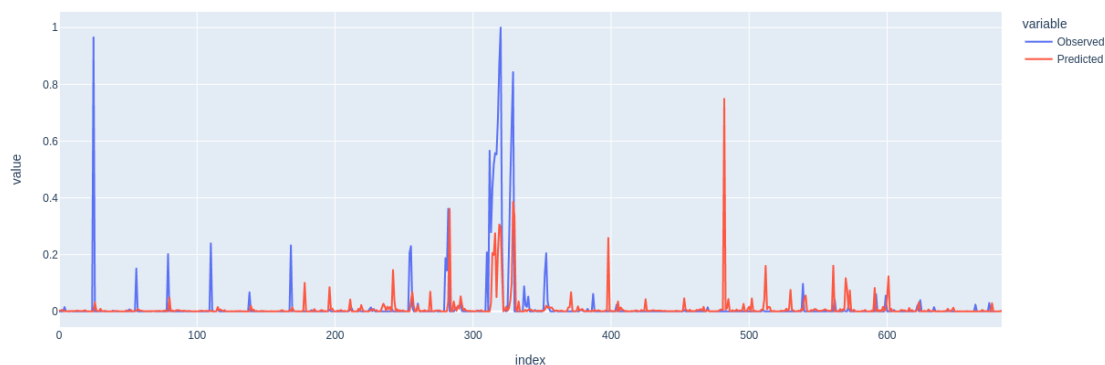
Figura 22.b - Resultados com treinamento incremental para o cenário 3 (MLP)



Fonte: De autoria própria.

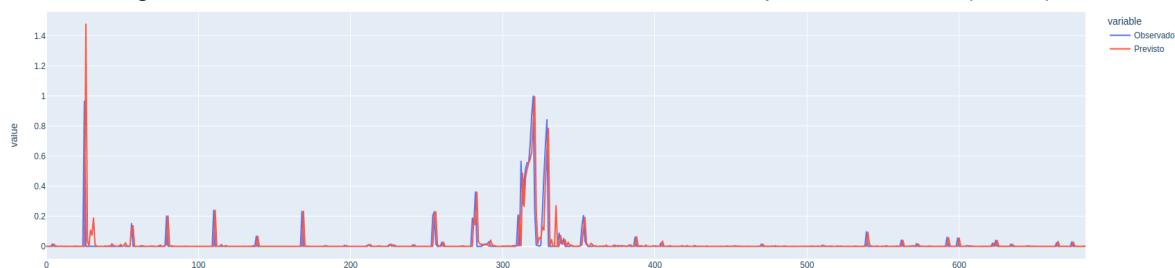
Já com o uso da RNA LSTM, os resultados antes e após o treinamento incremental podem ser vistos na Figura 23 e na Figura 24.a, respectivamente.

Figura 23 - Resultados sem treinamento incremental para o cenário 3 (LSTM)



Fonte: De autoria própria.

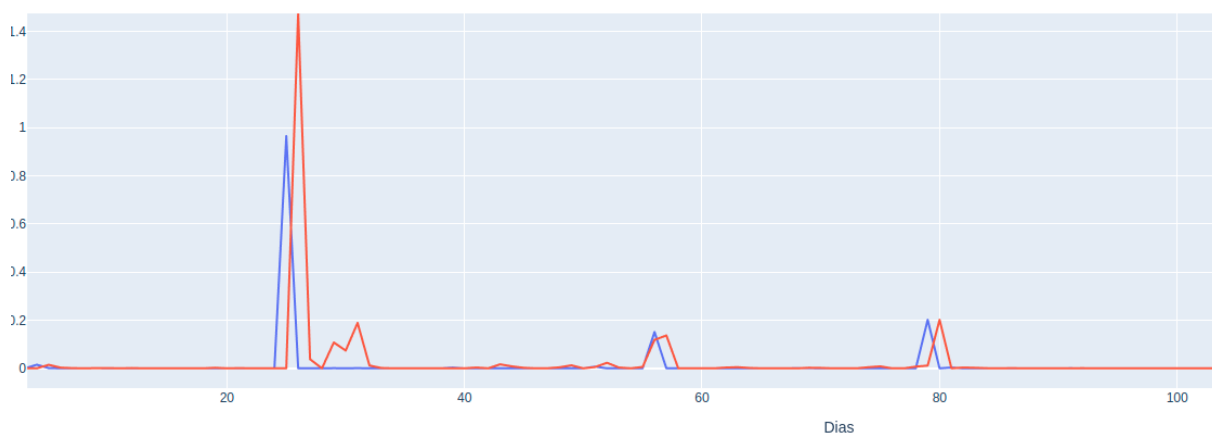
Figura 24.a - Resultados com treinamento incremental para o cenário 3 (LSTM)



Fonte: De autoria própria.

