



MIGUEL FERNANDES SALVADO

Licenciatura em Engenharia Mecânica

MODELO DE PREVISÃO DE AUSÊNCIAS DE TRIPULANTES DE TRANSPORTES FERROVIÁRIOS

MESTRADO EM ANÁLISE E ENGENHARIA DE BIG DATA

Universidade NOVA de Lisboa
Setembro, 2023



NOVA

NOVA SCHOOL OF
SCIENCE & TECHNOLOGY

DEPARTAMENTO DE
INFORMÁTICA

MODELO DE PREVISÃO DE AUSÊNCIAS DE TRIPULANTES DE TRANSPORTES FERROVIÁRIOS

MIGUEL FERNANDES SALVADO

Licenciatura em Engenharia Mecânica

Orientadores: Rui Rodrigues

Professor, Universidade NOVA de Lisboa

Ricardo Saldanha

Diretor do Departamento de Inovação, SISCOG - Sistemas Cognitivos, SA

MESTRADO EM ANÁLISE E ENGENHARIA DE BIG DATA

Universidade NOVA de Lisboa

Setembro, 2023

Modelo de previsão de ausências de tripulantes de transportes ferroviários

Copyright © Miguel Fernandes Salvado, Faculdade de Ciências e Tecnologia, Universidade NOVA de Lisboa.

A Faculdade de Ciências e Tecnologia e a Universidade NOVA de Lisboa têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objetivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

AGRADECIMENTOS

Primeiro gostaria de agradecer aos meus orientadores, Rui Rodrigues e Ricardo Saldanha, pela orientação e conhecimento compartilhado ao longo deste processo. A participação e apoio constante foram fundamentais para o sucesso desta tese.

Queria agradecer também à empresa SISCOG, pela oportunidade de fazer parte deste projeto, especialmente aos colegas Luís Albino e Gonçalo Matos, por dedicarem o seu tempo e conhecimentos para o desenvolvimento deste projeto.

Não posso deixar de agradecer aos meus pais e irmã pelo apoio constante ao longo desta jornada académica.

RESUMO

As ausências não planeadas podem causar situações extremamente indesejáveis para as empresas em geral e para os operadores ferroviários em especial. De forma a responder às ausências, o operador estabelece um contingente de tripulantes de reserva consoante as previsões que possam existir para a quantidade de ausências. Para minimizar os custos, o operador tenta limitar o número de reservas. Porém, se não existirem tripulantes de reserva suficientes para preencher as necessidades, é possível negociar com um trabalhador que não estava planeado trabalhar nesse dia, com um custo adicional. No caso de não existir nenhum trabalhador disponível, a empresa pode ver-se obrigada a atrasar ou até mesmo a cancelar comboios, criando assim um problema não só para o operador como para os passageiros.

A escassez de recursos humanos que se tem verificado no sector ferroviário e a incerteza sobre o número de tripulantes de reserva necessários para cada hora do dia, acentua o problema mencionado anteriormente.

Esta dissertação tem como objetivo a criação de um modelo preditivo que seja eficaz na previsão do número de maquinistas e revisores, de um operador ferroviário, que possam estar ausentes numa determinada data por fatores relacionados com a sua própria saúde ou com a saúde da sua família. Assim, o modelo criado irá auxiliar o operador no processo de planeamento dos tripulantes de reserva e conseqüentemente amenizar os custos associados às ausências dos mesmos. Foram criados vários modelos recorrendo às redes neuronais, como redes Feedforward, Transformers e recorrentes, nomeadamente Long Short-Term Memory, obtendo os melhores resultados nas redes Long Short-Term Memory.

Neste projeto são utilizados dados históricos, nomeadamente entre os anos 2009 e 2021, fornecidos por um operador ferroviário no Norte da Europa. Estes dados incluem dados biográficos e registos da atividade de 4000 tripulantes geograficamente distribuídos por 24 bases operacionais.

Palavras-chave: Rede neuronal artificial, Série temporal, Previsão de ausências, Aprendizagem automática

ABSTRACT

Unplanned absences can lead to highly undesirable situations for companies in general and especially for railway operators. In order to address these absences, the operator establishes a reserve crew contingent based on any forecasts for the expected number of absences. To minimize costs, the operator seeks to limit the number of reserves. However, if there are not enough reserve crew members to meet the needs, it is possible to negotiate with a worker who was not originally scheduled to work that day, incurring additional costs. In the absence of any available workers, the company may be forced to delay or even cancel trains, creating a problem not only for the operator but also for passengers.

The shortage of human resources in the railway sector and the uncertainty about the number of required reserve crew members for each hour of the day aggravate the previously mentioned problem. The objective of this dissertation is to develop a predictive model effective in forecasting the number of drivers and guards from a railway operator who may be absent on a given date due to factors related to their own health or the health of their family. Thus, the proposed model will assist the operator in the process of planning reserve crews and consequently mitigate the costs associated with their absences. Various models have been developed using neural networks, including Feedforward, Transformers and recurrent networks, notably Long Short-Term Memory, with the best results obtained in Long Short-Term Memory networks.

This project utilizes historical data, specifically from the years 2009 to 2021, provided by a railway operator in Northern Europe. These data include biographical information and activity records of 4000 crew members geographically distributed across 24 operational bases.

Keywords: Artificial neural network, Time series, Absence prediction, Machine learning

ÍNDICE

Índice de Figuras	viii
Índice de Tabelas	ix
Siglas	xi
1 Introdução	1
1.1 Descrição do Problema	2
1.2 Objetivo	2
2 Estado da Arte	3
2.1 Revisão de Literatura	3
2.1.1 Trabalho Prévio	3
2.1.2 Dados	4
2.1.3 Métodos	4
2.2 Métodos	6
2.2.1 Auto-Regressão	6
2.2.2 Rede Neuronal	6
2.3 Métricas	16
2.3.1 Erro quadrático médio	16
2.3.2 Erro médio absoluto	16
2.3.3 Erro médio absoluto percentual	16
2.3.4 Precisão e Recall	17
3 Trabalho Realizado	18
3.1 Descrição da Base de Dados	18
3.1.1 Dados de entrada	20
3.1.2 Dados de saída	21
3.2 Modelos	22
3.2.1 Dados agregados	22

3.2.2	Dados desagregados	24
3.2.3	Dados sequenciais	26
3.2.4	Resultados	28
3.3	Otimização de modelos	28
3.3.1	Dados de validação	28
3.3.2	Função de perda	29
3.3.3	Previsões	29
3.4	Tempos de antecedência	29
3.5	Adicionar revisores	31
3.6	Bases operacionais	32
4	Conclusão	36
	Bibliografia	37

ÍNDICE DE FIGURAS

2.1	Estrutura de uma Rede Neuronal	6
2.2	Exemplo de uma Rede Neuronal	7
2.3	Função de ativação Sigmoid	8
2.4	Função de ativação ReLU	9
2.5	Função de ativação Tanh	9
2.6	Percetrão - rede neuronal composta por apenas um neurónio	10
2.7	Exemplo de uma Rede Neuronal Convolutacional	12
2.8	Exemplo de uma Rede Neuronal Recorrente	13
2.9	Exemplo de uma Rede LSTM	13
2.10	Exemplo de uma Rede Transformer	14
2.11	Divisão dos dados	15
3.1	Diagrama do dataset	18
3.2	Exemplo do conjunto de dados agregados	20
3.3	Exemplo do conjunto de dados sequenciais	21
3.4	Exemplo do conjunto de dados de saída agregados	21
3.5	Exemplo do conjunto de dados de saída desagregados	21
3.6	Previsão do modelo de auto-regressão para o ano 2019	22
3.7	Previsão do modelo feedforward baseado no rácio para o ano 2019	24
3.8	Previsão do modelo Feedforward para o ano 2019	25
3.9	Previsão do modelo LSTM para o ano 2019	26
3.10	Exemplo do horizonte rolante	30
3.11	Número médio diário de tripulantes com trabalho planeado e ausências por base durante o ano 2019	33
3.12	Erro preditivo MAPE Auxiliar para os maquinistas de cada base e tipo de antecedência no ano 2019	34
3.13	Erro preditivo MAPE Auxiliar para os revisores de cada base e tipo de antecedência no ano 2019	35

ÍNDICE DE TABELAS

2.1	Erro preditivo de ausências de maquinistas referente ao modelo Regressão logística no ano 2019	3
3.1	Atributos disponíveis na base da dados	19
3.2	Erro preditivo referente ao modelo de auto-regressão no ano 2019	23
3.3	Erro preditivo referente ao modelo feedforward no ano 2019	23
3.4	Erro preditivo referente ao modelo feedforward baseado no rácio no ano 2019	23
3.5	Erro preditivo referente ao modelo com dados desagregados no ano 2019	25
3.6	Erro preditivo referente ao modelo LSTM no ano 2019	26
3.7	Erro preditivo referente ao modelo Transformers	27
3.8	Erro preditivo referente ao modelo LSTM com diferentes tipos de previsão para 1 dia de antecedência.	29
3.9	Erro preditivo dos modelos LSTM e Transformer com maquinistas para diferentes antecedências	31
3.10	Erro preditivo do modelo LSTM com revisores para diferentes antecedências	31
3.11	Erro preditivo do modelo LSTM com revisores e maquinistas para diferentes antecedências	32
3.12	Comparação entre MAPE e MAPE Auxiliar na base B1 com o modelo LSTM para maquinistas	34

SIGLAS

AR	Auto-Regressão (pp. 6, 22)
BPTT	Backpropagation Through Time (p. 12)
CNN	Convolutional Neural Network (p. 12)
FN	Falso Negativo (p. 17)
FP	Falso Positivo (p. 17)
LSTM	Long Short-Term Memory (pp. 13, 14, 20, 26–32, 35, 36)
MAE	Mean Absolute Error (pp. 16, 23, 25–27, 32, 33)
MAPE	Mean Absolute Percentage Error (pp. 16, 23, 25–27, 31–34)
MSE	Mean Squared Error (pp. 16, 23, 25–27)
ReLU	Rectified Linear Unit (pp. 8, 24, 25, 27, 28)
RNN	Recurrent Neural Network (p. 12)
SISCOG	Sistemas Cognitivos, SA (pp. 1, 3, 18, 19, 36)
SVM	Support Vector Machine (p. 5)
Tanh	Tangente Hiperbólica (pp. 9, 27)
VN	Verdadeiro Negativo (p. 17)
VP	Verdadeiro Positivo (p. 17)

INTRODUÇÃO

O tema desta dissertação foi proposto pela empresa Sistemas Cognitivos, SA (SISCOG)¹ e consiste em prever o número de ausências em cada dia de operação dos tripulantes de uma operadora ferroviária. A SISCOG é uma empresa de software que fornece soluções para o planeamento e gestão de recursos no setor dos transportes. A SISCOG, que permitiu que esta dissertação fosse feita em ambiente empresarial, desenvolve e comercializa sistemas de apoio à decisão para o planeamento e gestão operacional das viagens de comboio, do material circulante e dos tripulantes. Oferece produtos que, mediante a definição de uma camada de especialização, se adaptam às necessidades específicas de cada cliente, por forma a ajudar as empresas de transportes a melhorar a sua eficiência e rentabilidade. A empresa dispõe de quatro produtos:

- ONTIME - Criação e gestão dos horários das viagens ao longo da rede global (que liga as várias estações) tendo em conta as necessidades de mobilidade dos passageiros e as regras de trânsito ao nível da circulação de comboios.
- FLEET - Planeamento e gestão da atividade dos veículos ao longo da rede global por forma a fornecer os veículos necessários a cada viagem tendo em conta as manutenções periódicas e outras restrições operacionais.
- RAILNODE - Planeamento e gestão da atividade dos veículos dentro da rede local de cada estação tendo em conta as necessidades de estacionamento e outras determinadas pelas decisões tomadas ao nível do FLEET e do ONTIME.
- CREWS - Planeamento e gestão da atividade dos tripulantes quer na rede global quer na rede local, por forma a fornecer os recursos necessários a cada viagem ou operação tendo em conta as regras laborais e outras restrições operacionais.

O tema desta dissertação surgiu da necessidade sentida pela SISCOG em melhorar a capacidade de planeamento de reservas e da gestão de ausências do seu produto CREWS [1, 13], usando para tal a previsão de ausências de tripulantes, dando assim continuidade a

¹<https://www.siscog.pt>

um trabalho previamente realizado internamente [14]. O objetivo é portanto desenvolver um modelo de previsão de ausências que permita apoiar a decisão nos processos de planeamento de reservas, bem como planeamento e gestão das ausências e da formação, por forma a que se melhore a eficiência de ajustar a capacidade disponível às necessidades de trabalho, sobretudo em situações de contingência.

1.1 Descrição do Problema

As ausências não planeadas podem causar situações extremamente indesejáveis para as empresas, principalmente no setor dos transportes públicos e mais especificamente nos ferroviários. No caso de existirem ausências, podem surgir atrasos ou até mesmo cancelamentos de comboios. Isto não só cria um problema para os passageiros como para o próprio operador. Assim, de forma a tentar solucionar este problema, o operador tem um plano constituído por trabalhadores de reserva preparados para substituir alguma ausência que ocorra. No entanto, estes trabalhadores são pagos de igual forma quer sejam necessários ou não.

