UNIVERSIDADE FEDERAL DO CEARÁ

CENTRO DE CIÊNCIAS

DEPARTAMENTO DE COMPUTAÇÃO

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

MESTRADO EM CIÊNCIA DA COMPUTAÇÃO

ERICK LIMA TRENTINI

SOLUÇÕES EM DETECÇÃO DE ANOMALIAS EM SÉRIES TEMPORAIS MULTIVARIADAS UTILIZANDO MODELOS PREDITIVOS

FORTALEZA

2021

ERICK LIMA TRENTINI

SOLUÇÕES EM DETECÇÃO DE ANOMALIAS EM SÉRIES TEMPORAIS MULTIVARIADAS UTILIZANDO MODELOS PREDITIVOS

Dissertação apresentada ao Curso de Mestrado em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Ciência de Dados

Orientador: Prof. Dr. José Antônio Fernandes de Macedo

Coorientadora: Profª. Drª. Ticiana Linhares Coelho da Silva

FORTALEZA

2021

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

T729s Trentini, Erick Lima.
Soluções em detecção de anomalias em séries temporais multivariadas utilizando modelos preditivos / Erick Lima Trentini. – 2023.
53 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Ciências, Programa de Pós-Graduação em Ciência da Computação, Fortaleza, 2023.
Orientação: Prof. Dr. José Antônio Fernandes de Macedo.
Coorientação: Profa. Dra. Ticiana Linhares Coelho da Silva.

1. Séries temporais. 2. Detecção de anomalias. 3. Redes neurais. 4. Descoberta de períodos. 5. Ensembles. I. Título.

CDD 005

ERICK LIMA TRENTINI

SOLUÇÕES EM DETECÇÃO DE ANOMALIAS EM SÉRIES TEMPORAIS MULTIVARIADAS UTILIZANDO MODELOS PREDITIVOS

Dissertação apresentada ao Curso de Mestrado em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Ciência de Dados

Aprovada em: 30 de Abril de 2021

BANCA EXAMINADORA

Prof. Dr. José Antônio Fernandes de Macedo (Orientador)
Universidade Federal do Ceará (UFC)

Profª. Drª. Ticiana Linhares Coelho da Silva (Coorientadora)
Universidade Federal do Ceará (UFC)

Prof. Dr. Leopoldo Melo Júnior
Universidade Federal do Ceará (UFC)

Prof. Dr. César Lincoln Cavalcante Mattos
Universidade Federal do Ceará (UFC)

À minha mãe e minha vó que pavimentaram todos os caminhos que eu percorri. À minha namorada que sempre me apoiou e incentivou. Aos meus amigos que sempre acreditaram que eu conseguiria.

AGRADECIMENTOS

À minha mãe que sempre me apoiou e ajudou em todos os momentos da vida e à minha vó que sempre foi o pilar e a inspiração para todos da família.

Aos meus orientadores José Macedo e Ticiana Linhares, e ao meu amigo Leopoldo Melo que me acompanharam e orientaram por todo o mestrado e foram pilares importantes para a produção deste trabalho.

À minha namorada que esteve sempre ao meu lado durante todos esses anos desde o início da graduação e me apoiou e incentivou em todas as decisões tomadas.

Aos meus colegas e amigos do Insight Data Science Lab que fizeram grande parte do meu amadurecimento profissional e científico.

Aos meus amigos e minha família que sempre me apoiaram e me deram forças para seguir em frente.

À Fundação Cearense de Apoio ao Desenvolvimento Científico e Tecnológico (FUNCAP) pelo financiamento parcial deste trabalho, que contribuiu significativamente para seu desenvolvimento.

Por fim, ao Doutorando em Engenharia Elétrica, Ednardo Moreira Rodrigues, e seu assistente, Alan Batista de Oliveira, aluno de graduação em Engenharia Elétrica, pela adequação do *template* utilizado neste trabalho para que o mesmo ficasse de acordo com as normas da biblioteca da Universidade Federal do Ceará (UFC).

“I rarely end up where I was intending to go, but often I end up somewhere I needed to be.”

(Douglas Adams)

RESUMO

Detecção de anomalias em séries temporais é uma área de estudo em rápido crescimento atualmente, devido ao aumento exponencial da criação de novos dados temporais produzidos por sensores de diversos contextos, como, por exemplo, a Internet das Coisas (IoT). Muitos modelos preditivos foram propostos ao longo dos anos, e muitos trazem resultados promissores na diferenciação de pontos normais e anômalos nas séries temporais. Neste trabalho, serão propostas três contribuições. Em uma delas, buscamos encontrar e combinar os melhores modelos preditivos em detecção de anomalias em séries temporais, para que as diferentes estratégias e diferentes parâmetros na criação dos modelos possam contribuir para a análise das séries, propondo um ensemble baseado em modelos chamado **TSPME-AD** (*Time Series Prediction Model Ensemble for Anomaly Detection*, ou Ensemble de Modelos Preditivos em Séries Temporais para a Detecção de Anomalias). O **TSPME-AD** utiliza os modelos preditivos do estado-da-arte e combina seus scores de anomalias com uma função ponderada. As outras duas contribuições desse trabalho são uma técnica dinâmica de segmentação de janelas, que utiliza a periodicidade e o formato das séries para facilitar o treinamento dos modelos e a descoberta de padrões, e um novo modelo de auto-encoder que modifica a estrutura de um dos modelos do estado-da-arte. A efetividade das propostas do trabalho é analisada com o uso de dois conjuntos de dados reais, sendo esses os dados de um ano de demanda de energia elétrica e o banco de dados de eletrocardiogramas do MIT. Com os experimentos, demonstramos que a técnica de ensemble proposta melhora o score F_1 em até 22% comparado com o melhor score dentre os modelos individuais que a compõem, com nossa função de combinação específica apresentando uma melhora de até 13% com relação a outras funções de combinação mais simples. Também demonstramos que nossa nova arquitetura de auto-encoder, combinada com a nova estratégia de segmentação dinâmica de janelas, consegue melhorias de até 25% no score F_1 comparado com uma das técnicas de auto-encoder do estado-da-arte, e uma melhoria de até 64% comparado com um modelo de LSTM empilhada.

Palavras-chave: séries temporais; detecção de anomalias; redes neurais; descoberta de períodos; ensembles.

ABSTRACT

Time-series anomalies detection is a fast-growing area of study, due to the exponential growth of new data produced by sensors in many different contexts as the Internet of Things (IoT). Many predictive models have been proposed, and they provide promising results in differentiating normal and anomalous points in a time-series. In this work, we provide three contributions. We aim to identify and combine the best models for detecting anomalies in time-series, so that the different strategies or parameters of the models can contribute to the time series analysis by proposing a model-centered ensemble called TSPME-AD (stands for Time Series Prediction Model Ensemble for Anomaly Detection). TSPME-AD uses state-of-the-art predictive models, combining their anomaly scores using a weighted function. Other contributions are a dynamic window breaking technique based on scanning thresholds, leveraging the periodicity and shape of the series to generate windows that aid in training and testing models, and a new auto-encoder predictive model. The effectiveness of our proposals is analyzed using two real-world time-series datasets, a year of power demand data, and the MIT electrocardiogram database. We show that our ensemble technique improves on the F_1 score up to 22% on the best score of the individual models composing the ensemble, with our specific combination function improving on simpler functions on up to 13% F_1 score increase. We also show that our new auto-encoder architecture, combined with the new window breaking technique, can have an up to 25% F_1 score increase compared to another proposed auto-encoder technique, and a 64% score increase over a stacked LSTM model.

Keywords: time series; anomaly detection; neural networks; period discovery; ensembles.

LISTA DE FIGURAS

Figura 1 – Série temporal periódica com dois ciclos identificáveis e constantes 17
Figura 2 – Série quasi-periódica de um electrocardiograma que pode apresentar pequenas variações nos ciclos 18
Figura 3 – Série não periódica gerada randomicamente 18
Figura 4 – Anomalia contextual em série temporal apresentada por (CHEBOLI, 2010) . 19
Figura 5 – Anomalias de sub-sequência onde sequências de altas esperadas não ocorreram 19
Figura 6 – Nó (neurônio) de uma rede neural artificial 20
Figura 7 – Rede neural artificial simples com apenas uma camada interna 20
Figura 8 – Nó LSTM onde a seta horizontal superior representa a memória interna do nó 21
Figura 9 – Arquitetura do modelo de LSTM empilhada proposta por (MALHOTRA *et al.*, 2015) 25
Figura 10 – Modelo de *Encoder-Decoder* proposto por (MALHOTRA *et al.*, 2016) . . . 26
Figura 11 – Passo a passo completo da arquitetura do TSPME-AD 29
Figura 12 – Primeiro, todos os modelos tentam reconstruir a série temporal, e são calculados os escores de anomalia 30
Figura 13 – Aplicando uma função de amortecimento nos escores de anomalias, como a função de logaritmo natural 31
Figura 14 – Agregação dos conjuntos de escores de anomalia utilizando uma função de média ponderada 32
Figura 15 – Utilizando um *threshold* para discriminar entre pontos normais e anômalos na série 33
Figura 16 – Uma semana normal de demanda de energia elétrica, iniciando-se na quarta-feira 34
Figura 17 – Exemplos de partes dos dados dos electrocardiogramas dos dados do MIT . 35
Figura 18 – Quatro batimentos cardíacos de um dos eletrocardiogramas do conjunto de dados 36
Figura 19 – Arquitetura do modelo proposto (CRED) 43
Figura 20 – Uma escolha ruim de *threshold* 47
Figura 21 – Uma escolha de threshold aceitável, mas que pode ser melhorada 47
Figura 22 – Uma boa escolha de threshold que divide a série em cada um de seus ciclos . 47