Para o cenário 3, o uso da RNA LSTM oscilou menos próximo aos 100 dias, como pode ser visto na Figura 24.b.

Figura 24.b - Resultados com treinamento incremental para o cenário 3 (LSTM)



Fonte: De autoria própria.

5.4 Considerações

Em experimentos realizados, foi observado que a depender da taxa de aprendizagem utilizada, a defasagem entre o valor real e o previsto é alterada. Além

disso, a escolha da normalização min-max pode ter influenciado nos resultados, uma vez que os valores de N1 - N4 possuem um valor máximo devido a limitação da quantidade de bits transmitidos pelos sistemas de monitoramento, assim como o valor da umidade possui um intervalo bem definido, já que varia de 0 a 100.

De modo geral, o desempenho da RNA LSTM se mostrou superior ao desempenho da RNA MLP. Para as configurações utilizadas, as métricas de avaliação dos modelos se mostraram melhores com o uso de treinamento incremental. Porém, podem ser aplicados algoritmos de otimização para encontrar os melhores parâmetros para ambas as arquiteturas de RNAs.

6. CONCLUSÃO E TRABALHOS FUTUROS

O *flashover* que ocorre em cadeias de isoladores de alta tensão devido ao acúmulo de poluentes nos isoladores, é um problema que tem impactos diretos na sociedade. Uma vez que tenha ocorrido, é necessário realizar a substituição, o que apresenta um custo envolvido tanto na manutenção quanto em relação ao tempo gasto sem o fornecimento de energia.

Trabalhos citados anteriormente atuaram na investigação de mensurar o risco do *flashover*, bem como realizar sua previsão, de modo a se precaver ou mesmo evitar realizações de manutenções desnecessárias.

O presente trabalho contribui para a problemática, uma vez que introduz uma nova maneira de realizar esta tarefa, utilizando aprendizado incremental. Além disso, foi possível fornecer uma base de conhecimento que pode servir para trabalhos futuros. De modo geral, foi observado a redução das métricas RMSE e MAE utilizando o treinamento incremental para as duas arquiteturas utilizadas, onde a LSTM obteve melhores métricas.

Em relação a base de dados utilizada, Angelim apresentou melhores resultados, necessitando 300 dias até o modelo conseguir prever de maneira mais aceitável, para os cenários de não haver um treinamento prévio ou haver o uso de TL de um modelo treinado previamente utilizando uma base de dados distante. Enquanto que no cenário em que o modelo foi treinado previamente com uma base de dados mais próxima a Angelim, foi possível observar uma melhora da RNA após 100 dias.

Existe espaço para a melhoria do modelo utilizado neste trabalho, e contribuições futuras podem se atentar a aspectos como:

- Analisar a influência da normalização min-max nos resultados obtidos, visto que os valores das contagens de pulso possuem um limite máximo devido a capacidade total de bits que pode ser transmitida pelo sistema de monitoramento, assim como o valor da umidade varia de 0 a 100;
- Utilizar a abordagem proposta nesse trabalho considerando o seguinte cenário: utilizar uma RNA sem treinamento prévio, realizando um treinamento na base de dados alvo e em seguida, executar o treinamento incremental

nessa mesma base, com o objetivo de saber quando o modelo começa a prever de maneira satisfatória;

- Realizar uma otimização nos parâmetros utilizados pelas RNAs, de modo a reduzir a defasagem entre o valor previsto e o valor real que foi observado.
- Realizar a previsão de maneira incremental de outras variáveis, como as contagens de pulso ocorridas (N1 - N4);
- Investigar, de maneira mais detalhada, a definição do pré-processamento e a escolha dos cenários de teste. Nesse sentido, analisar como deverão ser agrupados os dados (em dias, horas...) assim como qual deve ser o tamanho ideal da janela deslizante;
- Estudar a influência que período inicial de treinamento escolhido para a RNA aprender de maneira incremental tem em relação ao desempenho final do modelo.

REFERÊNCIAS

BAYLISS, C. R.; HARDY, B. J. **Transmission and distribution electrical engineering**. Oxford ; Boston: Elsevier/ Newnes, 2012.

Contamination Flashover for porcelain and polymer insulator. Disponível em: <<https://www.powerinsulator.com/articles/flashover-for-porcelain-polymer.html>>. Acesso em: 10 maio. 2022.