O objetivo do operador é melhorar o planeamento de reservas, bem como o planeamento e gestão de ausências (e.g. férias) e formações sem comprometer o normal funcionamento da operação. No entanto, se este planeamento não for feito da melhor maneira, pode dar origem a situações em que não há trabalhadores de reserva suficientes para suprir as necessidades, sobretudo em situações de contingência. De forma a evitar uma interrupção do serviço e insatisfação dos clientes é possível ainda negociar com um trabalhador que não estava planeado trabalhar nesse dia, propondo uma compensação paga ou dias adicionais de descanso podendo o trabalhador aceitar ou contrapor. Este tipo de situações gera custos adicionais para o operador.

1.2 Objetivo

O objetivo principal desta dissertação é desenvolver um modelo de aprendizagem automática que possa apoiar o planeamento e gestão da força laboral, que consiste em ajustar a capacidade disponível às necessidades inerentes ao negócio, sobretudo em situações de contingência. Este modelo será usado para prever o número de tripulantes, de um determinado tipo e base operacional, ausentes devido a situações imprevistas num determinado dia, com base em dados históricos dos anos anteriores.

Para isso serão utilizadas redes neuronais feedforward, transformers e recorrentes, nomeadamente Long Short-Term Memory, de forma a prever o número de ausências de maquinistas e revisores de cada base operacional. A intenção é que este modelo possa ser utilizado pelo operador ferroviário para gerir ausências e determinar o número de tripulantes de reserva disponíveis para trabalhar em cada instante ao longo do dia, de forma a garantir que haja um número suficiente de pessoas disponíveis para atender às necessidades do serviço em situações de contingência.

2.1 Revisão de Literatura

Para a execução deste projeto, foram inicialmente explorados vários projetos com temas semelhantes de forma a compreender melhor o problema e possíveis soluções. Assim, foram agrupados vários artigos científicos que utilizam diferentes atributos e abordagens distintas para resolver problemas semelhantes ao objetivo deste projeto.

2.1.1 Trabalho Prévio

Como referido anteriormente na introdução, este tema surgiu de um trabalho previamente realizado pela SISCOG [14], com o objetivo de prever o número de ausências diárias por maquinistas de uma empresa ferroviária. Neste estudo foram utilizados diferentes modelos de aprendizagem automática como Random Forests, XGBoost e Regressão Logística

Estes modelos foram treinados com os dados entre os anos 2013 a 2018 e testados com o ano 2019, apenas com os dados dos maquinistas. Para o teste foram utilizados diferentes datas de previsão com uma antecedência de 7, 30 e 365 dias. Destes testes retiraram-se diferentes métricas para aferir a eficácia dos vários modelos. O modelo com o melhor desempenho foi o modelo de Regressão logística com os resultados apresentados na tabela 2.1.

	MAE	MAPE
Previsão a 7 dias	11.79	25.14%
Previsão a 30 dias	11.89	25.03%
Previsão a 1 ano	11.66	22.27%

Tabela 2.1: Erro preditivo de ausências de maquinistas referente ao modelo Regressão logística no ano 2019

2.1.2 Dados

Um dos fatores importantes a ter em conta nos vários relatórios abordados na pesquisa é o tipo de dados utilizados pelos mesmos. Neste caso, como o objetivo é fazer uma previsão para uma empresa ferroviária e sabendo que neste tipo de empresa os tripulantes trabalham por turnos, de forma a fazer uma boa comparação, os dados têm de seguir o mesmo padrão. Assim, foram explorados relatórios que utilizam dados tanto de outras empresas ferroviárias como de organizações com um horário por turnos. Alguns exemplos são:

- Empresas Ferroviárias [8]
- Seguranças Militares [10]
- Correios [18]
- Professores [5]

2.1.2.1 Atributos

Os autores de [3, 4, 15, 18, 23] utilizaram uma base de dados de uma empresa de correios brasileira que contém dados entre 2007 e 2010 com informações dos trabalhadores como a idade, número de filhos, distância ao trabalho, educação e número de anos na empresa. Utilizam também informações do trabalho realizado como a sua duração e em que dia e mês ocorreu.

Ferreira (2019) utilizou a mesma base de dados mencionada anteriormente. Para além dos atributos já mencionados, foram também criados novos atributos como a estação do ano, número de licenças médicas registadas no mês, custo de deslocação, carga total de trabalho mensal, remuneração média, absentismo mensal.

Fernandes & Filho (2020) utilizaram uma base de dados de instituições públicas e privadas do ensino superior entre 2014 e 2018 com atributos referentes aos professores como a idade, sexo, se é o seu primeiro emprego e número de meses empregado. Utilizaram também informações sobre o trabalho, como o número de alunos por aula e número de horas trabalhadas.

Lima et al. (2020) recorreram a uma base de dados de uma instituição de seguranças militares entre os anos 2012 e 2017, utilizaram informações demográficas dos trabalhadores, o local de trabalho, a sua posição na empresa e o número de dias ausente durante o ano.

2.1.3 Métodos

Rasheed (2021) utilizou vários algoritmos de classificação como Naive Bayes, regressão linear, k-nearest neighbor e random forest para perceber a relação dos atributos com o resultado final dos diferentes modelos, concluindo assim a importância de cada atributo. Jensen & Sun (2013) utilizaram diferentes modelos estatísticos como distribuição binomial,

poisson e binomial negativa para perceber quais os atributos mais importantes e que contêm mais informação para prever as ausências não planejadas. Alguns dos atributos relevantes encontrados para explicar o aumento de ausências foram a existência de tempestades, os turnos dos trabalhadores neste caso nos turnos mais tardios, os meses entre Janeiro e Abril e o mês de Dezembro.

Lima et al. (2020) utilizaram modelos como Support Vector Machine (SVM), redes neuronais simples, redes recorrentes e redes Long Short-Term Memory com o objetivo de prever o número de dias ausentes durante um ano por seguranças militares. Neste estudo os melhores resultados foram produzidos por redes neuronais, com uma precisão de 78%, e os modelos com piores resultados foram modelos de SVM.

Jayme et al. (2021) com o mesmo objetivo de prever o número de dias ausentes durante um ano, utilizaram métodos de redes neuronais, SVM e Random Forest, classificando o resultado entre 5 classes: 0 dias ausentes, até 2 dias, de 2 a 5 dias, de 5 a 14 dias e mais de 14 dias ausentes. Obtiveram os melhores resultados em redes neuronais com uma precisão de 77%.

Nath et al. (2022) tinham como objetivo prever o número de horas ausentes num mês, dividindo os resultados por três classes diferentes: uma classe sem ausências, ausente entre 1 a 15 horas por mês e entre 16 e 120 horas por mês. Utilizaram redes neuronais, SVM e Random Forest, tendo sido os melhores resultados obtidos por SVM com uma precisão de 0.915 e um recall de 0.831.

Wahid et al. (2019) utilizaram vários modelos de árvore para prever o número de horas ausentes dividindo os dados por quatro classes: 0 horas ausentes, até 8 horas, entre 8 a 40 horas e mais de 40 horas. Entre os modelos utilizados, Decision Tree, Gradient Boosted Tree, Random Forest e Tree Ensemble o que obteve melhores resultados foi o Gradient Boosted Tree com uma precisão de 0.824.

Araujo et al. (2019) e Martiniano et al. (2012) ambos recorreram a modelos de fuzzy neural network para prever o número de horas ausentes, e concluíram este tipo de modelo é uma excelente ferramenta para auxiliar a tomada de decisão.

Após a revisão de literatura, comparando todos os modelos aplicados e resultados dos mesmo, referidos anteriormente, decidiu-se aplicar nesta dissertação as redes neuronais, nomeadamente as redes recorrentes, para tentar resolver o problema em questão. Visto que as mesmas foram utilizadas para resolver problemas semelhantes obtendo bons resultados. Estas redes também são ideais para trabalhar com os dados temporais que serão utilizados nesta dissertação.

2.2 Métodos

2.2.1 Auto-Regressão

Um modelo de Auto-Regressão (AR) é uma técnica estatística utilizada na análise de séries temporais, que são conjuntos de observações ordenadas ao longo do tempo. Esse modelo pressupõe que o valor de uma variável no tempo atual está linearmente relacionado aos valores anteriores, sendo influenciado por um termo de erro. Ou seja, a previsão para o próximo ponto na série temporal é uma combinação linear dos pontos anteriores, acrescida de um componente de erro estocástico. A estrutura básica de um modelo AR é expressa pela equação:

$$X_t = c + \phi_1 X_{t-1} + \phi_2 X_{t-2} + \dots + \phi_p X_{t-p} + \varepsilon_t \quad (2.1)$$

Onde:

- X_t : Valor da variável no tempo t
- c : Constante
- $\phi_1, \phi_2, \dots, \phi_p$: Coeficientes de auto-regressão
- $X_{t-1}, X_{t-2}, \dots, X_{t-p}$: Valores anteriores da variável
- ε_t : Termo de erro no tempo t

Os modelos AR são utilizados na previsão dados sequenciais, como series temporais. A identificação da ordem correta do modelo e a estimação dos parâmetros são passos cruciais no processo de implementação de um modelo de auto-regressão eficaz.

2.2.2 Rede Neuronal

Uma rede neuronal consiste numa estrutura composta por neurónios e ligações que se organizam em camadas. A rede neuronal artificial estabelece relações entre um conjunto de variáveis de entrada e um conjunto de variáveis de saída, isto é, associa os valores das variáveis de input aos respetivos valores de output [17]. Na figura 2.1 está representada uma estrutura simplificada de uma rede neuronal.

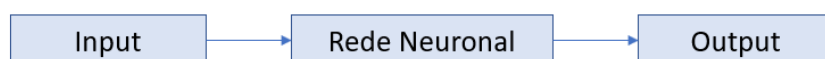


Figura 2.1: Estrutura de uma Rede Neuronal

Cada neurónio da rede neuronal recebe um conjunto de inputs. Estes inputs são combinados utilizando pesos específicos que determinam o contributo que cada input tem para o cálculo do output. De seguida é aplicada uma função de ativação ao output.

Estes outputs são utilizados como inputs nas camadas seguintes da rede. Se considerarmos uma rede feedforward com uma única camada, ou seja, nenhuma camada oculta e uma única saída, esta rede é semelhante a uma regressão linear. Onde a um input é multiplicado um valor, neste caso um peso, e somado um bias.

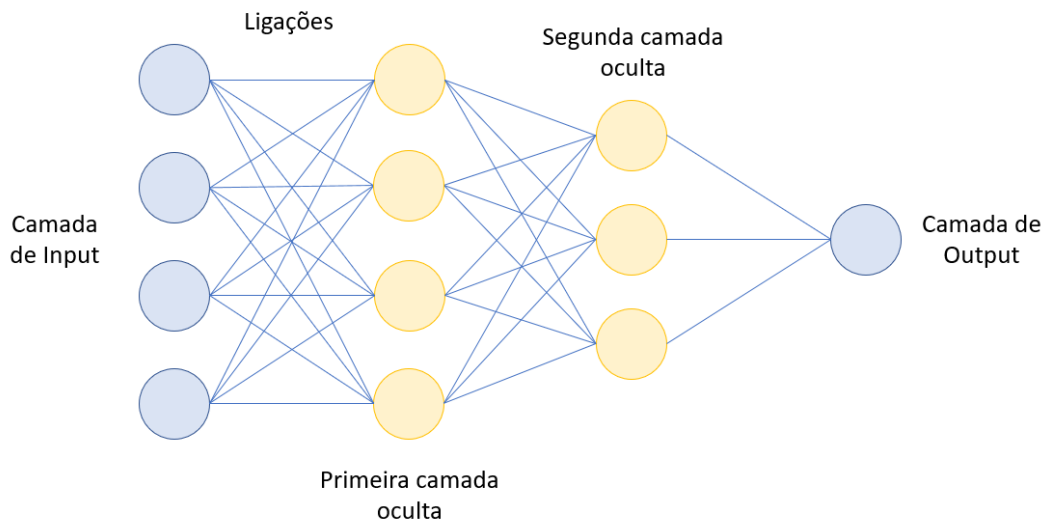


Figura 2.2: Exemplo de uma Rede Neuronal

As redes neuronais são treinadas utilizando um conjunto de dados de treino que inclui exemplos de inputs e outputs esperados. Durante o treino, os pesos das ligações entre os neurónios são ajustados de forma iterativa, com o objetivo de minimizar o erro entre o output produzido pela rede e o output esperado. Para isso, é utilizado um algoritmo de otimização, como o gradiente descendente, que ajusta os pesos de forma a minimizar o erro a cada passo. Uma vez treinada, a rede neural pode ser utilizada para fazer previsões ou tomar decisões com base em novos dados de entrada utilizando as camadas e os pesos que foram ajustados durante o treino.