LISTA DE TABELAS

Tabela 1 – Resultados do teste sobre os dados de demanda de energia elétrica 38
Tabela 2 – Resultados dos testes no electrocardiograma 40
Tabela 3 – Resultados dos testes para os dados de eletrocardiogramas do MIT 49

LISTA DE ABREVIATURAS E SIGLAS

CRED	Concrete Representation Encoder-Decoder
TSPME-AD	Time Series Prediction Model Ensemble for Anomaly Detection
WSST	Window Similarity Scanning Threshold

SUMÁRIO

1 INTRODUÇÃO 14
1.1 Definição do Problema 15
1.2 Contribuições 15
1.3 Estrutura da Dissertação 16
2 FUNDAMENTAÇÃO TEÓRICA 17
2.1 Séries Temporais 17
2.2 Anomalias em Séries Temporais 17
2.3 Redes Neurais Artificiais 18
2.4 Redes neurais recorrentes e LSTMs 21
3 TRABALHOS RELACIONADOS 23
3.1 LSTM Empilhada 24
3.2 Encoder-Decoder 26
4 ENSEMBLE DE MODELOS PREDITIVOS EM SÉRIES TEMPORAIS PARA DETECÇÃO DE ANOMALIAS 28
4.1 Introdução 28
4.2 Arquitetura 29
4.3 Combinação e Discriminação 31
4.4 Experimentação 33
4.4.1 Datasets 34
4.4.1.1 Power Demand *34*
4.4.1.2 MIT Electrocardiogram Dataset *35*
4.4.2 Resultados 36
4.4.2.1 Resultados sobre os dados de demanda de energia elétrica *37*
4.4.2.2 Resultados sobre o conjunto de dados de eletrocardiogramas *38*
4.5 Conclusão 39
5 ENCODER-DECODER DE REPRESENTAÇÃO CONCRETA E NOVAS TÉCNICAS PARA SEGMENTAÇÃO DE JANELAS 42
5.1 Introdução 42
5.2 Encoder-Decoder de Representação Concreta (CRED) 43
5.3 Estratégias de segmentação de janelas 44

5.3.1 ***Segmentação Baseada em Tamanho de Janela Constante*** 45
5.3.2 ***Segmentação Baseada em Picos*** 45
5.3.3 ***Segmentação por Similaridade de Janelas através de Varredura de Threshold (WSST)*** 46
5.4 **Experimentação e Resultados** 48
5.5 **Conclusão** 50
6 **CONCLUSÃO** 51
6.1 **Trabalhos Futuros** 52
REFERÊNCIAS 53

1 INTRODUÇÃO

Preços de ações, monitoramento do sono e trajetórias de objetos móveis são conjuntos de dados que frequentemente apresentam alguma noção de tempo. Quando um ou múltiplos parâmetros de algum desses contextos são coletados em uma sequência de tempo, obtém-se o que é chamado de série temporal.

A coleta de grandes volumes de dados de séries temporais abre várias oportunidades de descobertas de padrões desconhecidos nesses dados, assim como a possibilidade de identificação de desvios nesses padrões, os quais podem indicar informações muito relevantes. Por exemplo, médicos podem buscar anomalias no padrão de sono de um paciente para investigar algum diagnóstico. Uma aplicação de trajetórias em mapa pode identificar anomalias no fluxo de veículos em uma via e tomar decisões ideais de trajetória em tempo real para seus usuários. Aplicações bancárias podem detectar anomalias nos fluxos de transferências de usuários, alertando sobre possíveis fraudes em tempo real, entre outras inúmeras aplicações possíveis.

Existem várias abordagens para o problema de detecção de anomalias em séries temporais. Diversas técnicas são apresentadas na literatura, incluindo modelos preditivos, baseados em clusterização, em distância, entre outros (MENG *et al.*, 2018). Devido à dificuldade de modelar séries temporais, especialmente por conta do fator sequencial dos dados, o estado da arte tem investigado redes neurais utilizando LSTM (HOCHREITER; SCHMIDHUBER, 1997) para modelar o comportamento normal das séries temporais e utilizar desvios do comportamento padrão para detectar anomalias, sem a necessidade de um threshold pré-determinado ou fase de pré-processamento (MALHOTRA *et al.*, 2015; MALHOTRA *et al.*, 2016).

Os modelos baseados em LSTM são muito dependentes de um grande conjunto de dados. No entanto, como as anomalias, por definição, são eventos raros no mundo real, os modelos preditivos se beneficiam do fato de que a maior parte dos dados disponíveis para treinamento é composta por séries sem anomalias. Essa abordagem baseia-se na premissa de que, ao treinar um modelo para reconstruir pontos de uma série temporal normal, ele será incapaz de reconstruir adequadamente uma série que contenha um ou mais pontos anômalos, resultando em erros de reconstrução atípicos, que são utilizados para identificar as anomalias.

Este trabalho investiga um problema desafiador, pois a detecção de anomalias é realizada sobre séries temporais multivariadas. Como apresentado em (WANG *et al.*, 2018), anomalias podem ocorrer em sub-conjuntos de dimensões, os locais e tamanhos dessas anomalias podem variar entre as diferentes dimensões. Além disso, uma série com anomalias pode parecer

normal em suas componentes individuais, mas a combinação delas revela a anomalia.

1.1 Definição do Problema

Considere a série temporal multivariada $X = [x^{(1)}, x^{(2)}, \ldots, x^{(n)}]$ tal que $x^{(i)} \in \mathbb{R}^m$ é um vetor m-dimensional $x^{(i)} = [x_1^{(i)}, x_2^{(i)}, \ldots, x_m^{(i)}]$ no ponto $t = i$. Normalmente, os modelos preditivos procuram prever o próximo ponto, dada uma série como entrada. Ou seja, para um modelo preditivo M e uma série temporal $X = [x^{(1)}, x^{(2)}, \ldots, x^{(n)}]$, $M(x^{(i)}) = x^{(i+1)}$. Alguns modelos podem variar um pouco desta perspectiva, como, por exemplo, ao tentar prever mais de um ponto no futuro, $M(x^{(i)}) = [x^{(i+1)}, x^{(i+2)}]$, ou reconstruindo a série de forma retroativa, $M(x^{(i)}) = x^{(i-1)}$.

Dado um modelo preditivo M e uma série temporal X, $Y = M(X)$ representa a sequência gerada a partir de X com o uso de M, tal que $Y = [y^{(1)}, y^{(2)}, \ldots, y^{(n)}]$, onde $y^{(i)}$ é uma tentativa de reconstrução de $x^{(i)}$ por M. O objetivo principal é reconstruir a sequência X utilizando M, computar os erros de reconstrução baseados na predição $M(x^{(i)})$ em comparação com x_i, calcular os escores de anomalias com base na distribuição de erros computados, e utilizar os escores para identificar as anomalias em X.

1.2 Contribuições

Uma das técnicas que pode ser utilizada no contexto de séries temporais é o ensemble de modelos, que visa treinar vários modelos com propriedades diferentes sob um mesmo conjunto de dados, e combinar seus resultados utilizando algum tipo de função de combinação. Vários esquemas de combinação foram propostos ao longo dos anos, com alguns deles conseguindo demonstrar empiricamente serem consistentemente melhores que seus modelos individuais (KITTLER *et al.*, 1996). A literatura apresenta várias utilizações de ensembles, como para problemas de classificação, regressão, predição em séries temporais, entre outras utilizações possíveis.

Neste trabalho, é feito um estudo sobre a utilização de ensembles para detecção de anomalias em séries temporais, por meio da combinação de modelos preditivos. Além disso, o trabalho propõe o Time Series Prediction Model Ensemble for Anomaly Detection (TSPME-AD), um ensemble focado na combinação de modelos que treinam sob todo o conjunto de treino, utilizando uma função de damping (AGGARWAL, 2013) para a normalização dos resultados

dos modelos individuais e utilizando uma média ponderada como função de combinação.

Também será proposta uma nova arquitetura baseada em LSTM, que modifica o modelo de Encoder-Decoder proposto em Malhotra *et al.* (2016), em que, em vez de usar os estados internos do LSTM do encoder como estado inicial do decoder para a reconstrução da série, se utiliza uma camada densa entre o Encoder e o Decoder que servirá como entrada para o Decoder. Além disso, como as técnicas de detecção de anomalias em séries temporais apresentadas nesse trabalho, e as do estado da arte, incluindo as apresentadas nos trabalhos relacionados, como (MALHOTRA *et al.*, 2015; MALHOTRA *et al.*, 2016), normalmente quebram as séries temporais em janelas de tamanho igual para serem passadas como input para os modelos, também será proposta uma técnica de quebra de janelas dinâmica, que utiliza a similaridade entre as janelas geradas para escolher uma segmentação em janelas que melhor represente os períodos da série temporal.

1.3 Estrutura da Dissertação

Esta dissertação é estruturada da seguinte forma: No capítulo 2 apresenta-se a fundamentação teórica do trabalho. O capítulo 3 apresenta alguns trabalhos do estado da arte em detecção de anomalias em séries temporais utilizando modelos preditivos. O capítulo 4 apresenta uma análise de ensembles para o problema e propõe uma função de combinação. No capítulo 5, será apresentada uma nova arquitetura de modelo preditivo para a detecção de anomalias, além de uma técnica de quebra de janelas para pré-processamento das séries. Por fim, no capítulo 6 são apresentadas as conclusões finais do trabalho e possíveis linhas de pesquisa futuras.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão apresentadas algumas definições e arquiteturas que servirão de base teórica para os modelos propostos neste trabalho, e para a compreensão das arquiteturas apresentadas ao longo da dissertação.