FONTANA, E. et al. *Sensor Network for Monitoring the State of Pollution of High-Voltage Insulators Via Satellite*. **IEEE Transactions on Power Delivery**, v. 27, n. 2, p. 953–962, 2012.

GARDNER, M. W.; DORLING, S. R. *Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences*. **Atmospheric Environment**, v. 32, n. 14-15, p. 2627–2636, ago. 1998.

HAMPTON, B. *Flashover mechanism of polluted insulation*. **Electronics and Power**, v. 10, n. 4, p. 113, 1964.

HOCHREITER, S., SCHMIDHUBER, J.: *Long Short-Term Memory*. **Neural Computation**, Vol 9, No. 8, 1997.

KLIR, G. J.; YUAN, B. *Fuzzy sets and fuzzy logic: theory and applications*. Tradução . [s.l.] **Pearson**, 2015.

LIMA, C. J. T. D., **Processo de Tomada de Decisão em Projetos de Exploração e Produção de Petróleo no Brasil: Uma Abordagem Utilizando conjuntos Nebulosos**. 2003. Tese de Doutorado. UNIVERSIDADE FEDERAL DO RIO DE JANEIRO.

LIMA, H. O. D.; OLIVEIRA, S. C.; FONTANA, E. *Flashover risk prediction on polluted insulators strings of high voltage transmission lines*. **2011 11th International Conference on Intelligent Systems Design and Applications**, nov. 2011a.

LIMA, H. O. D.; OLIVEIRA, S. C.; FONTANA, E. *Fuzzy inference system for risk classification on polluted insulator strings of high voltage transmission lines*. **2011 SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (IMOC 2011)**, 2011b.

MCCULLOCH, W.; PITTS, W. *A logical calculus of the ideas immanent in nervous activity*. **Bulletin of Mathematical Biology**, v. 52, n. 1-2, p. 99–115, 1990.

MELO, G S. **Aplicação de Aprendizado de Máquina para Previsão de Fluxo de Caixa em ATMs**. Orientador: Ricardo Bastos Cavalcante Prudêncio. 37f. Monografia - Engenharia da Computação, Centro de Informática, Universidade Federal de Pernambuco, Recife, 2018.

MENEGOLA, A. et al. *Knowledge transfer for melanoma screening with deep learning*. **2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)**, abr. 2017.

MITCHELL, T. M. **Machine learning**. Singapore: McGraw-Hill, 1997.

MOHANASUNDARAM, R. et al. *Deep Learning and Semi-Supervised and Transfer Learning Algorithms for Medical Imaging*. **Deep Learning and Parallel Computing Environment for Bioengineering Systems**, p. 139–151, 2019.

MOREIRA, S. **Rede Neural Perceptron Multicamadas**. Disponível em: <<https://medium.com/ensina-ai/rede-neural-perceptron-multicamadas-f9de8471f1a9>>. Acesso em: 05 maio. 2022.

-
- R.R, A.; P. R, D. *Methods for Incremental Learning : A Survey*. **International Journal of Data Mining & Knowledge Management Process**, v. 3, n. 4, p. 119–125, 31 jul. 2013.
- RUSSELL, S.; NORVIG, P. **Artificial intelligence : a modern approach**. 3. ed. New Jersey: Pearson, 2010.
- SILVA, L. H. S.; OLIVEIRA, S. C.; FONTANA, E. “Evaluation of Machine Learning Methods in a Rain Detection System for Partial Discharge Data Analysis”. **The Second International Conference on Intelligent Systems and Applications, 2013, Venice**. INTELLI 2013, 2013. p. 176-183.
- SILVA, L. H. S. **Aplicação de Reconhecimento de Padrões para Detecção de Chuva e Melhoria em Sistema de Classificação de Risco de Flashover em Linhas de Transmissão de Alta Tensão**. Orientador: Sérgio Campello Oliveira. 93f. Dissertação (Mestrado) - Engenharia da Computação, Escola Politécnica de Pernambuco, Universidade de Pernambuco, Recife, 2013.
- TSARABARIS, P.; KARAGIANNOPOULOS, C.; THEODOROU, N. *A model for high voltage polluted insulators suffering arcs and partial discharges*. **Simulation Modelling Practice and Theory**, v. 13, n. 2, p. 157–167, 2005.
- Yu, Y. et al. *A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures*. **Neural Computation**, v. 31, n. 7, p. 1235–1270, jul. 2019.
- ZADEH, L. A. *Fuzzy sets*, **Information Control**, vol. 8, pp. 338–353, 1965.