Para o treino de uma rede neuronal existem dois termos utilizados que controlam como é que os dados são apresentados ao modelo. Estes termos são *batches* e *épocas*. Os *batches* são subconjuntos dos dados de entrada totais do treino, é possível alterar a quantidade de *batches* utilizados de forma a otimizar a velocidade do modelo. Em vez do modelo calcular os pesos necessários a cada observação, estes pesos apenas são calculados a cada subconjunto, ou seja, a cada *batch* de modo a melhorar a performance do mesmo. Uma *época* é uma única passagem completa por todas as observações do conjunto de entrada do treino. Ou seja, o número de *épocas* escolhido representa o número de vezes que o modelo observa o conjunto de entrada completo.

Existem vários tipos diferentes de redes neuronais, cada uma com as suas próprias características e propriedades. Alguns exemplos são:

- **Redes FeedForward:** Estas redes têm camadas sequenciais de neurónios, onde os dados passam pelas camadas da rede num único sentido.

- **Redes recorrentes:** Estas redes têm ligações recorrentes que permitem que as camadas iniciais recebam informações das camadas seguintes, o que permite que armazenem e processem essas informações ao longo do tempo. São normalmente utilizadas para processar séries temporais.
- **Redes convolucionais:** Estas redes são especialmente eficientes para processar imagens. Utilizam camadas de filtros que permitem que a rede aprenda padrões em diferentes regiões da imagem.

2.2.2.1 Funções de ativação

As funções de ativação são utilizadas entre camadas nas redes neuronais, estas funções são os interruptores numa rede neuronal. O seu objetivo é decidir se um neurónio deve ser ativado ou não com base no valor de entrada. As funções de ativação introduzem a não linearidades nas redes, permitindo que as mesmas aprendam padrões mais complexos nos dados. Sem estas funções uma rede neuronal seria apenas uma combinação linear simples [16].

Sigmoid

A sigmoid, representada na figura 2.2, é um dos tipos de funções de ativação, normalmente utilizada em problemas binários, visto que a mesma, dependendo do valor x , devolve um valor entre 0 e 1, podendo ser considerado uma probabilidade.

$$f(x) = \frac{e^x}{1 + e^x} \quad (2.2)$$

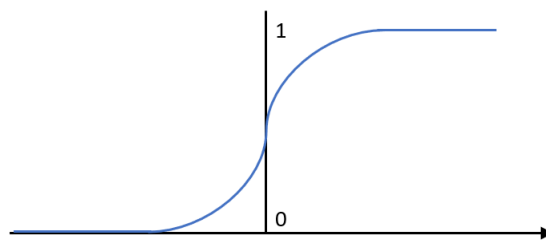


Figura 2.3: Função de ativação Sigmoid

ReLU

A função de ativação Rectified Linear Unit (ReLU), é uma função utilizada principalmente nas camadas ocultas, esta função devolve valores de 0 a infinito. No caso do valor de entrada ser negativo a função devolve 0, se o valor de entrada for positivo devolve o valor de entrada.

$$f(x) = \begin{cases} x, & \text{se } x > 0 \\ 0, & \text{se } x \leq 0 \end{cases} \quad (2.3)$$

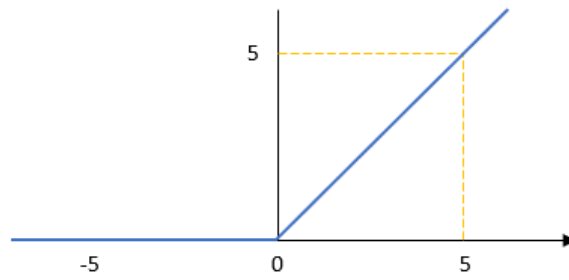


Figura 2.4: Função de ativação ReLU

Tanh

A função de ativação Tangente Hiperbólica (Tanh) é semelhante à sigmoid, no entanto retorna valores entre -1 e 1. Esta função é utilizada normalmente para problemas que envolvem padrões com polaridade, ou seja, quando os padrões nos dados têm uma orientação clara em direção a valores positivos ou negativos.

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.4)$$

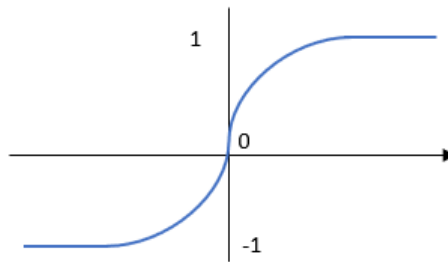


Figura 2.5: Função de ativação Tanh

2.2.2.2 Rede FeedForward

As redes FeedForward são o tipo mais simples de redes neuronais. Uma das características destas redes é o facto dos dados só seguirem um sentido, ou seja, a camada de input da rede recebe os dados de entrada, sendo processados e transmitidos para a próxima camada, chamada camada oculta. Este processo é repetido sequencialmente, camada após camada, até que os dados cheguem à camada de output, onde são produzidos os resultados da rede. Em cada camada existem um ou mais neurónios e a informação entre cada um é transmitida apenas para os neurónios da próxima camada [22].

Percetrão

Para ilustrar melhor o funcionamento de uma rede neuronal será introduzido o **Percetrão** que é uma rede composta por um só neurónio. Esta rede é composta pelo vetor de input, um neurónio e pelo output booleano (0 ou 1).

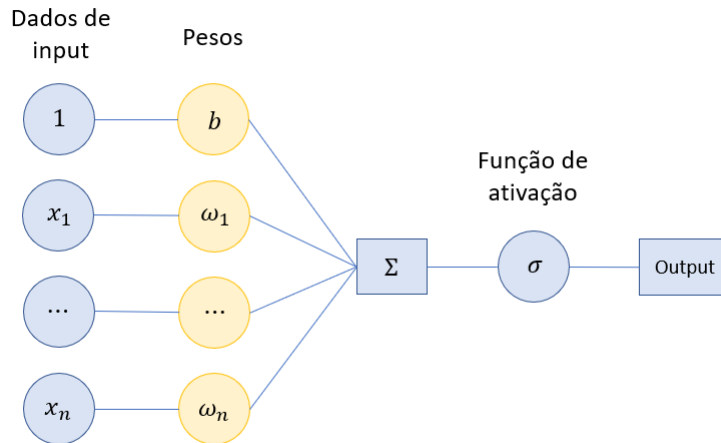


Figura 2.6: Percetrão - rede neuronal composta por apenas um neurónio

Consideremos o exemplo de um conjunto de pontos (x, y) e uma reta de regressão linear $Y = \beta x + b$ cujos parâmetros β e b são desconhecidos. Esta relação linear terá como objetivo fazer a classificação binária dos pontos, atribuindo o valor 1 se os pontos estiverem acima da reta e o valor 0 se estiverem abaixo. O objetivo do percetrão será ajustar os parâmetros (β e b) de modo a que a reta consiga separar, o melhor possível, os diferentes pontos.

Começa-se por fazer o produto escalar entre os pesos e os dados de input e adiciona-se um bias b .

$$z = \sum_{i=1}^n x_i \omega_i + \mathbf{b} = \boldsymbol{\omega}^T \mathbf{x} + \mathbf{b} \quad (2.5)$$

Por conveniência o bias será incorporado no vetor dos pesos como ω_0 e o vetor de inputs x será estendido para incluir mais uma dimensão, x_0 , com a constante 1 obtendo-se:

$$z = \sum_{i=0}^n x_i \omega_i = \boldsymbol{\omega}^T \mathbf{x} \quad , \quad \omega_0 = b \quad (2.6)$$

De seguida este valor será passado a uma função de ativação. Neste caso foi escolhida a função *Sigmoid* que retorna valores entre 0 e 1.

$$f(z) = \frac{e^z}{1 + e^z} \Leftrightarrow f(\boldsymbol{\omega}^T \mathbf{x}) = \frac{e^{\boldsymbol{\omega}^T \mathbf{x}}}{1 + e^{\boldsymbol{\omega}^T \mathbf{x}}} \quad (2.7)$$

Deste modo pode-se considerar que se a função de ativação retornar um valor superior a 0.5, o neurónio é ativado e o output será 1. Caso contrário, se o valor emitido for menor que 0.5, considera-se que o neurónio não foi ativado e valor de output será 0.

$$\sigma(z) = \begin{cases} 1, & \text{se } f(z) > 0.5 \\ 0, & \text{c.c} \end{cases}$$

Este valor é posteriormente comparado com o valor original. De seguida para o ajuste dos pesos iremos minimizar uma função de custo e para isto podem ser usados métodos como a descida do gradiente estocástica ou o ADAM [9]. No Percetrão será minimizada a função de custo como por exemplo a função do erro quadrático médio (2.11).

No contexto das redes neuronais o número de exemplos vistos pela rede antes da atualização dos pesos dá-se o nome de *batch size* e no caso de Percetrão como se irá atualizar os pesos após cada exemplo visto, tem-se $N = 1$. A equação (2.11) será multiplicada por $\frac{1}{2}$ para futuras simplificações no cálculo das derivadas parciais:

$$L(\omega) = \frac{(y_i - \hat{y}_i)^2}{2} = \frac{(y_i - \frac{e^{\omega^T x}}{1+e^{\omega^T x}})^2}{2} \quad (2.8)$$

Se se optar pelo uso do método da descida do gradiente para atualizar os pesos, este método utiliza o gradiente da função custo para minimizar o erro. Assim é preciso fazer a derivada parcial, em ordem aos pesos, da função de custo.

$$\begin{aligned} \frac{\partial L}{\partial w_j} &= \frac{\partial}{\partial w_j} \left(\frac{(y^{(i)} - f(\omega^T x^{(i)}))^2}{2} \right) \\ &= (y^{(i)} - f(\omega^T x^{(i)})) \left(-\frac{\partial}{\partial w_j} f(\omega^T x^{(i)}) \right) \\ &= (f(\omega^T x^{(i)}) - y^{(i)}) f(\omega^T x^{(i)}) (1 - f(\omega^T x^{(i)})) \\ &= \Delta w_j. \end{aligned} \quad (2.9)$$

O resultado desta derivada representa o gradiente ($\nabla \omega$). Sabendo que o gradiente representa a direção de maior crescimento e como o objetivo é minimizar a função custo, deve ser tomada a direção contrária. Assim para atualizar os pesos tem-se que:

$$\omega_{novo} = \omega_{antigo} - \eta \nabla \omega. \quad (2.10)$$

tal que η é a taxa de aprendizagem, podendo ser fixa ou variável. Esta taxa irá determinar o quão rápido a rede irá aprender. A condição de paragem deste método, isto é, quando se deve parar de atualizar os pesos pode ser definido por um número de iterações ou quando a diferença entre as últimas estimativas dos pesos for muito baixa.

Concluindo, numa rede neuronal cada neurónio pode ser interpretado como um percetrão. Consequentemente uma rede FeedForward é um conjunto de percetrões organizados em camadas, recebendo como inputs os outputs das camadas anteriores, atualizando os respetivos pesos e emitindo os correspondentes outputs.

2.2.2.3 Rede Convolutiva

As Convolutional Neural Network (CNN) são um tipo de redes neuronais especializadas para o processamento de dados bidimensionais, como imagens e vídeos. As CNN incluem camadas convolutivas, que são responsáveis por aprender padrões e características dos dados de entrada [2]. Estas camadas aplicam filtros convolucionais aos dados de forma a detetar características relevantes dos mesmos. Outras camadas utilizadas nas redes convolucionais são as camadas de *Pooling*, estas camadas são responsáveis por reduzir a dimensionalidade dos dados, preservando as características mais importantes dos mesmos. Normalmente no fim de uma rede convolutiva existe também uma camada *Fully Connected* para realizar a classificação final dos dados de entrada. Na figura 2.7 é possível ver um exemplo simplificado de uma rede neuronal convolutiva.

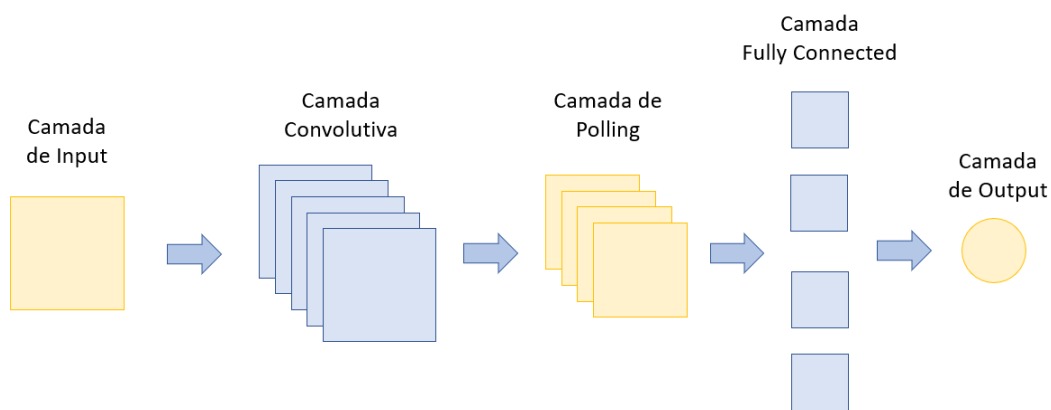


Figura 2.7: Exemplo de uma Rede Neuronal Convolutiva

2.2.2.4 Rede Recorrente

As Recurrent Neural Network (RNN) são um tipo de redes neuronais adequadas para processar dados sequenciais, como, por exemplo séries temporais. As RNN contêm uma característica de "memória" que permite lembrar e utilizar informações de etapas temporais anteriores no processamento dos dados de entrada, tornando-se assim eficazes para tarefas que exigem compreender o contexto do problema ou em que exista uma dependência entre eventos ao longo de tempo [20].