O capítulo irá apresentar os seguintes conceitos: Séries Temporais, Anomalias em Séries Temporais, Redes Neurais Artificiais, Redes Neurais Recorrentes e LSTMs.

2.1 Séries Temporais

Uma série temporal X é uma sequência de pontos ou vetores $x \in \mathbb{R}^n$, onde cada ponto possui uma marcação de tempo, com intervalos de tempo constantes ou variáveis entre os pontos.

Ciclos em uma série temporal podem ser definidos como um padrão identificável que se repete ao longo da série, e, com base neles, uma série temporal pode ser classificada de três formas: **Não Periódica**, **Quasi-Periódica** e **Periódica**. Uma série **periódica**, como exemplificado na Figura 1, possui ciclos bem definidos, em que a duração dos ciclos é constante. Uma série **quase-periódica** possui ciclos bem definidos, mas as durações dos ciclos e o padrão dos ciclos podem ter pequenas variações, como exemplificado na Figura 2. Por fim, uma série **não periódica** não apresenta ciclos identificáveis, o que a torna mais complexa para análise, como exemplificado na Figura 3.

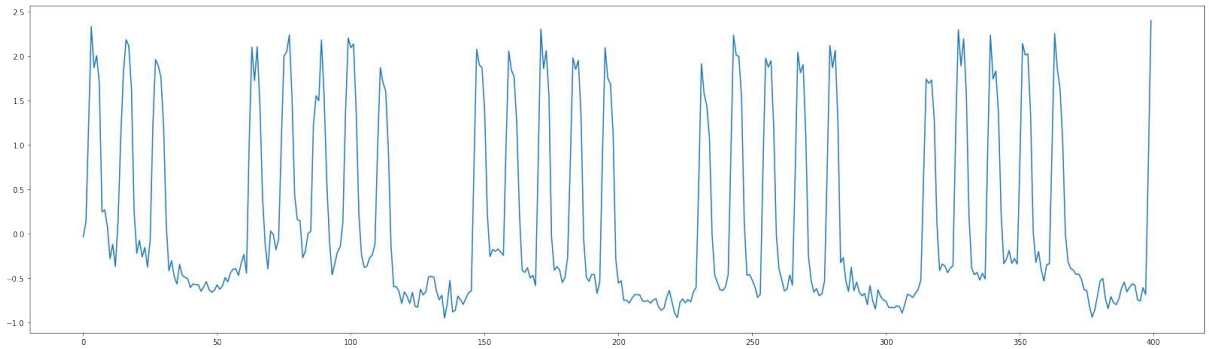

Figura 1 – Série temporal periódica com dois ciclos identificáveis e constantes

2.2 Anomalias em Séries Temporais

Anomalias em séries temporais podem ser descritas como "quebras"nos padrões dos ciclos da série, que podem se apresentar de diversas formas conforme apresentado em

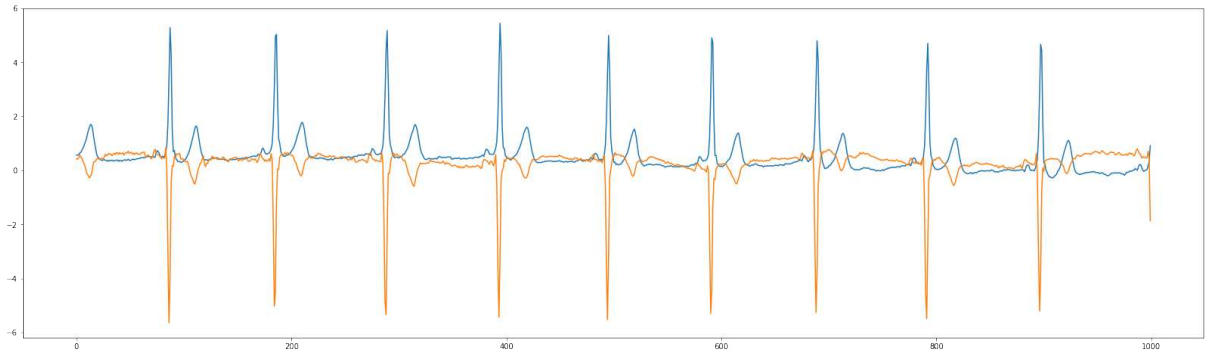

Figura 2 – Série quasi-periódica de um electrocardiograma que pode apresentar pequenas variações nos ciclos

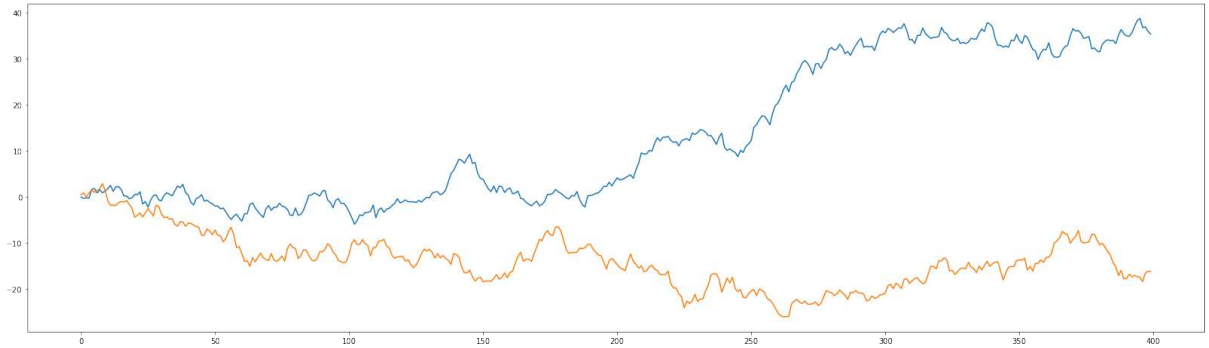

Figura 3 – Série não periódica gerada randomicamente

(CHEBOLI, 2010), como um valor baixo em um local de um ciclo que se esperaria um valor alto, ou um valor alto em um local que se espera um valor baixo, chamadas de anomalias de contexto, exemplificadas na Figura 4, sequências de valores que não seguem o padrão esperado da série, chamadas de anomalias de sub-sequências e exemplificadas na Figura 5, e em séries multivariadas, anomalias podem se apresentar de forma mais complexa, sendo detectadas apenas por meio de uma combinação de suas dimensões.

2.3 Redes Neurais Artificiais

Redes neurais artificiais são modelos de camadas de nós interconectados que tentam mimetizar o comportamento dos neurônios do cérebro humano, utilizando essa estrutura para aprender padrões em conjuntos de dados, resolvendo problemas de classificação, regressão, entre outros.

Como no neurônio biológico, o neurônio ou nó da rede neural, exemplificado na Figura 6, recebe várias entradas com um peso para cada um, soma os valores ponderados e aplica uma função de ativação como a *sigmoid* ou *tanh* sobre esse resultado para obter a saída do nó.

A rede neural artificial é construída a partir da junção de camadas desses nós, como

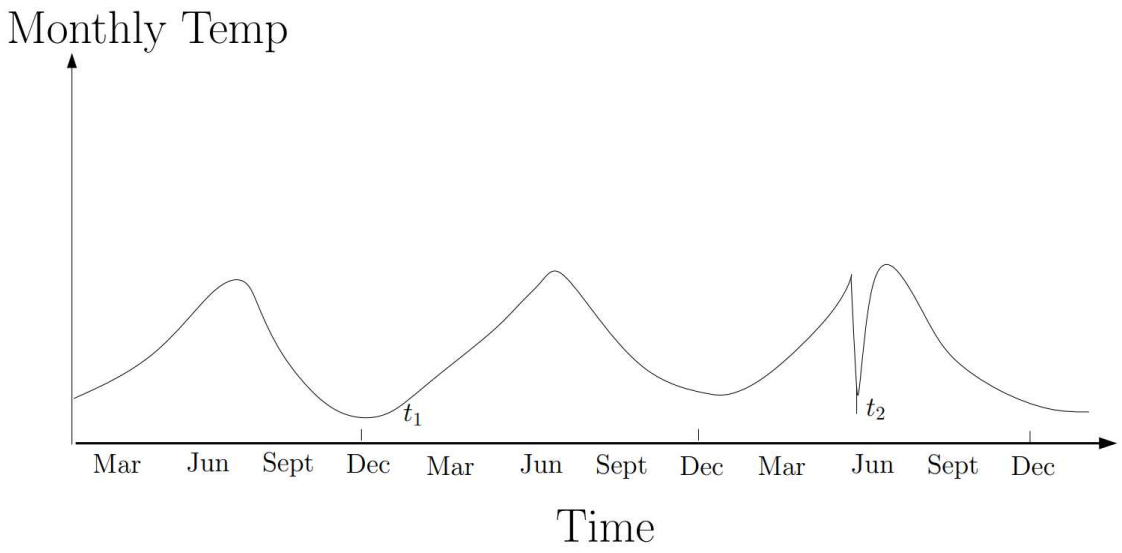

Figura 4 – Anomalia contextual em série temporal apresentada por (CHEBOLI, 2010)

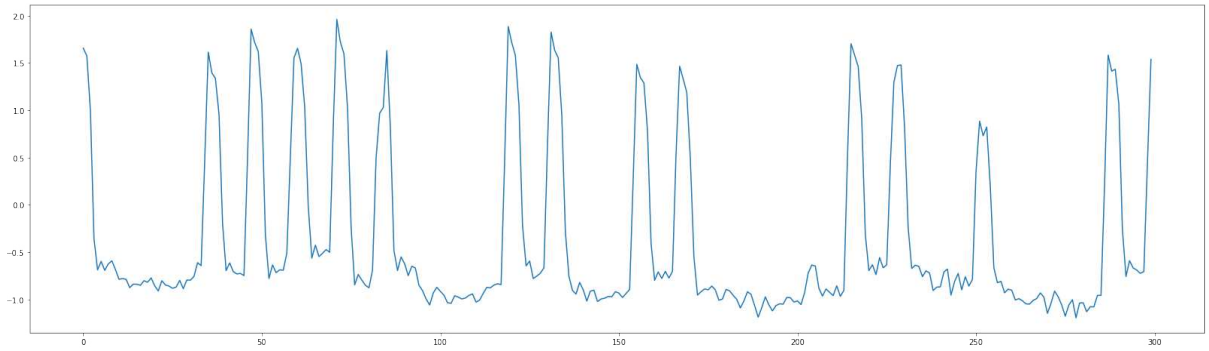

Figura 5 – Anomalias de sub-sequência onde sequências de altas esperadas não ocorreram

exemplificado na Figura 7, em que cada nó de uma camada posterior tem conexões com todos os nós da camada anterior, sendo composta por uma camada de *input*, que recebe os dados de entrada e precede todas as outras camadas, e uma camada de *output* ao final, responsável por retornar o valor computado da rede neural.

O treinamento para reconhecimento de padrões com redes neurais é dependente do contexto do problema, mas segue um certo conjunto de etapas comuns. Por exemplo, em um problema de classificação, a camada de *output* deve retornar a classe correta a partir de uma determinada entrada. Para isso, o treinamento da rede utiliza os erros de classificação para ajustar gradualmente os pesos dos nós da rede, e com essas alterações, reduzir progressivamente os erros de classificação.

Existem vários algoritmos de otimização desses pesos, como algoritmos genéticos, buscas aleatórias, sendo o mais comumente utilizado o *backpropagation*, que emprega um processo que calcula a derivada da função de ativação com os pesos e erros obtidos em uma

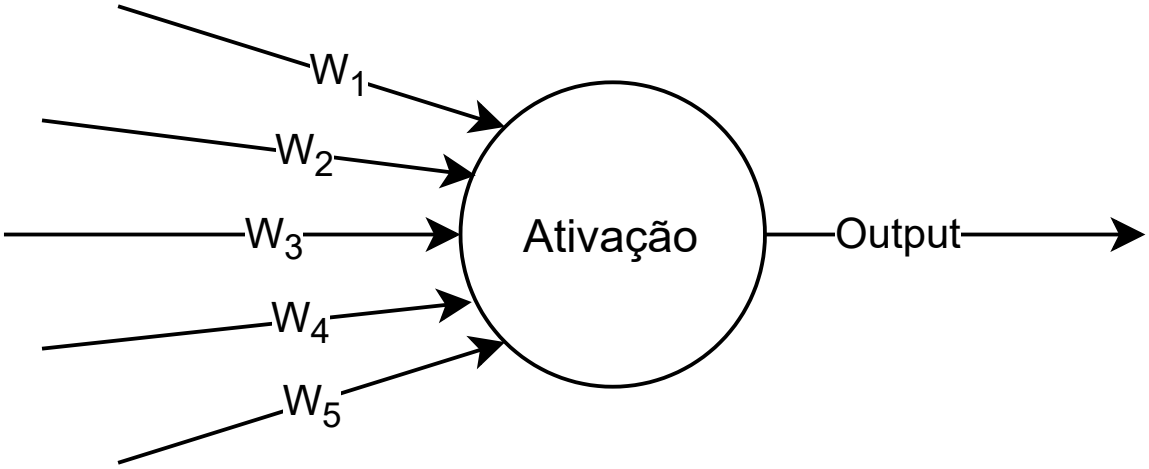

Figura 6 – Nó (neurônio) de uma rede neural artificial

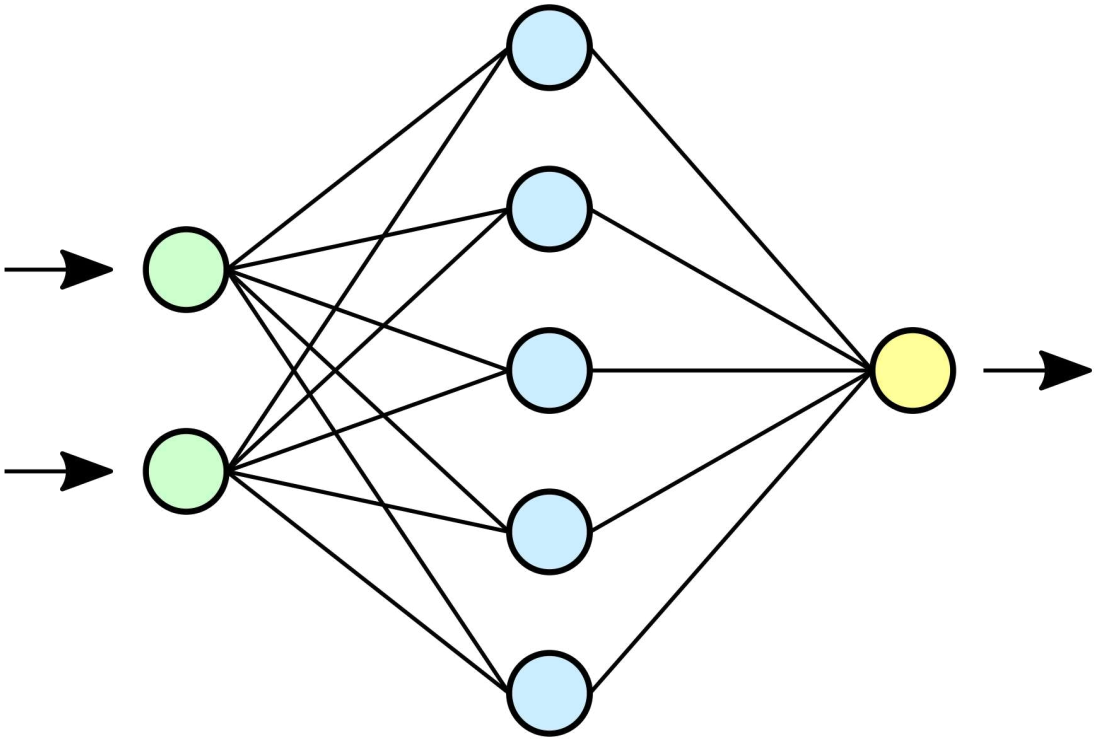

Figura 7 – Rede neural artificial simples com apenas uma camada interna

camada e ajusta os pesos com base nesses gradientes, o que acelera o processo de treinamento e convergência do modelo ao direcionar a atualização dos pesos em direção a um mínimo local da função de erro.

2.4 Redes neurais recorrentes e LSTMs

Redes neurais recorrentes ou RNNs são uma classe de redes neurais que recebem como *input* uma sequência temporal de dados de entrada, e utilizam o *output* de camadas internas da rede como *input* da própria camada, agindo como um tipo de "memória"para o reconhecimento de padrões em dados sequenciais.

A RNN é uma tentativa mais simples de modelar dados sequenciais com redes neurais, mas por sua "simplicidade"acaba sofrendo com problemas de "esquecimento"de padrões em sequências mais longas, comprometendo sua capacidade de modelagem à medida que a sequência de dados de entrada aumenta em tamanho. Para resolver esse problema, algumas variantes de redes recorrentes foram propostas, com a mais conhecida delas sendo a LSTM.

LSTM é uma abreviação para *Long Short-Term Memory*, e define uma arquitetura de nós em redes neurais recorrentes que utiliza pesos internos dedicados apenas à "memória"de padrões, e "portões"que controlam a memorização ou o esquecimento desses padrões com base nos dados de entrada ou nos estados recorrentes, podendo ser visualizado na Figura 8.

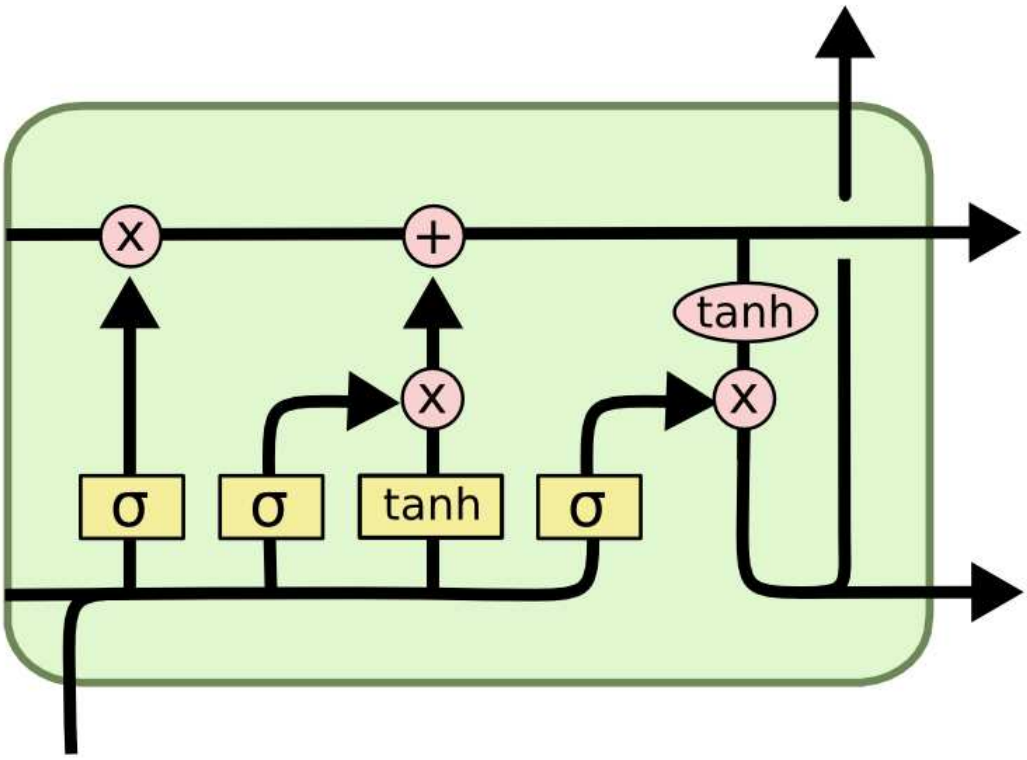

Figura 8 – Nó LSTM onde a seta horizontal superior representa a memória interna do nó

Devido à sua memória interna, a LSTM é capaz de aprender padrões de sequências bem maiores, obtendo resultados melhores que as **RNNs** nesses casos, com o ponto negativo de uma convergência no treinamento mais demorada, tanto pela maior quantidade de pesos a serem treinados em um único nó, quanto pela complexidade das funções utilizadas para cada "portão"e *output*.