As redes neuronais recorrentes utilizam ligações de feedback. Ao contrário das redes FeedForward, estas redes permitem ter ligações a camadas ocultas anteriores ou até entre si. Para treinar este tipo de rede é utilizado um algoritmo de Backpropagation Through Time (BPTT), que é específico para dados sequenciais [21].

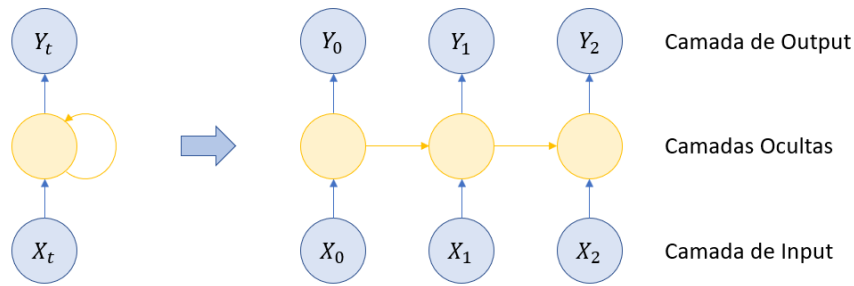


Figura 2.8: Exemplo de uma Rede Neuronal Recorrente

LSTM

Long Short-Term Memory (LSTM) é uma variação de redes neurais recorrentes que foi desenhada para lidar com o problema da perda de informações ao longo de sequências longas. Uma das características das redes neurais recorrentes é possuírem uma memória de curto prazo para guardar informações anteriores, no entanto, esta capacidade diminui muito rapidamente para sequências mais longas [20].

Para contornar este problema, as redes LSTM incluem uma célula de memória adicional para controlar o que deve ser guardado na memória de curto prazo e o que deve ser descartado. Esta célula é controlada por três portas que utilizam uma função de ativação sigmóid para produzir um valor entre 0 e 1, funcionando como um factor de decisão [20].

- Porta de esquecimento - A porta de esquecimento é um mecanismo que determina que informação deve ser esquecida ou guardada no estado da célula de memória.
- Porta de entrada - A porta de entrada determina a informação que deve ser adicionada ao estado da célula de memória
- Porta de saída - A porta de saída determina a informação que deve ser utilizada do estado da célula para calcular o output do modelo.

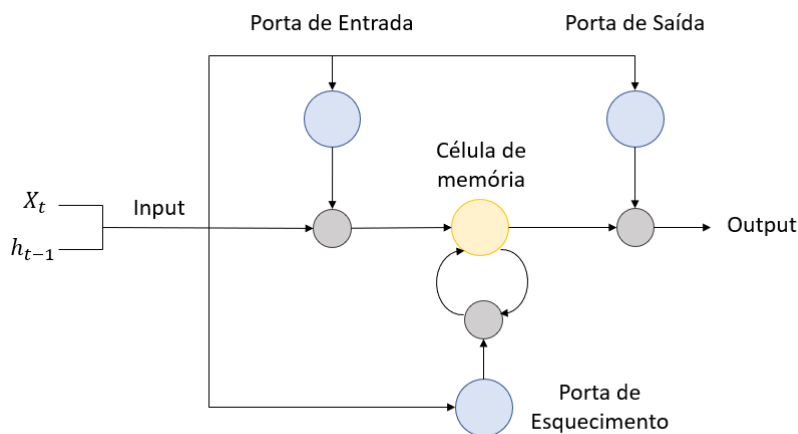


Figura 2.9: Exemplo de uma Rede LSTM

2.2.2.5 Transformers

As redes Transformers, tal como as redes LSTM, têm um bom desempenho com dados sequenciais. A principal diferença das redes Transformers é a camada de atenção, que permite que o modelo pondere a importância de diferentes partes do conjunto de entrada durante o processamento. Esta camada de atenção calcula pesos para cada elemento do conjunto de entrada, independentemente da distância entre eles, possibilitando assim um melhor desempenho com dados sequenciais complexos [24].

Geralmente dividem-se as redes Transformers em duas etapas, a codificação e decodificação. A codificação é a etapa onde são processados os dados de entrada, começando com uma codificação da posição dos elementos de entrada, seguido pela camada de atenção referida anteriormente. Depois da camada de atenção o conjunto de entrada passa por uma ou mais camadas de redes FeedForward.

A etapa de decodificação é onde são processados os dados de saída contendo uma arquitetura semelhante ao codificador. Adicionalmente na decodificação existe também a camada de atenção cruzada. Esta camada é responsável por calcular os pesos de atenção entre os dados de entrada e os dados de saída.

O final de uma rede Transformer varia dependendo da tarefa e das especificações da camada de saída. Por exemplo, se os dados de saída esperados forem divididos em diferentes classes, é possível utilizar uma função de ativação como a Softmax de forma a obter a previsão para cada classe. Na figura 2.10, é possível observar uma representação simplificada de uma rede Transformer.

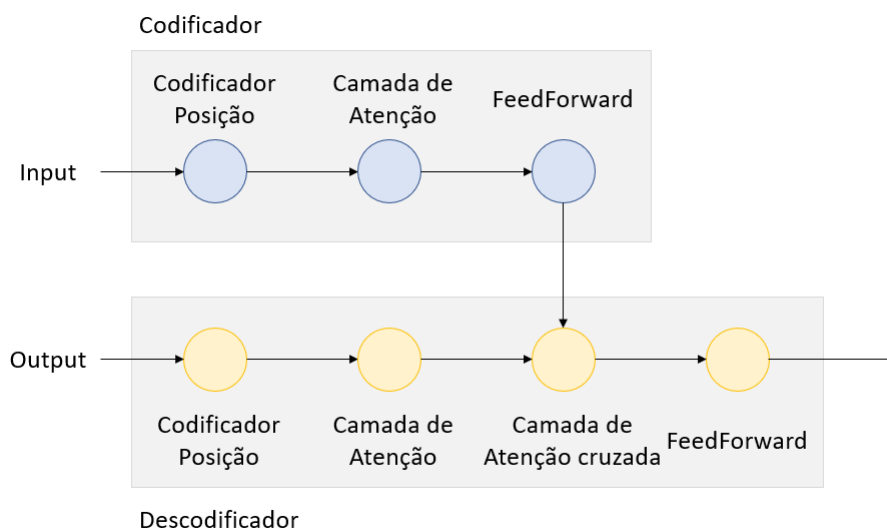


Figura 2.10: Exemplo de uma Rede Transformer

2.2.2.6 *Underfitting e Overfitting*

Para a utilização de redes neuronais, é essencial dividir os dados em dois conjuntos distintos: conjunto de treino e conjunto de teste. O conjunto de treino inclui os dados usados para o processo de aprendizagem. Com estes dados o modelo irá aprender a estabelecer as relações necessárias entre o input e o output pretendido. Este conjunto pode ser novamente subdividido de forma a obter um conjunto de validação que terá o objetivo de avaliar a aprendizagem do modelo enquanto este treina. O conjunto de validação permite-nos retirar informações sobre o desempenho do modelo, sendo muitas vezes utilizado para evitar o treino excessivo do modelo que não só prejudica o desempenho como em certos casos tem um elevado custo computacional. Por fim, o conjunto de teste é composto pelos dados que serão usados para testar o desempenho do modelo com novos dados, isto é, com dados que não foram usados no conjunto de treino nem de validação. Na figura 3.1 observa-se um exemplo da divisão do conjunto de dados.

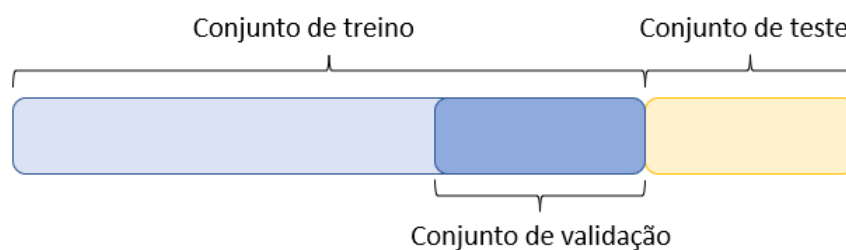


Figura 2.11: Divisão dos dados

Para considerar que um modelo está bem ajustado, os seguintes aspetos têm de ser verificados:

- O desempenho do modelo tem de estar no limiar desejado.
- O nível de erro obtido no conjunto de treino deve ser baixo.
- A diferença entre o erro obtido no conjunto de treino e o erro obtido no conjunto de teste deve ser baixo.

Estes aspetos constituem os principais desafios para a criação de um modelo que esteja bem ajustado e que seja eficaz. Será dada especial atenção aos últimos dois pontos mencionados, uma vez que estão relacionados dois desafios bastante conhecidos das redes neuronais: *Underfitting* e *Overfitting*. Começando pelo segundo ponto o método deve apresentar bons resultados nos dados do conjunto de treino e consequentemente ter um erro baixo. Quando isto não acontece estamos perante o *Underfitting*. Tomando especial atenção ao terceiro ponto, a diferença entre o erro obtido no conjunto de treino e o obtido no conjunto de teste deve ser baixa, isto é, o modelo deve ser capaz de produzir bons resultados com os dados que utilizou para aprender, bem como com dados nunca

antes vistos. Quando isto não acontece, isto é, quando o erro no conjunto de treino é bastante inferior ao erro no conjunto de teste estamos perante o *Overfitting*, significando que o modelo ajustou-se demasiado aos dados do treino, perdendo a capacidade para generalizar para outros dados.

Existem algumas características das redes neuronais que podem ser ajustadas de modo a amenizar estes desafios. Em situações de *Underfitting* podem ser aumentadas o número de *épocas* que o modelo deve treinar, o número de camadas e o número de neurónios em cada camada. Em situações de *Overfitting* pode-se diminuir o número de *épocas* que o modelo deve treinar, reduzir o número de camadas e/ou o número de neurónios em cada camada bem como adicionar regularizadores (Dropout, L1, L2, etc.).

2.3 Métricas

2.3.1 Erro quadrático médio

O erro quadrático médio, Mean Squared Error (MSE), é uma medida que calcula a diferença entre os valores previstos por um modelo (\hat{y}_i) e os valores reais (y_i). Utiliza a diferença média quadrada entre os valores observados e previstos. Quanto menor for o MSE, melhor o modelo se ajusta aos dados.

$$MSE = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n} \quad (2.11)$$

2.3.2 Erro médio absoluto

O erro médio absoluto, Mean Absolute Error (MAE), é uma medida semelhante ao MSE, no entanto, esta medida oferece uma informação mais compreensiva, pois o erro está na mesma unidade de grandeza que o output do modelo. O valor dado por esta medida é a média da diferença entre todas as previsões do modelo e os valores reais. O MAE é também menos sensível aos outliers que existem na previsão.

$$MAE = \frac{\sum_{i=1}^n |y_i - \hat{y}_i|}{n} \quad (2.12)$$

2.3.3 Erro médio absoluto percentual

O erro médio absoluto percentual, Mean Absolute Percentage Error (MAPE), é similar ao erro médio absoluto, utilizando a percentagem do erro médio absoluto. O MAPE é útil para conhecer a magnitude do erro do valor previsto pelo modelo.

$$MAPE = \frac{\sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right|}{n} \times 100\% \quad (2.13)$$

2.3.4 Precisão e Recall

A Precisão e o Recall são métricas utilizadas para avaliar a qualidade de um modelo de classificação com duas classes. Considerando que uma das classes é 0 (negativa) e outra classe é 1 (positiva) temos quatro conceitos:

- Verdadeiro Positivo (VP) - Classificação do modelo positiva, classificação real positiva
- Verdadeiro Negativo (VN) - Classificação do modelo negativa, classificação real negativa
- Falso Positivo (FP) - Classificação do modelo positiva, classificação real negativa
- Falso Negativo (FN) - Classificação do modelo negativa, classificação real positiva

A Precisão mede a fração de previsões positivas que são realmente corretas. O resultado é um valor que varia entre 0 e 1, onde 1 significa que todas as previsões do modelo positivas estão corretas.

$$Precisão = \frac{VP}{VP + FP} \quad (2.14)$$

O Recall mede a fração de classificações positivas que são corretamente identificadas pelo modelo. O resultado é um valor que varia entre 0 e 1, onde 1 significa que todos os valores da classe positiva foram corretamente classificados pelo modelo.

$$Recall = \frac{VP}{VP + FN} \quad (2.15)$$

TRABALHO REALIZADO

3.1 Descrição da Base de Dados

A base de dados disponibilizada pela SISCOG consiste em dados históricos de um operador ferroviário no norte da Europa com informações biográficas dos tripulantes (maquinistas e revisores) e registos de atividade realizada entre 2009 e 2021. Esta base de dados está dividida em três conjuntos:

- Short-Term - Registo de trabalho planeado feito com pouca antecedência face ao dia da operação.
- Long-Term - Registo de trabalho planeado feito com grande antecedência face ao dia da operação.
- Workdone - Registo de trabalho e ausências realizados.