3 TRABALHOS RELACIONADOS

Modelos de detecção de anomalias em séries temporais têm sido investigados na literatura utilizando-se técnicas de *machine learning* e métodos estatísticos, conforme apresentado em Chandola *et al.* (2009), e uma nova gama de técnicas que vêm ganhando tração recentemente são as redes neurais profundas, que são geralmente utilizadas para modelagens não-lineares. Apesar disso, poucos estudos consideram as redes profundas para abordar o problema de detecção de anomalias no contexto de séries temporais, tendo ganhado mais destaque nos últimos anos.

O estudo de (MALHOTRA *et al.*, 2016) por exemplo, propõe um modelo de *encoder-decoder* baseado em LSTM, que é treinado para reconstruir instâncias de séries temporais com comportamento normal. A ideia é que quando uma série anômala é passada para o modelo, ele não consegue reconstruí-la tão bem como reconstrói as séries normais, fazendo com que os valores dos erros de reconstrução sejam significativamente mais altos que o normal, sinalizando as anomalias. Outro trabalho que propõe um modelo baseado em redes de LSTM é o (MALHOTRA *et al.*, 2015), que propõe uma arquitetura de camadas de LSTM empilhadas. Similarmente, (KIEU *et al.*, 2018) propõe um *framework* de detecção de *outliers* para identificar anomalias em séries multidimensionais, incorporando várias redes de *auto-encoders* para reconstruir as séries passadas como *input* e discriminar as anomalias baseadas em erros de reconstrução.

A técnica proposta em (KONG *et al.*, 2018) consegue detectar anomalias de tráfego em grandes espaços de tempo utilizando dados de trajetórias de ônibus. Segmentos de séries temporais são extraídos dos dados de trajetórias dos ônibus para descrever tanto os aspectos espaciais quanto temporais da situação do tráfego de toda a cidade. (KONG *et al.*, 2018) extrai a velocidade média e o tempo de parada médio dos ônibus, que descrevem, respectivamente, as condições do trânsito e a demanda. Depois, segmentos "ruins"são encontrados nos dados pelo cálculo dos índices de anomalia dos segmentos. Já o trabalho (TARIQ *et al.*, 2019) propõe um detector de anomalias para um sistema de satélite utilizando uma LSTM convolucional multivariada, combinada com um modelo de *Mixtures of Probabilistic Principal Component Analyzer*. O modelo proposto treina em um grande conjunto de dados normais de telemetria e discrimina entre sequências de telemetria normais e anômalas.

Outros tipos de algoritmos de detecção de anomalias utilizam técnicas de clusterização. Por exemplo, o trabalho (WANG *et al.*, 2018) propõe um algoritmo de clusterização que discretiza a série temporal em janelas de tempo e clusteriza todas as subsequências dentro de cada janela. Subsequências univariadas no mesmo *cluster* dentro de uma janela são similares

umas com as outras; assim, os padrões de comportamento dos objetos são obtidos pelos centros dos *clusters* e, caso uma série temporal não siga esse padrão de comportamento, ela é considerada anômala. Para séries multivariadas, o algoritmo transforma a série original em um novo espaço de *features*, onde cada *feature* é a distância para um padrão, fazendo com que, quanto menor for a distância, mais similar é o dado do padrão. (WANG *et al.*, 2018) realiza a clusterização no dado transformado, e calcula um escore de anomalia para cada série temporal, baseado nos resultados da clusterização e das distâncias para os *clusters* normais. Outras técnicas baseadas em clusterização são propostas por (GAO *et al.*, 2012; IVERSON, 2004).

O trabalho (GAO; TAN, 2006) apresenta um modelo de ensemble que se assemelha, no processo de combinação, ao modelo proposto neste trabalho, utilizando uma combinação dos escores de anomalia obtidos pelos modelos base do ensemble, embora seja aplicado em um contexto mais geral de detecção de anomalias.

Nesse capítulo serão enfatizados os modelos propostos por (MALHOTRA *et al.*, 2015; MALHOTRA *et al.*, 2016), pois são técnicas de detecção de anomalias em séries temporais utilizando modelos preditivos, o que segue a proposta deste trabalho, e, além disso, são os modelos utilizados para a criação do ensemble apresentado no Capítulo 4, sendo que, especialmente, o modelo proposto por (MALHOTRA *et al.*, 2016) serve de base para a proposta de uma nova arquitetura no Capítulo 5.

3.1 LSTM Empilhada

Considere quatro conjuntos de séries temporais: s_N e v_N, contendo apenas instâncias de séries temporais sem anomalias, v_A e t_A, contendo tanto séries normais quanto séries anômalas, o conjunto s_N será usado para o treinamento do modelo preditivo M, v_N para a geração da distribuição normal dos vetores de erro, v_A para a definição do *threshold* de discriminação de anomalias, e t_A para a avaliação da qualidade do modelo M. Seja $s_N = [s_N^{(1)}, s_N^{(2)}, \ldots, s_N^{(n)}]$, tal que $s_N^{(i)} \in \mathbb{R}^m$ é um vetor m-dimensional e $s_N^{(i)} = [s_{N_1}^{(i)}, s_{N_2}^{(i)}, \ldots, s_{N_m}^{(i)}]$ no ponto temporal $t = i$. O mesmo se aplica aos conjuntos v_N, v_A e t_A.

Para $s_N^{(i)}$, cada uma das m dimensões ($s_N^{(i)} \in \mathbb{R}^m$) é lida por um nó na camada de *input* do modelo. Na camada de *output*, existe um nó para cada um dos l pontos temporais futuros previstos, e m dimensões para cada ponto, ou seja, $l \times m$ nós na camada de *output*. Os nós de LSTM na camada oculta são totalmente conectados através de conexões recorrentes. Malhotra *et al.* (2015) empilha camadas de nós LSTM de forma que cada nó em uma camada inferior é

totalmente conectado com todos os nós da camada posterior, por meio de conexões simples de *feedforward*. A figura 9 mostra a arquitetura do modelo de LSTM empilhada.

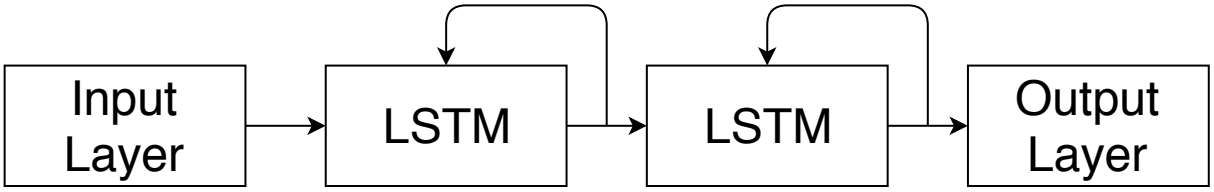

Figura 9 – Arquitetura do modelo de LSTM empilhada proposta por (MALHOTRA *et al.*, 2015)

Seja M o modelo de LSTM empilhada, X uma série temporal e l o número de pontos futuros preditos. Para cada ponto de tempo t na série temporal X (onde $l < t \leq n - l$) e para cada uma das suas d dimensões, o modelo prevê l pontos no futuro. Vetores de erro são computados para cada $x^{(t)}$, tal que $e^{(t)} = [e_{11}^{(t)}, \ldots, e_{1l}^{(t)}, \ldots, e_{d1}^{(t)}, \ldots, e_{dl}^{(t)}]$ onde $e_{ij}^{(t)}$ é a diferença entre $x_i^{(t)}$ e o valor predito pelo modelo M no ponto de tempo $t - j$. Conforme Malhotra *et al.* (2015), o modelo preditivo, treinado no conjunto de treinamento s_N, é usado para computar os vetores de erro para cada ponto nas sequências de dados de validação e teste. O conjunto de vetores de erro gerados pelo modelo M no conjunto de treinamento é modelado para criar uma distribuição gaussiana multivariada $\mathcal{N}(\mu, \Sigma)$. O conjunto de validação é utilizado para estimar os valores de μ e Σ através da *Estimativa de Máxima Verossimilhança* (Maximum Likelihood Estimation). O escore de anomalia $p^{(t)}$ de um vetor de erro $e^{(t)}$ é dado pelo valor da distribuição $\mathcal{N}$ em $e^{(t)}$, ou seja, $p^{(t)}$ é computado como $(e^{(t)} - \mu)^T \Sigma^{(-1)} (e^{(t)} - \mu)$ para uma observação $x^{(t)}$. Em $x^{(t)}$, o valor predito é considerado anômalo se $p^{(t)} > \tau$, senão, o ponto é considerado normal. O valor de τ é determinado utilizando o conjunto de validação v_A, com o objetivo de maximizar o escore F_β. A escolha de β no escore F_β depende da natureza do problema: $\beta < 1$ enfatiza a redução de falsos positivos, enquanto $\beta > 1$ enfatiza a redução de falsos negativos. Por fim, o modelo M é avaliado pelo escore F_β utilizando-se o conjunto t_A.