Em cada um destes conjuntos existem 4 tabelas, sobre os trabalhadores, os trabalhos planeados e as tarefas, com as relações que se podem ver na figura 3.1. Adicionalmente no conjunto de Workdone, existem também a tabela de ABSENCE, que contém a informação sobre as ausências que se realizaram.



Figura 3.1: Diagrama do dataset

3.1. DESCRIÇÃO DA BASE DE DADOS

A SISCOG disponibilizou também alguns atributos extraídos desta base de dados com as informações dos maquinistas, dos trabalhos e ausências, onde cada entrada corresponde a um dia de trabalho de um tripulante.

Após um estudo prévio realizado na SISCOG [14] foi possível concluir que os anos de 2009 até Abril de 2013 são anos com dados inconsistentes. Como tal, não foram utilizados para treinar os modelos. A partir do ano de 2020 também se detetou uma anomalia em relação aos dados devido ao COVID-19, pelo que também não foram utilizados. Na tabela 3.1 apresentam-se todos os atributos disponíveis dos dados extraídos desta base de dados.

Nº	Atributos	Descrição
1	WAS_ABSENT_YESTERDAY	Ausente no dia anterior
2	HAD_DAY_OFF_YESTERDAY	Folgou no dia anterior
3	HAS_DAY_OFF_TOMORROW	Folga no dia seguinte
4	WAS_IN_VACATION_YESTERDAY	Esteve de férias no dia anterior
5	IS_IN_VACATION_TOMORROW	Férias no dia seguinte
6	DAY_OF_WEEK	Dia da semana (0 a 6)
7	MONTH	Mês
8	SENIORITY	Número de anos como trabalhador na empresa
9	AGE	Idade do trabalhador
10	DUTY_DURATION	Duração do turno em segundos
11	DAYS_SINCE_LAST_ABSENCE	Número de dias desde a última ausência
12	AVERAGE_ABSENCES	Média de ausências durante o ano
13	LAST_YEAR_ABSENCES	Número de ausências no ano anterior
14	ABSENCE_HISTORY_YEARS	Média de ausências nos últimos anos
15	DUTY_TIMING_CATEGORY	Altura do dia do trabalho (manhã, tarde, noite)
16	DUTY_DURATION_CATEGORY	Duração do trabalho com categoria (curto, médio, longo)
17	HOLIDAY	Feriado
18	HOLIDAY_YESTERDAY	Feriado no dia anterior
19	PERSONNEL_TYPE	Tipo de tripulante (maquinista ou revisor)

Tabela 3.1: Atributos disponíveis na base da dados

Para o treino dos modelos foram utilizados os dados entre 2014 a 2018 e para teste foram utilizados os dados do ano de 2019. Previamente ao treino foi realizado um controlo de qualidade em todos os atributos, procurando atributos mal classificados ou vazios, removendo os mesmos. No caso do atributo DUTY_DURATION, o mesmo continha demasiados valores vazios. Assim, foi aplicado um método diferente: em vez de remover estes valores, foram alterados para o valor -1. Esta alteração tinha como objetivo preservar o número total de observações mantendo a diferença destes dados em relação aos dados existentes.

Antes das alterações feitas aos atributos, entre o ano de 2014 e 2019, para os maquinistas, existiam 1.354.121 observações, das quais 108.383 (8%) eram ausências. Após remover os valores vazios ficaram 1.160.445 observações, das quais 107.798 (9%) são ausências.

Uma das alterações feitas aos atributos foi a normalização dos dados numéricos, não categóricos, de forma a obter uma escala comum em cada um dos campos apresentados aos modelos.

Outra alteração foi aplicar o one-hot encoding aos atributos categóricos com o objetivo de melhorar o desempenho do modelo com estes atributos. A ideia por trás do one-hot encoding é criar uma nova representação binária para cada categoria única presente na variável categórica. Por exemplo no atributo dia da semana existem 7 valores onde 1 representa segunda-feira e 7 domingo. Com o one-hot encoding este atributo transforma-se em 7 atributos diferentes cada um representando cada dia da semana com um valor binário.

Após todas as alterações referidas anteriormente obteve-se os atributos da tabela 3.1 devidamente preparados para o treino e teste dos modelos.

3.1.1 Dados de entrada

Com o objetivo de utilizar os dados da base de dados num modelo, é necessário transformá-los em dados de entrada com o formato esperado pelo modelo a ser utilizado. Neste caso existem três tipos de dados de entrada: dados agregados, desagregados e sequenciais. Os dados agregados são utilizados nos modelos mais simples como regressão linear e feedforward. Estes dados são agrupados pela data de forma a obter um conjunto de informações de todos os trabalhadores para um certo dia, como é possível observar na figura 3.2. Neste tipo de dados não existem informações singulares de cada trabalhador, apenas uma soma ou média de todos os trabalhadores e turnos.

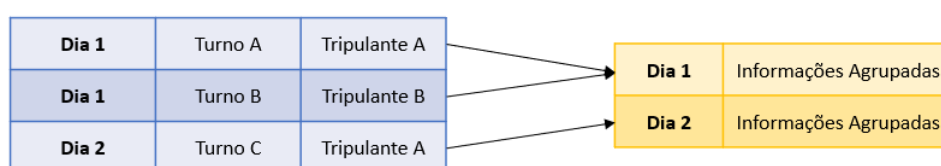


Figura 3.2: Exemplo do conjunto de dados agregados

Os dados desagregados, ao contrário dos dados anteriores, são um conjunto de informações individuais sobre cada tripulante para cada turno.

Por fim, existem também os dados sequenciais utilizados em redes recorrentes, como as LSTM. Estes dados são semelhantes aos dados desagregados onde cada informação é específica de cada trabalhador. No entanto, são agrupados pelo tripulante obtendo um conjunto de informações individuais do mesmo ordenado cronologicamente. Ou seja, cada tripulante terá um conjunto de dados contendo a informação de todos os turnos

que o mesmo realizou. Na figura 3.3 observa-se um exemplo do agrupamento dos dados sequenciais.

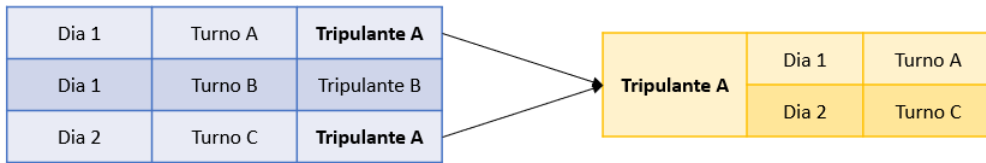


Figura 3.3: Exemplo do conjunto de dados sequenciais

3.1.2 Dados de saída

Os dados de saída são o conjunto de dados que o modelo retorna. Neste caso os dados de saída dependem dos dados de entrada. Se os dados de entrada forem agregados o modelo devolve um valor inteiro para cada dia, onde este valor é interpretado como o número total de ausências para esse mesmo dia. Na figura 3.4 está representado um exemplo do conjunto de saída para o conjunto de dados de entrada agregados.

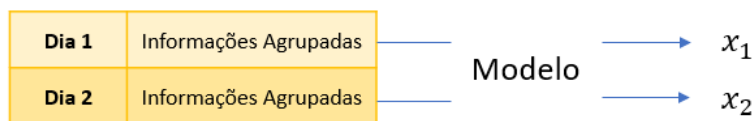


Figura 3.4: Exemplo do conjunto de dados de saída agregados

No caso dos dados de entrada serem desagregados ou sequenciais, o modelo retorna um valor entre 0 e 1, que representa a probabilidade do tripulante estar ausente. Como o objetivo é calcular o número total de tripulantes ausentes para um dia, é necessário, neste caso, agrupar estes dados de saída, por data, somando todas as ausências previstas pelo modelo. Na figura 3.5 é possível observar um exemplo para os dados de saída quando os dados de entrada são desagregados, o mesmo exemplo pode ser aplicado para os dados de entrada sequenciais.



Figura 3.5: Exemplo do conjunto de dados de saída desagregados

3.2 Modelos

Nesta secção serão apresentados todos os modelos construídos, organizados por dados de entrada. Estes modelos foram treinados e testados apenas utilizando os dados dos maquinistas. Com base nos resultados dos modelos utilizados nos dados dos maquinistas, será utilizado o modelo com o melhor desempenho para aplicar aos dados dos revisores, ou seja, treinar e testar este modelo com os dados referentes aos dados dos revisores, abordado na secção 3.5. Na secção 3.4 pretende-se aumentar a antecedência das previsões. Finalmente na secção 3.6 irá ser dividido o número de ausências por base operacional.

Todos os modelos como métricas retiradas foram desenvolvidos em *Python*. Para a criação dos modelos foram utilizadas as bibliotecas de *Tensorflow* e *Keras*. Para as métricas e resultados foram utilizadas as bibliotecas *Pandas* e *scikit-learn*.

3.2.1 Dados agregados

Para o primeiro modelo foram agrupados todos os dados disponíveis por data, obtendo-se assim variáveis agrupadas como o número de ausências por dia e o número de trabalhadores planeados.

3.2.1.1 Auto-regressão

De forma a criar um ponto de referência para os modelos mais avançados, começou-se por utilizar um modelo simples de AR de $p = 1$, com a informação do número de ausências do dia anterior com o propósito de prever o número de ausências do dia seguinte. Todos os modelos foram treinados com dados dos trabalhadores entre os anos 2014 e 2018. Para os testes foram utilizadas as informações do ano 2019. Na figura 3.6 é possível visualizar a previsão do modelo de auto-regressão para o ano de 2019.

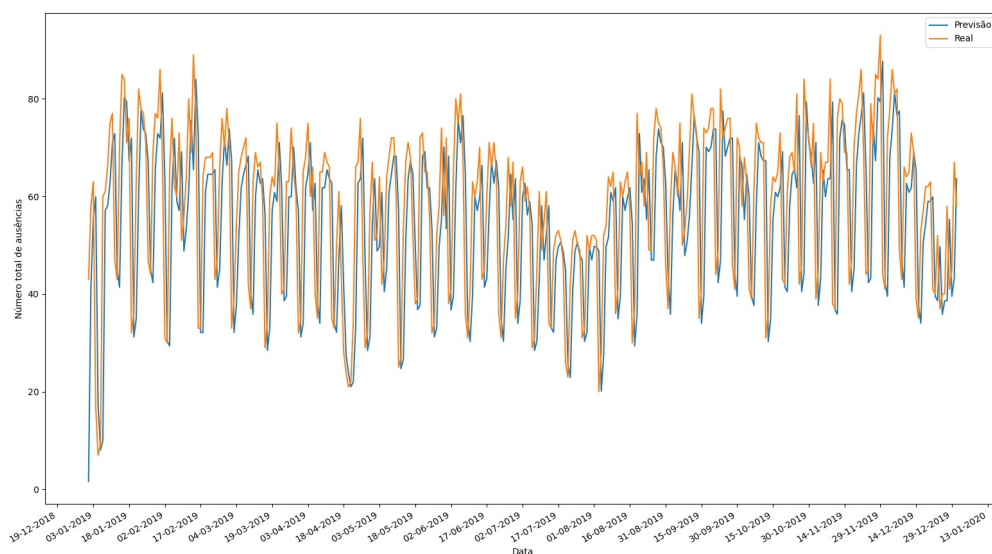


Figura 3.6: Previsão do modelo de auto-regressão para o ano 2019

De forma a comparar os modelos foram utilizadas três métricas nos resultados do teste, o MSE, o MAE e o MAPE. Na tabela 3.2 estão os resultados de cada métrica para o modelo de auto-regressão.

MSE	MAE	MAPE
263.39	12.27	24.85%

Tabela 3.2: Erro preditivo referente ao modelo de auto-regressão no ano 2019

3.2.1.2 Feedforward

De seguida foram criados vários modelos de redes neuronais feedforward, adicionando mais complexidade ao modelo e alterando os dados de entrada. O modelo que apresenta melhores resultados, como é possível ver na tabela 3.3, utiliza apenas o número de trabalhadores ausentes do dia anterior e o número de pessoas planeadas para trabalhar no dia de calendário para o qual é feita a previsão.

MSE	MAE	MAPE
60.61	6.27	13.03%

Tabela 3.3: Erro preditivo referente ao modelo feedforward no ano 2019

Comparando os resultados da tabela 3.3 com o modelo de auto-regressão, existe uma diminuição do erro, obtendo-se um erro médio absoluto de 6.27 pessoas, ou seja, em média a previsão de um dia tem um erro cerca de 6 pessoas. No entanto, este modelo, devido aos seus dados de entrada, pode apenas ser utilizado para prever o número de pessoas ausentes para o dia seguinte. Assim, foi criado um novo modelo que, em vez de utilizar o número de pessoas ausentes do dia anterior, utiliza a informação agregada ao longo de vários anos anteriores do número de pessoas ausentes para o mesmo dia do ano. Para isso foi necessário fazer o rácio entre o número médio de trabalhadores ausentes e o número de pessoas planeadas para cada dia. Obtiveram-se assim os resultados da tabela 3.4.