Essa técnica é mais simples de ser treinada e converge mais rápido que o modelo apresentado a seguir, devido ao uso mais natural do modelo de LSTM, que reconstrói uma sequência lendo cada elemento e predizendo os próximos. Contudo, este método apresenta uma desvantagem: a sua alta sensibilidade. Um único ponto anômalo na série temporal pode propagar-se e causar múltiplos erros de predição nos pontos de tempo subsequentes.

3.2 Encoder-Decoder

A arquitetura descrita nesta seção é constituída por um modelo *Encoder* baseado em LSTMs. Este *Encoder* aprende representações vetoriais de tamanho fixo para séries temporais passadas como *input*. A arquitetura inclui também um modelo *Decoder*, igualmente baseado em LSTMs, que utiliza essa representação vetorial para reconstruir a série temporal passada como *input* ponto a ponto. Em cada passo da reconstrução, o *Decoder* utiliza como *input* o estado interno atual dos nós de LSTM e o valor do ponto predito no passo precedente.

Para a criação e o treinamento deste modelo, utilizam-se os mesmos conjuntos s_N, V_N, v_A, t_A que foram apresentados na Seção 3.1. Essa arquitetura foi proposta por (MALHOTRA *et al.*, 2016) e é ilustrada na figura 10.

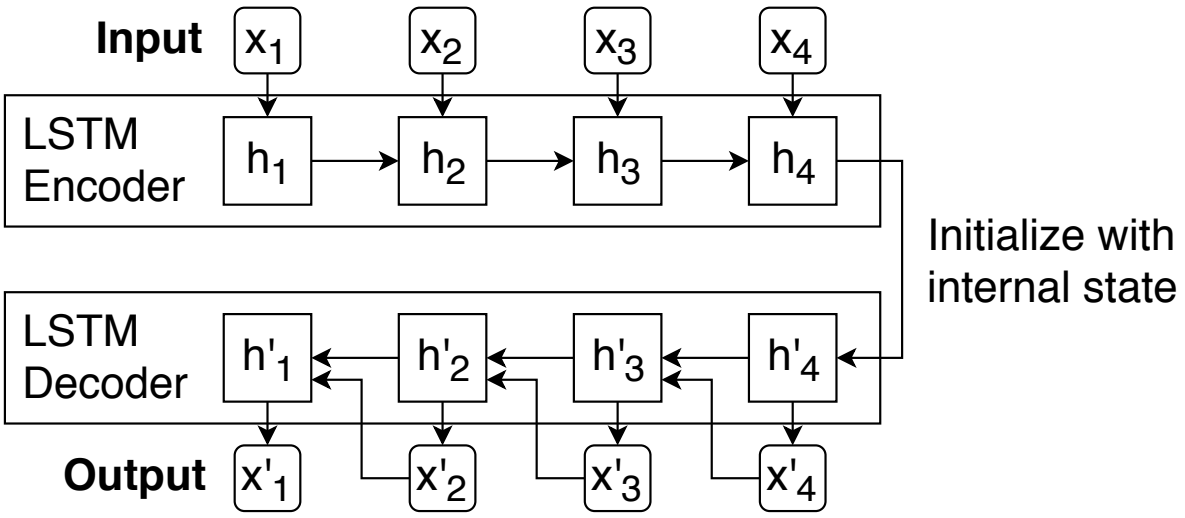

Figura 10 – Modelo de *Encoder-Decoder* proposto por (MALHOTRA *et al.*, 2016)

Dado um conjunto de treinamento s_N contendo séries temporais normais $X = [x^{(1)}, \ldots, x^{(n)}]$, o modelo de predição M é treinado da seguinte forma: Para cada ponto de tempo t_i (onde $i \in \{1, 2, ..., n\}$), $h_E^{(i)}$ representa o estado interno do *encoder*. Cada $h_E^{(i)}$ é um vetor em $\mathbb{R}^c$, onde c corresponde ao número de nós de LSTM na camada oculta do *encoder*. O *encoder* e o *decoder* são treinados em conjunto para reconstruir a série de *input* em ordem inversa, ou seja, ele é treinado para que $M(X) = \{x^{(n)}, x^{(n-1)}, \ldots, x^{(1)}\}$. O estado final do *encoder*, $h_E^{(n)}$, é então utilizado como o estado inicial do *decoder*. Adicionalmente, uma camada linear conectada à camada interna de LSTM do *decoder* é empregada para realizar as predições dos valores. Durante a fase de decodificação, o *decoder* utiliza $x^{(i)}$ e o estado interno das LSTMs $h_D^{(i-1)}$ para gerar o valor $x^{`(i-1)}$ correspondente ao objetivo $x^{(i-1)}$. Considerando que s_N é um conjunto de séries temporais sem anomalias, o modelo *Encoder-Decoder* é treinado para minimizar a

seguinte função objetivo: $\sum_{X \in s_N} \sum_{i=1}^{n} \left\| x^{(i)} - x^{`(i)} \right\|^2$. Por fim, para uma observação $x^{(t)}$, o escore de anomalia $p^{(t)}$ em (MALHOTRA *et al.*, 2016) é computado de forma similar à abordagem apresentada e explicada na Seção 3.1.

Diferentemente do modelo de LSTM empilhada, este modelo apresenta a vantagem de processar a totalidade da série temporal antes de efetuar sua reconstrução. Essa característica torna as anomalias individuais menos impactantes na precisão da reconstrução dos pontos de tempo vizinhos, resultando em reconstruções mais estáveis, mesmo em séries temporais que contenham anomalias. A contrapartida deste benefício é um processo de treinamento mais intensivo e uma convergência potencialmente mais demorada.

4 ENSEMBLE DE MODELOS PREDITIVOS EM SÉRIES TEMPORAIS PARA DETECÇÃO DE ANOMALIAS

4.1 Introdução

Neste capítulo, apresentaremos um ensemble de modelos preditivos, denominado TSPME-AD, que utiliza os modelos propostos por (MALHOTRA *et al.*, 2015) e (MALHOTRA *et al.*, 2016), com variações em seus hiper-parâmetros. O sistema calcula os escores de anomalias para cada ponto das séries temporais, aplica uma função de normalização nestes escores, combina-os utilizando uma média ponderada dos valores e emprega um *threshold* para discriminar entre pontos anômalos e normais das séries. O objetivo do ensemble é aproveitar modelos que apresentam comportamentos diferentes quando aplicados sobre uma mesma série temporal, realizando um processo semelhante a uma votação ponderada. Isto é, quando mais modelos, ou modelos com maior peso de importância, concordam sobre a classificação de um ponto como anômalo ou normal, obtemos um resultado mais preciso e estável sobre sua real natureza.

Existem alguns desafios no processo de combinação do ensemble e, de acordo com o trabalho de (AGGARWAL, 2013), os principais são a normalização e a função de combinação. A normalização aborda o problema de diferentes modelos gerarem *outputs* em escalas diferentes ou até em formatos diferentes que não são facilmente comparáveis. Já o segundo desafio consiste em descobrir qual a melhor função de combinação a ser utilizada (Mínimo, Máximo, Média, etc.). Essas são questões ainda em aberto, conforme apontado por (AGGARWAL, 2013). Apesar do vasto uso de ensembles em outros contextos na literatura, como classificações e regressões, os trabalhos em análises de anomalias utilizando ensembles ainda são muito esparsos, e, consequentemente, as soluções para esses tipos de problemas não são completamente conhecidas.

Este capítulo será estruturado nas seguintes seções: Na seção 4.2, a arquitetura geral do TSPME-AD será apresentada e explicada. Na seção 4.3, serão apresentadas algumas funções de combinação além da proposta para o modelo, explicando o processo de treinamento dos pesos para a função ponderada, e apresentando o processo de descoberta do *threshold* para a discriminação das anomalias. Na seção 4.4, serão apresentados os conjuntos de dados utilizados na experimentação, os processamentos realizados nos dados, e os resultados obtidos pelo TSPME-AD em comparação com os modelos individuais utilizados e outras funções de combinação. Por fim, na seção 4.5, serão apresentadas algumas conclusões sobre o modelo e os

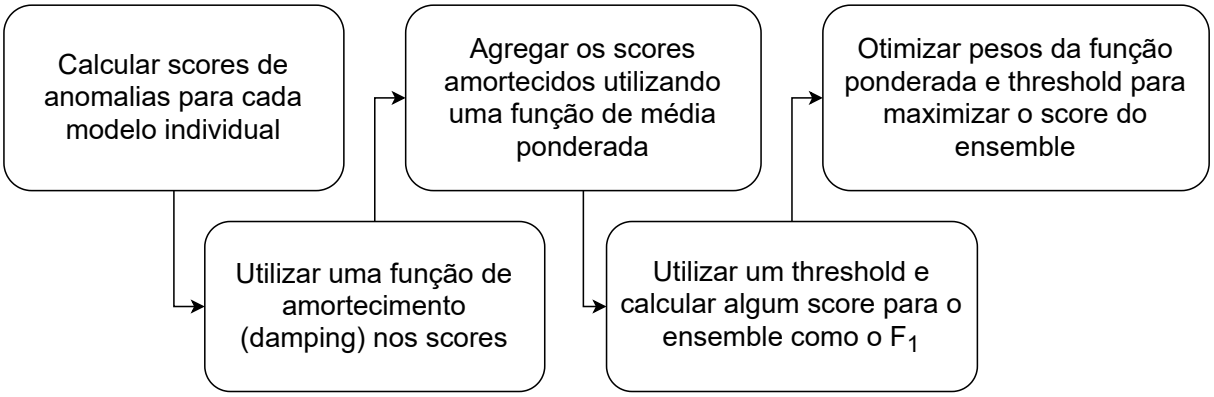

Figura 11 – Passo a passo completo da arquitetura do TSPME-AD

resultados.