MSE	MAE	MAPE
58.68	6.12	12.53%

Tabela 3.4: Erro preditivo referente ao modelo feedforward baseado no rácio no ano 2019

Embora estes resultados não melhorem significativamente o modelo anterior, existe agora a possibilidade de prever o número de ausências não planeadas sem limite de tempo prévios a estas previsões, ou seja, é possível prever ausências com um intervalo de um ano ou mais. Para exemplificar foi simulada uma previsão para o ano 2019 que se pode ver na figura 3.7. No entanto, é importante lembrar que como estes dados dependem

de informações dos anos anteriores se a mesma previsão foi realizada a mais de um ano de antecedência pode existir uma degradação da performance do modelo.

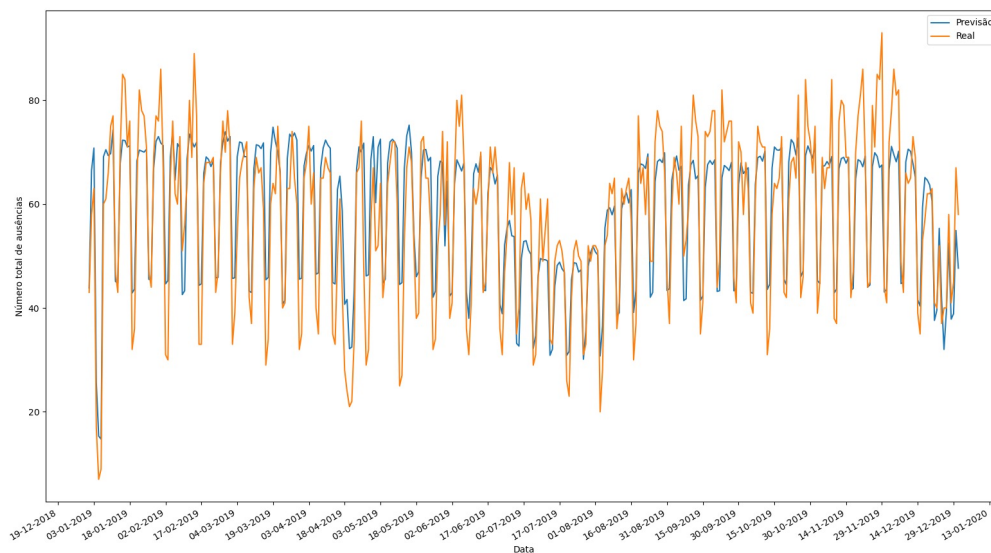


Figura 3.7: Previsão do modelo feedforward baseado no rácio para o ano 2019

Arquitetura do modelo

Para ambos os modelos feedforward com dados de entrada agregados, foi utilizada uma arquitetura simples começando com 2 camadas densas com 8 neurónios cada uma, seguidas por uma função de ativação ReLU. Terminando numa camada densa com 1 neurónio responsável por transformar o output num valor inteiro, que representa o número total de ausências para a data dos dados de entrada.

3.2.2 Dados desagregados

Foram também criados modelos utilizando os dados desagregados de forma a captar mais informação sobre os turnos planeados e os tripulantes. Ao utilizar os dados desagregados, o modelo irá prever se o tripulante irá estar ausente para um determinado dia em que é suposto ele estar a trabalhar. Como o objetivo é saber o número total de ausências, é necessário somar todas as previsões que foram consideradas ausências pelo modelo.

Um dos desafios ao utilizar os dados desagregados é a diferença entre o número de ausências e presenças no dataset, ou seja, como o número de presenças é muito superior ao número de ausências, o modelo não consegue aprender. Como referido anteriormente o número de ausências é cerca de 9% do total de observações, assim se o modelo considerar todas as observações como presenças estará 91% das vezes correto. Para evitar isto foram dados pesos a cada classe, ou seja, o modelo ao calcular o erro que existe na previsão, terá em consideração que é mais importante acertar na classe de ausências do que nas presenças. Este peso foi colocado como 90% na classe de ausência e 10% na classe de presença. Para treinar este modelo foram utilizados os atributos 1 a 13 da tabela 3.1.

Após o teste do modelo obtemos os resultados da tabela 3.5, onde a Precisão e o Recall são as métricas relacionadas com o seu output. Recorda-se que, com os dados desagregados o modelo prevê a probabilidade do tripulante estar ausente, arredondando todos os valores superiores a 0.5 para 1, indicando que o tripulante em questão estará ausente. Depois de arredondar todos os valores são somados todos os tripulantes que foram considerados como ausentes pelo modelo e calculam-se as métricas MSE, MAE e MAPE.

Precisão	Recall	MSE	MAE	MAPE
0.77	0.78	34.48	4.46	8.39%

Tabela 3.5: Erro preditivo referente ao modelo com dados desagregados no ano 2019

Para este modelo os resultados melhoraram em relação ao modelo com dados agregados, pois temos mais atributos com informações importantes para conseguir prever as ausências. No entanto, com o tipo de atributos utilizados neste modelo só é possível prever as ausências para o dia seguinte. Na figura 3.8 é possível ver a previsão do modelo para o ano de 2019.

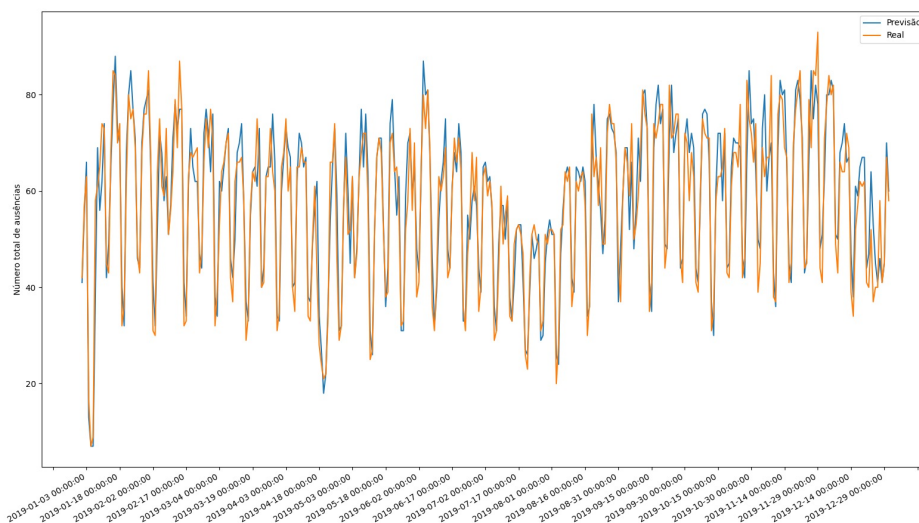


Figura 3.8: Previsão do modelo Feedforward para o ano 2019

Arquitetura do modelo

Para o modelo de dados desagregados a arquitetura utilizada foi composta por 3 camadas densas com 32, 16 e 5 neurónios, cada uma seguida por uma função de ativação ReLU. No caso das duas primeiras camadas densas depois da função de ativação aplicou-se uma camada *Dropout* com uma taxa de 0.25. Para terminar utilizou-se uma camada densa com 1 neurónio. Ao contrário do modelo desagregado os dados de saída esperados por este modelo são uma probabilidade do tripulante estar ausente. Assim aplicou-se uma função de ativação sigmoid depois da última camada densa de forma a obter um valor entre 0 e 1.

3.2.3 Dados sequenciais

3.2.3.1 LSTM

O modelo LSTM trabalha com dados sequenciais de forma a criar uma memória de informações. Assim, cada conjunto de entrada corresponde apenas a uma pessoa, com todos os dias que a mesma prestou serviço. Este conjunto de entrada é ordenado cronologicamente. Tal como no modelo de dados desagregados, o modelo LSTM irá prever para cada dia a probabilidade do tripulante estar ausente. No final são somadas todas as probabilidades para o mesmo dia obtendo assim o total de ausências de cada dia. Para este modelo foram utilizados os atributos 1 a 18 da tabela 3.1. De forma semelhante ao modelo anterior de dados desagregados, foram calculados a Precisão e o Recall através dos dados de saída e os MSE, MAE e MAPE através da soma dos mesmos. Os resultados estão descritos na tabela 3.6.

Precisão	Recall	MSE	MAE	MAPE
0.97	0.94	6.07	1.78	3.21%

Tabela 3.6: Erro preditivo referente ao modelo LSTM no ano 2019

De seguida é possível observar a figura 3.9 que contém a soma das previsões por dia dos maquinistas treinados com o modelo LSTM e os valores reais para o ano de 2019. Ao comparar este modelo com os modelos anteriores, principalmente comparando com o modelo Feedforward dos dados desagregados, tabela 3.5, é possível observar uma melhoria no desempenho, reduzindo o MAPE em menos de metade do modelo anterior.

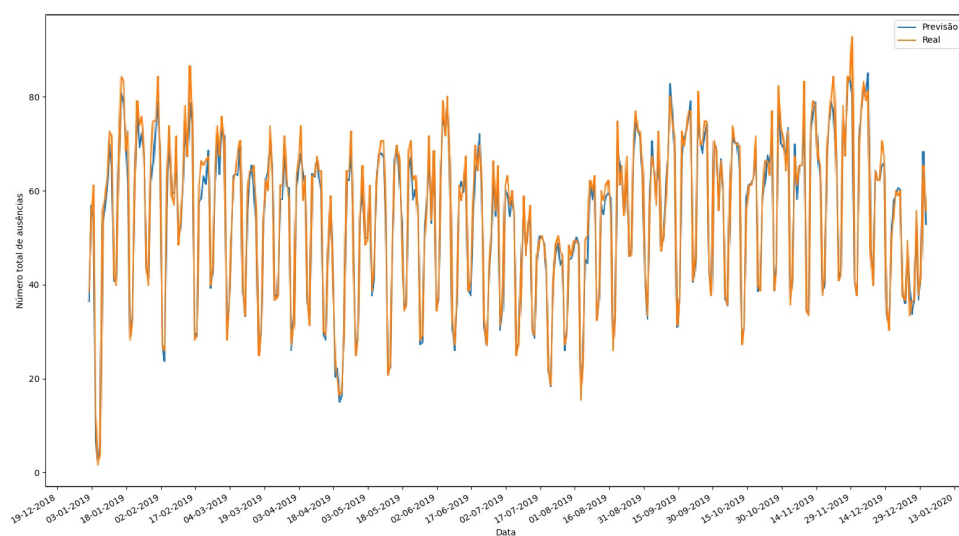


Figura 3.9: Previsão do modelo LSTM para o ano 2019

Arquitetura do modelo

O modelo LSTM é composto por:

- Uma camada convulsional com 50 filtros de 2x2 com uma função de ativação ReLU seguida por um *max pooling* com um tamanho de 2.
- Uma camada de normalização, de forma a normalizar o conjunto de dados de entrada.
- Uma camada LSTM com 35 neurónios com uma função de ativação Tanh.
- Uma camada de *dropout* com uma taxa de 0.4, de modo a evitar o *overfitting*.
- Uma camada convulsional com 25 filtros de 2x2 com uma função de ativação ReLU seguida por um *max pooling* com um tamanho de 2.
- Uma camada LSTM com 15 neurónios com uma função de ativação Tanh.
- Por fim utilizou-se um *reshape* seguido por uma função de ativação sigmoid, com o objetivo de transformar o conjunto de saída na probabilidade de o tripulante estar ausente, entre 0 e 1.

3.2.3.2 Transformers

Semelhante ao modelo LSTM, o modelo Transformers trabalha com os mesmos dados de entrada sequenciais com informações dos maquinistas para prever a ausência dos mesmos para o dia seguinte. Foram retiradas as mesmas métricas dos modelos anteriores obtendo os valores da tabela 3.7.

Precisão	Recall	MSE	MAE	MAPE
0.93	0.93	10.94	2.38	4.17%

Tabela 3.7: Erro preditivo referente ao modelo Transformers

Quando comparado com os modelos anteriores de Feedforward, os resultados do Transformers são significativamente melhores. No entanto existe uma queda de qualidade nos resultados quando comparados com os resultados do modelo LSTM, aumentando os erros preditivos de MSE, MAE e MAPE. Existe também uma diminuição da Precisão e do Recall em relação ao modelo LSTM.

Arquitetura do modelo

O modelo Transformer é composto por:

- Uma camada convulsional com 50 filtros de 2x2 com uma função de ativação ReLU seguida por um *max pooling* com um tamanho de 2.
- Uma camada de normalização, de forma a normalizar o conjunto de dados de entrada.
- Uma camada de *dropout* com uma taxa de 0.3, de modo a evitar o *overfitting*.
- Uma camada de codificação de posição com um tamanho de 2500, este tamanho está relacionado com quantidade de dias no conjunto de entrada.
- Duas camadas iguais de Transformers com uma dimensão de 64 e um número de redes Feedforward de 64.
- Para finalizar utilizou-se um *reshape* seguido por uma função de ativação sigmoid, com o objetivo de transformar o conjunto de saída na probabilidade de o tripulante estar ausente, entre 0 e 1.

3.2.4 Resultados

Após aplicar todos os modelos ao mesmo conjunto de teste e observando todos os resultados, conclui-se que os melhores modelos são o modelo LSTM e o modelo Transformers. Assim, ambos os modelos serão aplicados a várias antecedências de previsão, na secção 3.4, de forma a perceber qual o melhor modelo nas diferentes situações.

3.3 Otimização de modelos

A otimização dos modelos tem como objetivo melhorar o desempenho e eficiência dos algoritmos de aprendizagem. Esta otimização encontra-se na escolha dos parâmetros a utilizar, métricas retiradas no modelo e camadas utilizadas nas redes neuronais.

3.3.1 Dados de validação

Antes de aplicar os dados de treino a qualquer um dos modelos referidos anteriormente, foi retirado um conjunto de validação destes dados. Este conjunto é utilizado para entender qual o comportamento do modelo numa situação "real", ou seja, como este conjunto de validação não foi utilizado no treino, o modelo não se ajusta a estes dados. Assim, um conjunto de validação ajuda a identificar o *overfitting* ou *underfitting* do modelo e otimizar os parâmetros escolhidos para o mesmo. Foram retiradas aleatoriamente 10% das observações do conjunto de treino para a validação.

3.3.2 Função de perda

No âmbito do problema em questão irá ser usada a função de custo *Binary Cross-Entropy* [19] uma vez que é adequada para problemas de classificação com duas classes. Esta função é dada por:

$$L = -t \log(p) - (1 - t) \log(1 - p) \quad (3.1)$$

Para interpretar esta função pode-se tomar como exemplo um conjunto de pontos $(x, y) \in \mathbb{R}^2$ que estão separados por uma reta. Esta função pretende aproximar uma probabilidade que determinará se é mais provável o ponto estar acima ou abaixo da reta.

3.3.3 Previsões

No caso dos modelos de classificação, como referido anteriormente, estes devolvem um valor entre 0 e 1, que corresponde a uma probabilidade de um tripulante se ausentar. Quanto mais perto do valor 1 estiver maior a probabilidade de ausência.

Para calcular o valor total de ausências de um dia foram aplicados dois métodos. No primeiro método arredondaram-se todos os valores da previsão com o limite de 0.5, ou seja, todos os valores superiores a 0.5 foram arredondados para 1 e todos os valores abaixo de 0.5 para 0. De seguida somaram-se estes valores para obter o número total de ausências por dia. No segundo método foram somadas apenas as probabilidades previstas pelo modelo arredondando os valores no final de forma a calcular com um número inteiro para o total de ausências. Este segundo método apenas pode ser aplicado visto que não é necessário saber qual o tripulante que faltou, mas sim quantos faltaram para um certo dia.

Depois de testar vários modelos com estes dois tipos de previsões obtiveram-se melhores resultados no segundo método, com a soma das probabilidades, como é possível ver pelos exemplos dos modelos LSTM e Transformers com uma antecedência de um dia e com os dados dos maquinistas na tabela 3.8.

Modelo	Tipo	MSE	MAE	MAPE
LSTM	Probabilidade	6.07	1.78	3.21%
	Arredondamento	8.69	2.08	3.56%
Transformers	Probabilidade	10.94	2.38	4.17%
	Arredondamento	11.89	2.48	4.32%

Tabela 3.8: Erro preditivo referente ao modelo LSTM com diferentes tipos de previsão para 1 dia de antecedência.

3.4 Tempos de antecedência

O tempo de antecedência da previsão é um fator importante para o problema em causa, visto que as previsões são úteis para apoiar decisões que podem ter que ser tomadas desde horas antes do dia da operação até meses antes do dia da operação. De um modo geral, quanto mais cedo existir uma previsão das possíveis ausências, melhor será a gestão das

Modelo	Antecedência (dias)	MAE	MAPE
LSTM	1	1.78	3.21%
	3	1.91	3.46%
	7	2.27	3.98%
	30	2.04	3.75%
	180	3.46	6.68%
	365	3.49	6.90%
Transformers	1	2.38	4.17%
	3	4.85	8.67%
	7	4.80	8.57%
	30	6.36	11.54%
	180	7.87	14.53%
	365	8.38	17.01%

Tabela 3.9: Erro preditivo dos modelos LSTM e Transformer com maquinistas para diferentes antecedências

Comparando os resultados entre o modelo LSTM e Transformers, observa-se uma distinção entre os modelos. Neste caso o modelo LSTM, é o modelo com o melhor desempenho sobretudo em tempos de antecedências superiores. Se compararmos estes resultados do modelo LSTM com os resultados o trabalho prévio da tabela 2.1, observa-se uma melhoria no MAPE de 6 vezes inferior para 7 dias de antecedência e 3 vezes inferior para 365 dias de antecedência.

3.5 Adicionar revisores

Após obter bons resultados com os dados aplicados aos maquinistas para todos os tempos de previsão escolhidos anteriormente, foi aplicado também o modelo LSTM para os dados dos revisores. Os modelos foram treinados e testados com os dados dos revisores.

Primeiro adicionaram-se apenas os revisores com os mesmos primeiros 18 atributos da tabela 3.1 utilizados para os maquinistas. Na tabela 3.10 é possível ver as métricas retiradas do conjunto de teste.

Modelo	Antecedência (dias)	MAE	MAPE
LSTM	1	2.03	1.80%
	3	2.67	2.42%
	7	2.58	2.41%
	30	4.18	3.73%
	180	5.84	5.83%
	365	7.76	7.75%

Tabela 3.10: Erro preditivo do modelo LSTM com revisores para diferentes antecedências

Comparando os resultados dos revisores com os resultados dos maquinistas observa-se uma melhoria em termos percentuais, pois o conjunto de dados dos revisores é superior ao dos maquinistas, ou seja, existem mais revisores a trabalhar por dia, o que leva a uma maior ausência dos mesmos.

A segunda abordagem para adicionar os revisores foi colocar um conjunto de dados com ambos os maquinistas e os revisores, adicionando aos 18 atributos utilizados anteriormente da tabela 3.1 o atributo 19. Este atributo define se o tripulante é um maquinista ou revisor, de forma a que o modelo consiga reconhecer a diferença entre os dois tipos de tripulantes. Depois de obter as previsões do teste, separaram-se os dados por tipo de tripulante para obter as métricas relativas a cada um dos tipos, facilitando assim a comparação entre os modelos anteriores. Na tabela 3.11 estão descritos os erros de previsão do modelo LSTM.

Modelo	Tripulante	Antecedência (dias)	MAE	MAPE
LSTM	Maquinistas	1	1.72	3.13%
		3	2.61	4.50%
		7	4.46	7.55%
		30	5.86	9.99%
	Revisores	1	2.31	2.04%
		3	3.79	3.32%
		7	6.72	6.02%
		30	10.33	9.25%

Tabela 3.11: Erro preditivo do modelo LSTM com revisores e maquinistas para diferentes antecedências

Ao comparar as duas abordagens, embora ambas tenham obtido bons resultados para as antecedências menores, é possível observar que a primeira abordagem obteve os melhores resultados, com um erro de previsão mais baixo de MAE e MAPE. Assim foi a abordagem escolhida para o problema em questão.

3.6 Bases operacionais

Uma vez que grande parte da gestão do trabalho dos tripulantes é feita no âmbito de cada base operacional, torna-se importante fornecer aos planeadores de uma dada base previsões sobre o número de ausências de tripulantes dessa mesma base. Para esse efeito, é fundamental ter um registo de ausências de cada tripulante associadas à sua base de trabalho. De modo a obter-se um número total de ausências por base operacional, foram utilizados os dados de saída do modelo LSTM anteriormente referidos, agrupando os tripulantes pela sua base operacional de trabalho.

Existe um total de 24 bases operacionais diferentes para os maquinistas e revisores. Cada base tem associado um número de 1 a 24 que representa a posição em que a base

aparece na lista de bases ordenada por ordem decrescente. Sendo assim, a base com o maior número de tripulantes tem o nome de B1 e a menor B24.

Na figura 3.11 estão representadas todas as bases ordenadas pelo número médio de tripulantes com trabalho planeado por dia durante o ano de 2019. Nesta figura é possível observar tanto o número médio de tripulantes com trabalho planeado por dia como o número médio diário de ausências por parte dos maquinistas e revisores. É possível ver também que existe uma grande discrepância entre a quantidade de tripulantes por bases, sendo que B1 tem em média 431 tripulantes por dia onde cerca de 13.2% são ausências.

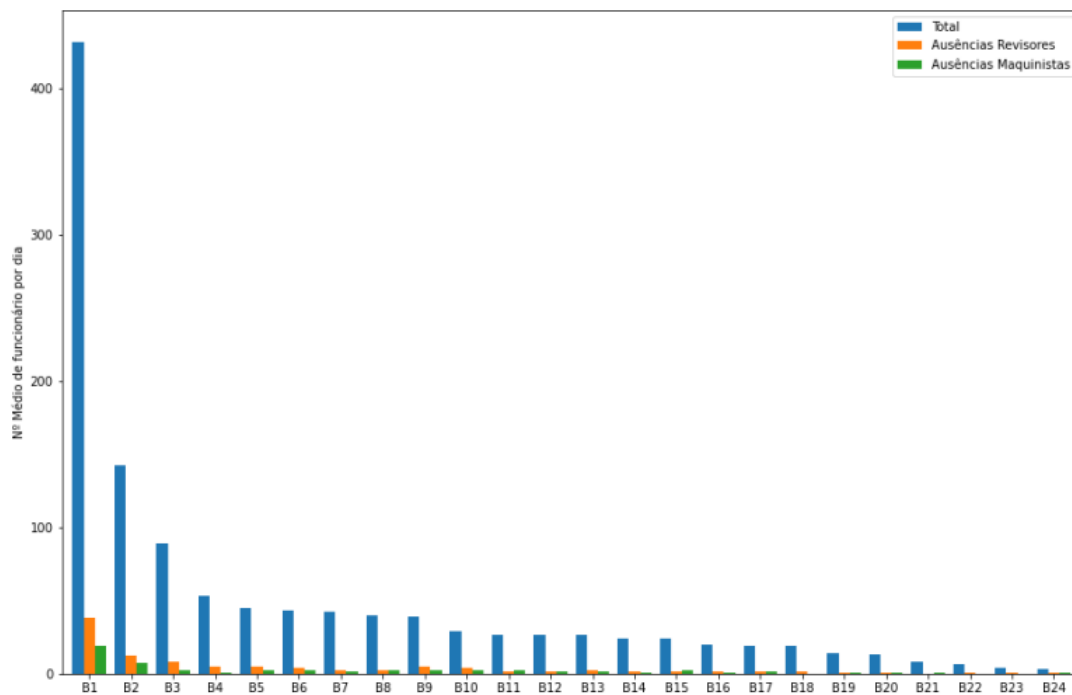


Figura 3.11: Número médio diário de tripulantes com trabalho planeado e ausências por base durante o ano 2019

Como existem bases com um número muito reduzido de tripulante e, por conseguinte, um número bastante reduzido de ausências, o erro do modelo em relação às bases mais pequenas será um pouco inconsistente. Assim, escolheram-se as 10 maiores bases para serem analisadas com maior detalhe.

De forma a avaliar a eficiência do modelo para cada uma das bases foi necessário utilizar uma métrica semelhante ao MAPE. Para estes casos não foi possível utilizar o MAPE pois existem bases com dias onde o valor de ausências é igual a zero, impossibilitando o cálculo do mesmo, porque dá origem a uma divisão por zero. Assim foi criada uma métrica semelhante utilizando o MAE, referida como MAPE auxiliar:

$$MAPE\ Auxiliar = \frac{MAE}{Número\ médio\ de\ Ausências} \quad (3.2)$$

Para conferir a semelhança do MAPE Auxiliar ao MAPE utilizou-se a maior Base (B1), visto que a mesma não contém nenhum dia sem ausências. Foi aplicado o modelo LSTM aos maquinistas com as diferentes antecedências e calculado o MAPE e o MAPE Auxiliar para cada uma, representado na tabela 3.12.

Antecedência (dias)	MAPE	MAPE Auxiliar
1	5.67%	5.20%
3	5.63%	5.57%
7	6.32%	6.02%
30	6.28%	5.88%
180	9.66%	8.82%
365	10.37%	10.15%

Tabela 3.12: Comparação entre MAPE e MAPE Auxiliar na base B1 com o modelo LSTM para maquinistas

Comparando os valores do MAPE Auxiliar com o MAPE é possível observar que embora o MAPE auxiliar seja ligeiramente inferior, os resultados tendem a ser semelhantes.

Na figura 3.12 estão representados os valores do MAPE auxiliar para as dez maiores bases e tipos de antecedência de previsão, ordenando as bases pelo número médio de tripulantes disponíveis como na figura 3.11. Este valores foram retirados através do erro do modelo LSTM para os maquinistas no ano de 2019.