4.2 Arquitetura

Uma das principais diferenças entre o ensemble proposto e outros ensembles está na não junção dos resultados dos modelos para gerar uma predição melhorada. Ao invés disso, os modelos treinam e predizem as séries individualmente, calculando seus escores de anomalias conforme proposto em seus respectivos trabalhos originais. O TSPME-AD é aplicado apenas após a geração dos escores de anomalias de cada modelo, seguindo os cinco passos representados pela figura 11 que são:

- Para cada série temporal, utilizar os modelos treinados para reconstruí-las e calcular os escores de anomalia de cada ponto da série
- Utilizar uma função de amortecimento para normalizar os escores de anomalia entre modelos
- Utilizar uma função de média ponderada para unir os escores de anomalias entre modelos
- Escolher um *threshold* que irá discriminar, baseado no resultado da função ponderada, os pontos anômalos e normais, e calcular um score para o ensemble como o F_1
- Otimizar os pesos da função de média ponderada e o *threshold*, visando maximizar o score do ensemble

Sejam os conjuntos de séries temporais s_N, v_A e t_A, onde s_N possui apenas séries sem anomalias, e os outros dois possuem algumas séries com pontos anômalos. Para os modelos preditivos $M_1, M_2, \ldots, M_p$, todos treinarão sobre o conjunto s_N de modo que, dada uma série temporal $X = [x_1, \ldots, x_n] \in s_N$, e um modelo qualquer M, o objetivo é que $M(X) = X$. Como as reconstruções das séries não são perfeitas, se $M(X) = Y$, a diferença $Y - X$ será um vetor de erros $E = [\varepsilon_1, ..., \varepsilon_n]$, onde $\varepsilon \in \mathbb{R}^{\geq}$ é um vetor m-dimensional, com m sendo o número de

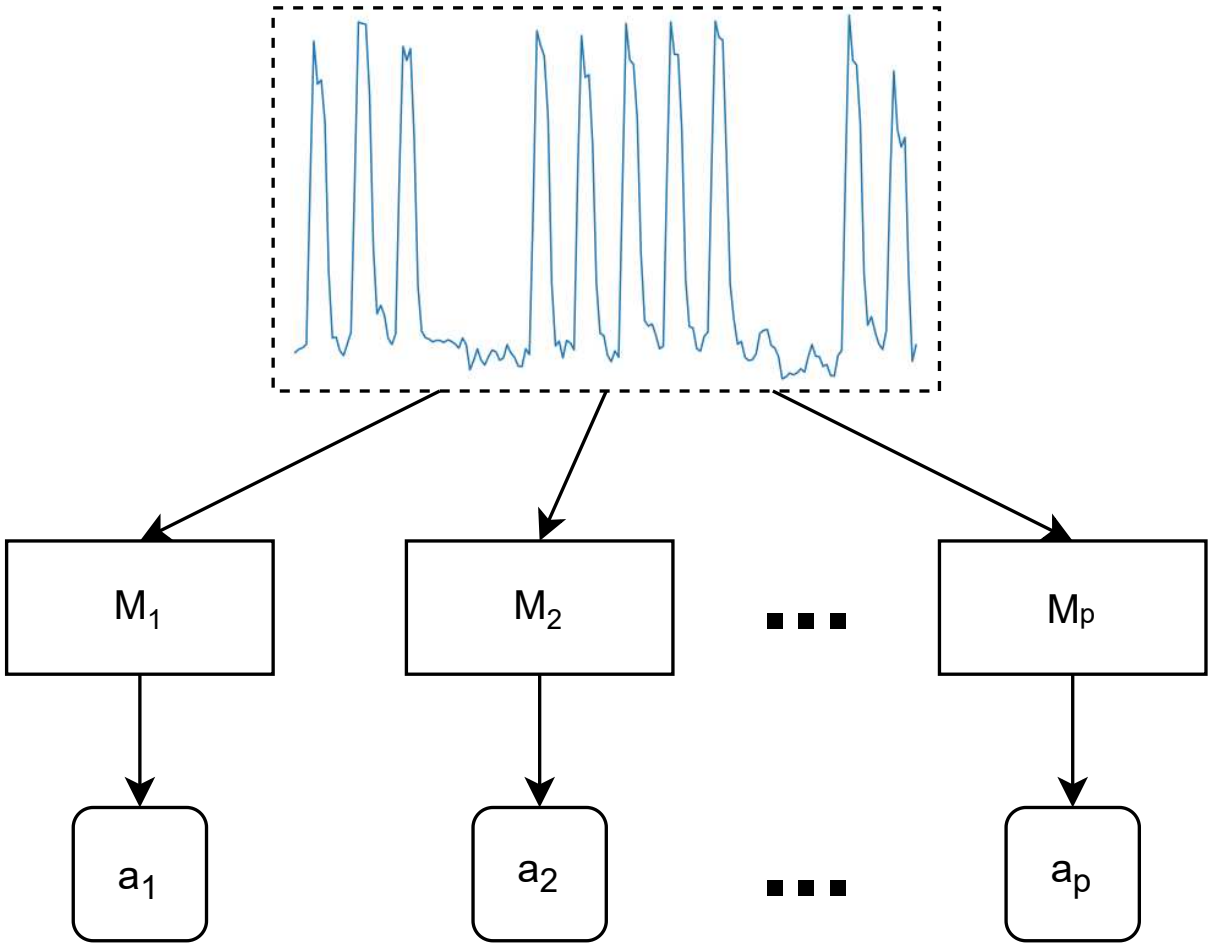

Figura 12 – Primeiro, todos os modelos tentam reconstruir a série temporal, e são calculados os escores de anomalia

dimensões das séries passadas como *input*.

Com isso, cria-se uma distribuição normal dos vetores de erro. Se $E_i^{(j)}$ é o conjunto de vetores de erro gerados por $M^{(j)}(X_i)$, é criada a distribuição $\mathcal{N}(\vec{\mu}, \Sigma)$, onde $\vec{\mu}$ é o vetor médio de todos os vetores de erro $\varepsilon \in E_i^{(j)}, \forall i, j$ e Σ é a matriz de covariância desses vetores. A partir dessa distribuição, os escores de anomalia são calculados utilizando a distância de Mahalanobis (MAESSCHALCK *et al.*, 2000), que mede a distância de um vetor $\vec{x}$ a uma distribuição com vetor médio $\vec{\mu}$ e matriz de covariância Σ, por meio da função $\sqrt{(\vec{x}-\vec{\mu})^{\mathbf{T}}\Sigma^{-1}(\vec{x}-\vec{\mu})}$, cujo resultado é um valor pertencente ao conjunto dos Reais. Assim, para cada $X = [x_1, \ldots, x_n] \in s_N$ e para cada modelo M, é gerado um vetor de escores de anomalias $\vec{A} = [a_1, \ldots, a_n]$, como representado na figura 12.

Como os escores de anomalias são gerados a partir da distância dos vetores de erro para uma distribuição de vetores, caso o erro desvie completamente do padrão, o escore de anomalia pode se tornar muito alto, com a possibilidade de sobrepor escores de outros modelos no passo de combinação do ensemble (AGGARWAL, 2013). Para diminuir o impacto

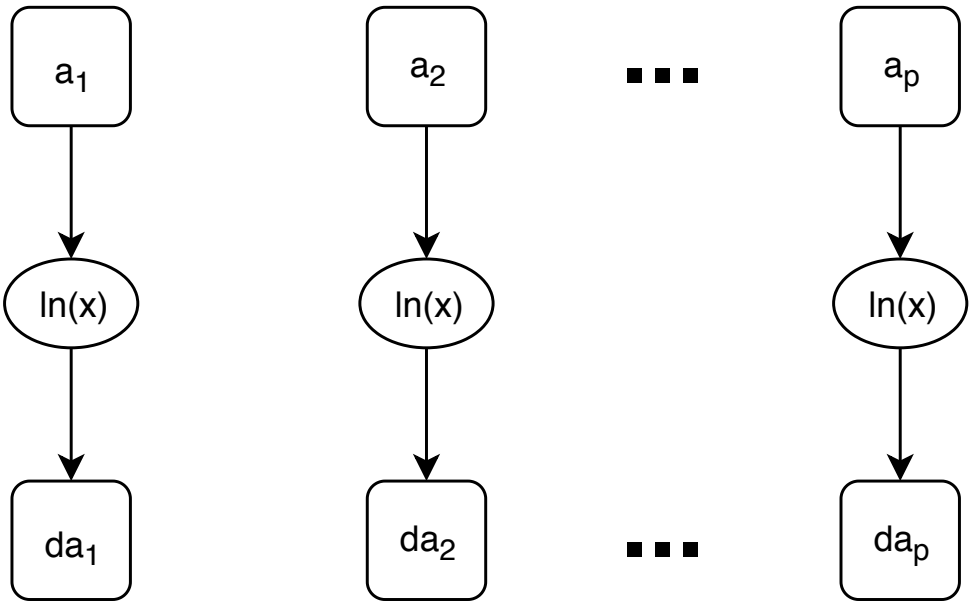

Figura 13 – Aplicando uma função de amortecimento nos escores de anomalias, como a função de logaritmo natural

desses escores, utilizamos uma função de amortecimento que diminui escores muito altos, como representado na figura 13. Várias funções podem ser utilizadas para esse propósito, como $f(x) = \sqrt{x}$, $f(x) = 1 - \frac{1}{x+1}$ e a função logarítmica $f(x) = \ln(x+1)$. Esta última será utilizada no TSPME-AD por ter uma boa capacidade de amortecer escores muito altos e por apresentar a vantagem de não limitar os valores a um intervalo. Isso mantém um certo impacto de escores mais altos que não devem ser descartados, sem fazê-los sobrepor de forma absoluta os outros escores. Além disso, com a adição de 1 no argumento, o escore amortecido sempre será positivo e permite $a_i = 0$.