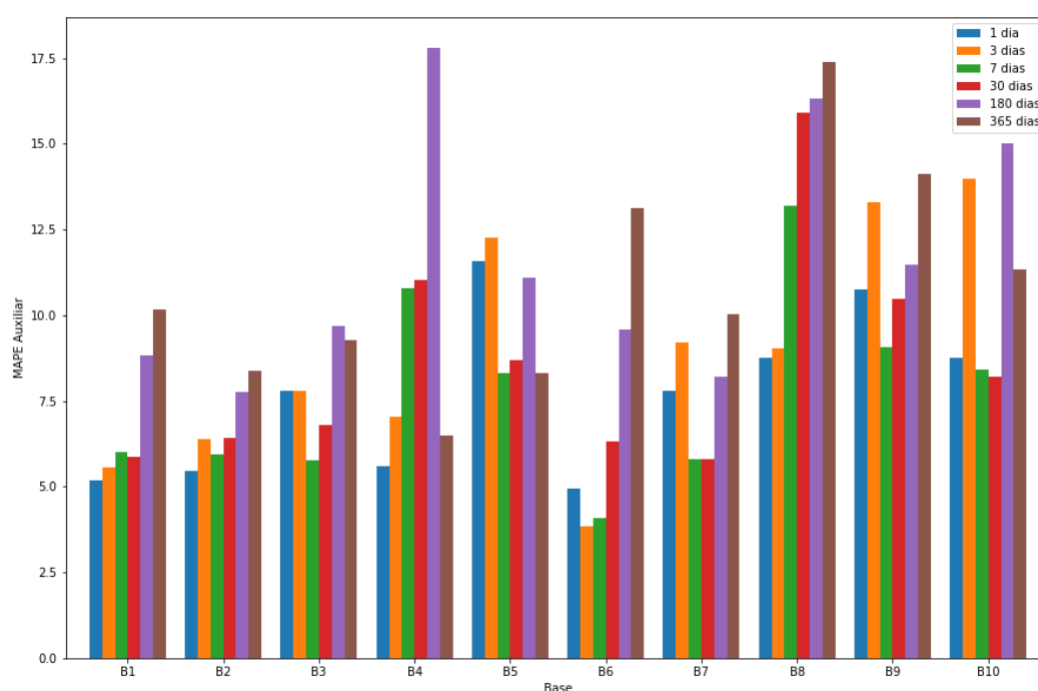


Figura 3.12: Erro preditivo MAPE Auxiliar para os maquinistas de cada base e tipo de antecedência no ano 2019

Comparando os resultados da figura 3.12 com os resultados do modelo LSTM com as bases agrupadas, tabela 3.9, é possível verificar que apesar do erro apresentar um aumento, o comportamento por base se mantém similar. Como era esperado quanto menor for a base mais inconsistente será o erro, e na maior parte dos casos quanto mais elevado for o número de dias de antecedência maior será o erro.

Os mesmos cálculos foram aplicados ao conjunto de dados dos revisores, obtendo-se assim a figura 3.13.

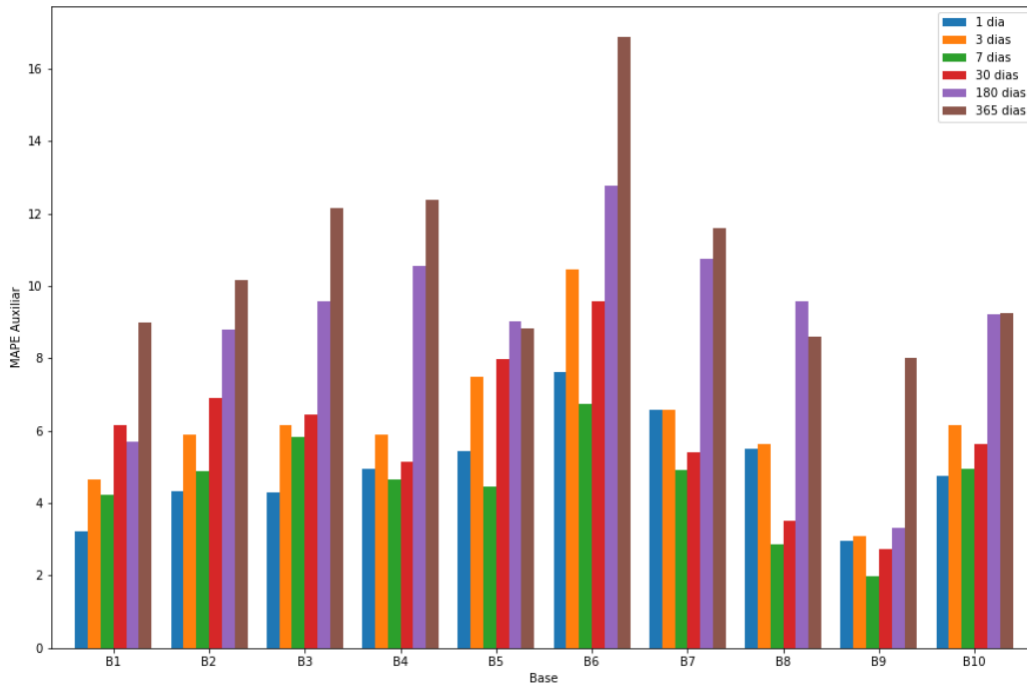


Figura 3.13: Erro preditivo MAPE Auxiliar para os revisores de cada base e tipo de antecedência no ano 2019

No caso dos conjunto de dados dos revisores, estes apresentam um maior número de ausências por base. Assim podemos observar que os erros apresentados na figura 3.13 são mais consistentes que os erros observados no conjunto de maquinistas. Retirando as mesmas conclusões que anteriormente, onde o erro aumenta com o aumento do número de dias de antecedência e o erro é mais inconsistente em bases menores.

CONCLUSÃO

O principal objetivo deste projeto foi o desenvolvimento e implementação de um modelo de previsão de ausências de tripulantes em transportes ferroviários. O modelo foi desenvolvido com o intuito de vir a constituir uma ferramenta de apoio à decisão para melhorar o planeamento de reservas, bem como o planeamento e gestão de ausências (e.g. férias) e da formação, que por sua vez irão melhorar a capacidade de ajustar a capacidade disponível às necessidades de trabalho, sobretudo em situações de contingência.

Para desenvolver este modelo, recorreu-se a redes neuronais, especificamente redes FeedForward, Transformers e Recorrentes como as LSTM. Foram utilizados dados históricos de um operador ferroviário do norte da Europa, com informações biográficas dos tripulantes e registos de atividade realizada entre 2014 e 2019. Estes dados foram divididos em dois tipos de tripulantes, maquinistas e revisores. Foram considerados vários tempos de antecedência de previsões: a 1 dia, a 3, a 7, a 30, a 180 e a 365 dias.

Todos os modelos desenvolvidos foram aplicados aos dados referentes à totalidade dos maquinistas, sendo que o modelo com os melhores resultados foi o modelo de redes recorrentes LSTM com um erro médio absoluto percentual entre 3.21% e 6.90% dependendo da antecedência de previsão. Comparado estes resultados com o trabalho prévio da SISCOG, em relação à regressão logística, foi possível melhorar significativamente a eficácia na previsão de ausências por parte dos tripulantes.

Após selecionar o melhor modelo, o mesmo foi treinado e testado com os dados dos revisores, obtendo-se um erro médio absoluto percentual compreendido entre 1.80% e 7.75%. As previsões obtidas foram divididas pelas bases operacionais correspondentes com o objetivo de validar os resultados dos modelos utilizados num contexto mais realista, onde é necessário calcular o número de ausências por base.

Dado o âmbito desta tese, como trabalho futuro podiam ser exploradas com mais detalhe as redes Transformers, que devido à sua natureza, têm capacidade para apresentar bons resultados. Para isto é necessário um estudo e otimização de parâmetros de forma mais aprofundada, o que não foi possível realizar dado o tempo limitado disponível.

BIBLIOGRAFIA

- [1] E. J. W. Abbink et al. "Solving Large Scale Crew Scheduling Problems in Practice". In: *Public Transport* (2011) (cit. on p. 1).
- [2] S. Albawi, T. A. Mohammed, and S. Al-Zawi. "Understanding of a convolutional neural network". In: *2017 International Conference on Engineering and Technology (ICET)*. 2017, pp. 1–6. DOI: 10.1109/ICEngTechnol.2017.8308186 (cit. on p. 12).
- [3] V. S. Araujo et al. "A hybrid approach of intelligent systems to help predict absenteeism at work in companies". In: *SN Applied Sciences* (2019). URL: <https://doi.org/10.1007/s42452-019-0536-y> (cit. on p. 4).
- [4] S. Devi et al. "Linear Attribute Projection and Performance Assessment for Signifying the Absenteeism at Work using Machine Learning". In: *International Journal of Recent Technology and Engineering* (2019) (cit. on p. 4).
- [5] F. T. Fernandes and A. D. P. C. Filho. "Prediction of absenteeism in public schools teachers with machine learning". In: *Revista de Saúde Pública* (2020) (cit. on p. 4).
- [6] R. P. Ferreira. "Inteligência computacional na previsão do absentismo e identificação de tendências absenteeístas". In: *Universidade Nove de Julho* (2019).
- [7] A. Jayme et al. "Comparison of Machine Learning Methods for Predicting Employee Absences". In: *Preprint Series of the Engineering Mathematics and Computing Lab* (2021). URL: <https://journals.ub.uni-heidelberg.de/index.php/emcl-pp/article/view/81078>.
- [8] T. Jensen and Q. Sun. "Absenteeism prediction and labor force optimization in rail dispatcher scheduling". In: *Signature* (2013). URL: <https://dspace.mit.edu/handle/1721.1/85457> (cit. on p. 4).
- [9] D. P. Kingma and J. Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: 1412.6980 [cs.LG] (cit. on p. 11).

- [10] E. Lima, T. Vieira, and E. de Barros Costa. "Evaluating deep models for absenteeism prediction of public security agents". In: *Applied Soft Computing* 91 (2020). URL: <https://www.sciencedirect.com/science/article/abs/pii/S1568494620301769> (cit. on p. 4).
- [11] J. M. Lourenço. *The NOVAthesis L^AT_EX Template User's Manual*. NOVA University Lisbon. 2021. URL: <https://github.com/joaomlourenco/novathesis/raw/master/template.pdf> (cit. on p. ii).
- [12] A. Martiniano et al. "Application of a neuro fuzzy network in prediction of absenteeism at work". In: *7th Iberian Conference on Information Systems and Technologies (CISTI 2012)*. 2012. URL: https://www.researchgate.net/publication/261280158_Application_of_a_neuro_fuzzy_network_in_prediction_of_absenteeism_at_work.
- [13] J. P. Martins, E. Morgado, and R. Haugen. "TPO: A System for Scheduling and Managing Train Crew in Norway". In: *Annual Conference on Innovative Applications of Artificial Intelligence (IAAI-2003)* (2003) (cit. on p. 1).
- [14] G. Matos, L. Albino, and R. L. Saldanha. "Um modelo de aprendizagem automática para previsão de ausências de maquinistas". In: *10^o Congresso Rodoferroviário Português* (2022) (cit. on pp. 2, 3, 19).
- [15] G. Nath et al. "Incorporating a Machine Learning Model into a Web-Based Administrative Decision Support Tool for Predicting Workplace Absenteeism". In: *Information* (2022). URL: <https://www.mdpi.com/2078-2489/13/7/320> (cit. on p. 4).
- [16] C. Nwankpa et al. *Activation Functions: Comparison of trends in Practice and Research for Deep Learning*. 2018. arXiv: 1811.03378 [cs.LG] (cit. on p. 8).
- [17] P. Picton. *Introduction to Neural Networks*. Macmillan Education UK, 1994. URL: https://doi.org/10.1007/978-1-349-13530-1_1 (cit. on p. 6).
- [18] A. Al-Rasheed. "Identification of important features and data mining classification techniques in predicting employee absenteeism at work". In: *International Journal of Electrical & Computer Engineering* (2088-8708) (2021) (cit. on p. 4).
- [19] A. Ruby et al. "Binary cross entropy with deep learning technique for Image classification". In: *International Journal of Advanced Trends in Computer Science and Engineering* (2020) (cit. on p. 29).
- [20] R. C. Staudemeyer and E. R. Morris. "Understanding LSTM A tutorial into long short-term memory recurrent neural networks". In: *arXiv preprint arXiv:1909.09586* (2019) (cit. on pp. 12, 13).
- [21] I. Sutskever. *Training recurrent neural networks*. University of Toronto, Canada, 2013 (cit. on p. 12).

- [22] D. Svozil, V. Kvasnicka, and J. Pospichal. "Introduction to multi-layer feed-forward neural networks". In: *Chemometrics and intelligent laboratory systems* 39.1 (1997) (cit. on p. 9).
- [23] Z. Wahid et al. "Predicting Absenteeism at Work Using Tree-Based Learners". In: *Association for Computing Machinery* (2019). URL: <https://doi.org/10.1145/3310986.3310994> (cit. on p. 4).
- [24] Z. Wang et al. *R-Transformer: Recurrent Neural Network Enhanced Transformer*. 2019. arXiv: 1907.05572 [cs.LG] (cit. on p. 14).