4.3 Combinação e Discriminação

Com os escores de anomalias calculados e amortecidos, o próximo passo é combiná-los utilizando alguma função de combinação. Diversas funções podem ser utilizadas, como o máximo, o mínimo, a média simples, média harmônica, média ponderada, entre outras. Funções como máximo e mínimo possuem desvantagens quando existem escores dissonantes. Por exemplo, se a grande maioria dos modelos aponta um escore baixo e apenas um dos modelos aponta um escore alto, ao utilizar a função de máximo, os modelos que formam a maioria seriam descartados. O mesmo ocorre na situação inversa quando se utiliza a função de mínimo. Já as funções de média simples e média harmônica podem sofrer com alguns modelos "ruins", pois atribuem o mesmo peso a todos os modelos participantes do ensemble. Ou seja, se uma

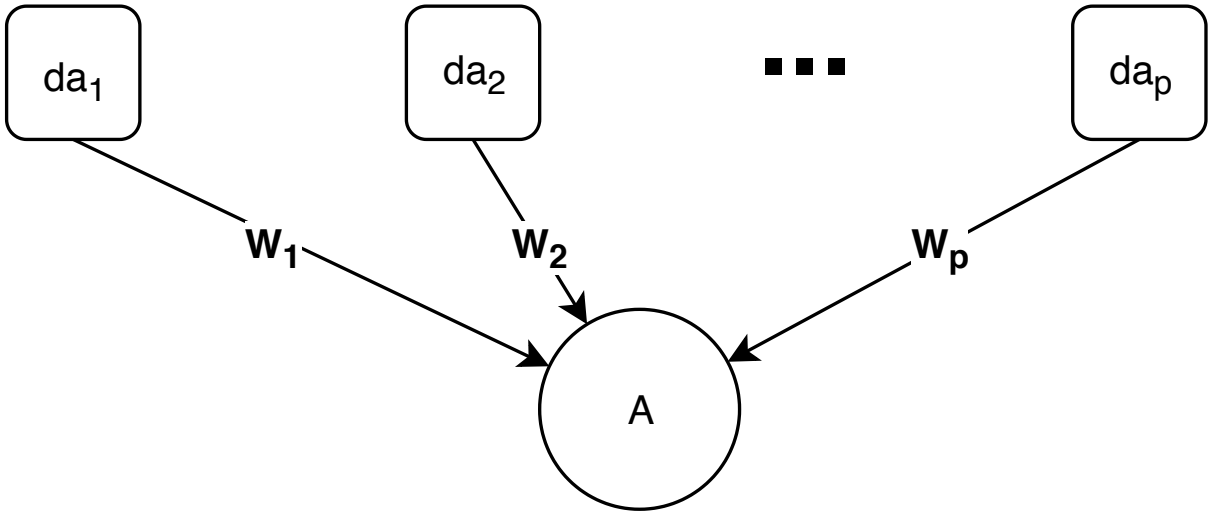

Figura 14 – Agregação dos conjuntos de escores de anomalia utilizando uma função de média ponderada

pequena minoria de modelos obtiver escores de anomalia muito divergentes dos demais, eles podem dominar o valor do escore resultante, influenciando negativamente na discriminação das anomalias.

O TSPME-AD utiliza uma função de combinação de média ponderada, que atribui e treina pesos para cada modelo do ensemble, como representado na figura 14. Com isso, modelos que não conseguem obter resultados satisfatórios na contribuição para o escore de anomalia final têm sua influência atenuada com um peso baixo, enquanto modelos eficientes são beneficiados com pesos altos.

O passo de combinação pode ser definido como: Sejam $M_{(1)}, M_{(2)}, \ldots, M_{(p)}$ modelos preditivos para reconstrução de série temporal, $X = [x_1, \ldots, x_n]$ uma série temporal, e $[a_1^{(j)}, \ldots, a_n^{(j)}]$ os escores de anomalia obtidos através dos erros de predição de $M^{(j)}(X)$ para cada $j \leq p$. Considerando $[w^{(1)}, w^{(2)}, \ldots, w^{(p)}]$ como os pesos atribuídos a cada modelo do ensemble, com $w^{(j)} \in [0,1]$, o score de anomalia α_i referente ao ponto $x_i \in X$ é definido por

$$\alpha_i = \frac{w^{(1)} \ln(a_i^{(1)} + 1) + w^{(2)} \ln(a_i^{(2)} + 1) + \cdots + w^{(p)} \ln(a_i^{(p)} + 1)}{w^{(1)} + w^{(2)} + \cdots + w^{(p)}} \geq 0 \tag{4.1}$$

Com isso, dada a série temporal $X = [x_1, \ldots, x_n]$, o TSPME-AD computa os escores de anomalias $[\alpha_1, \ldots, \alpha_n]$. Assim, as anomalias serão discriminadas a partir de um *threshold* $\tau > 0$, onde para cada $1 \leq i \leq n$, se $\alpha_i \leq \tau$, o ponto x_i é considerado normal, e caso contrário, se $\alpha_i > \tau$, x_i é considerado anômalo, como exemplificado na figura 15.

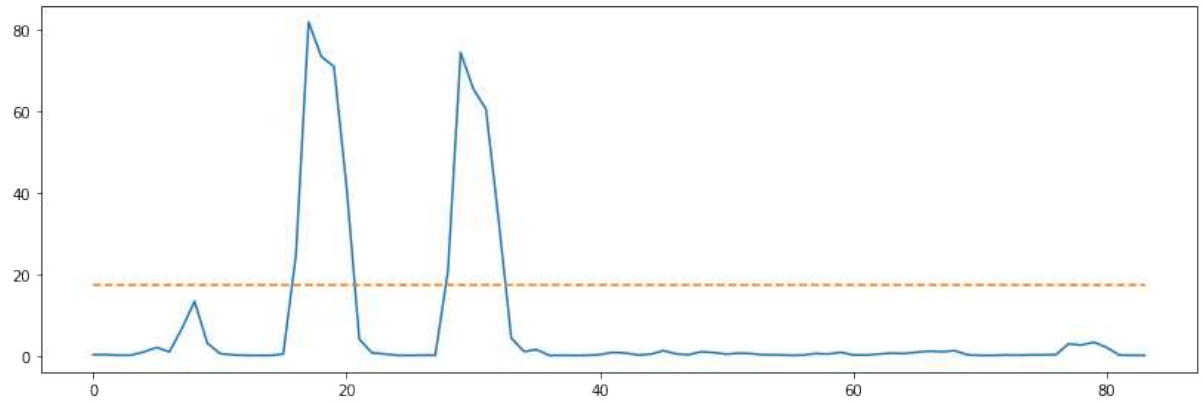

Figura 15 – Utilizando um *threshold* para discriminar entre pontos normais e anômalos na série

Por último, é necessário descobrir os pesos e *threshold* ótimos para a maximização do escore desejado. Para isso, utilizamos o conjunto v_A definido na seção 4.2, que contém séries com anomalias anotadas, e treinamos pesos e um *threshold* que maximizem o escore F_1 do ensemble. Um processo de Busca Randômica Direcionada (*Directed Random Search*), proposto por (SEIFFERT; MICHAELIS, 2001) como método alternativo de treinamento de redes neurais, é aplicado aos pesos limitados entre 0 e 1 e simultaneamente ao *threshold* $\tau \geq 0$, visando a maximização do score F_1 sobre o conjunto de dados v_A.

Outras técnicas de otimização de pesos foram consideradas, como a Otimização por Enxame de Partículas (*Particle Swarm Optimization*) proposta em (KENNEDY; EBERHART, 1995), mas verificou-se que a técnica de busca randômica converge mais rapidamente a um score ótimo, além de trabalhar melhor com valores não limitados, já que o *threshold* também faz parte do campo de busca e, diferentemente dos pesos dos modelos, pode ser maior que um.

4.4 Experimentação

Nesta seção, apresentamos os experimentos realizados sobre dois conjuntos de dados reais e comparamos os escores F_1 obtidos pelos modelos propostos por (MALHOTRA *et al.*, 2015; MALHOTRA *et al.*, 2016), descritos no capítulo 3. Utilizamos variações nos hiperparâmetros dos modelos para obter maior diversidade nos resultados de reconstrução das séries temporais e, consequentemente, melhorar o desempenho dos ensembles. Também analisamos diferentes funções de combinação dos escores de anomalias para avaliar seus impactos sobre o score final.

4.4.1 Datasets

Para a experimentação, utilizamos dois conjuntos de dados reais: uma série temporal de demanda de energia elétrica durante um ano e um conjunto de eletrocardiogramas disponibilizados pela Universidade do MIT. Ambos os conjuntos são detalhados nas subseções 4.4.1.1 e 4.4.1.2, respectivamente.

4.4.1.1 Power Demand

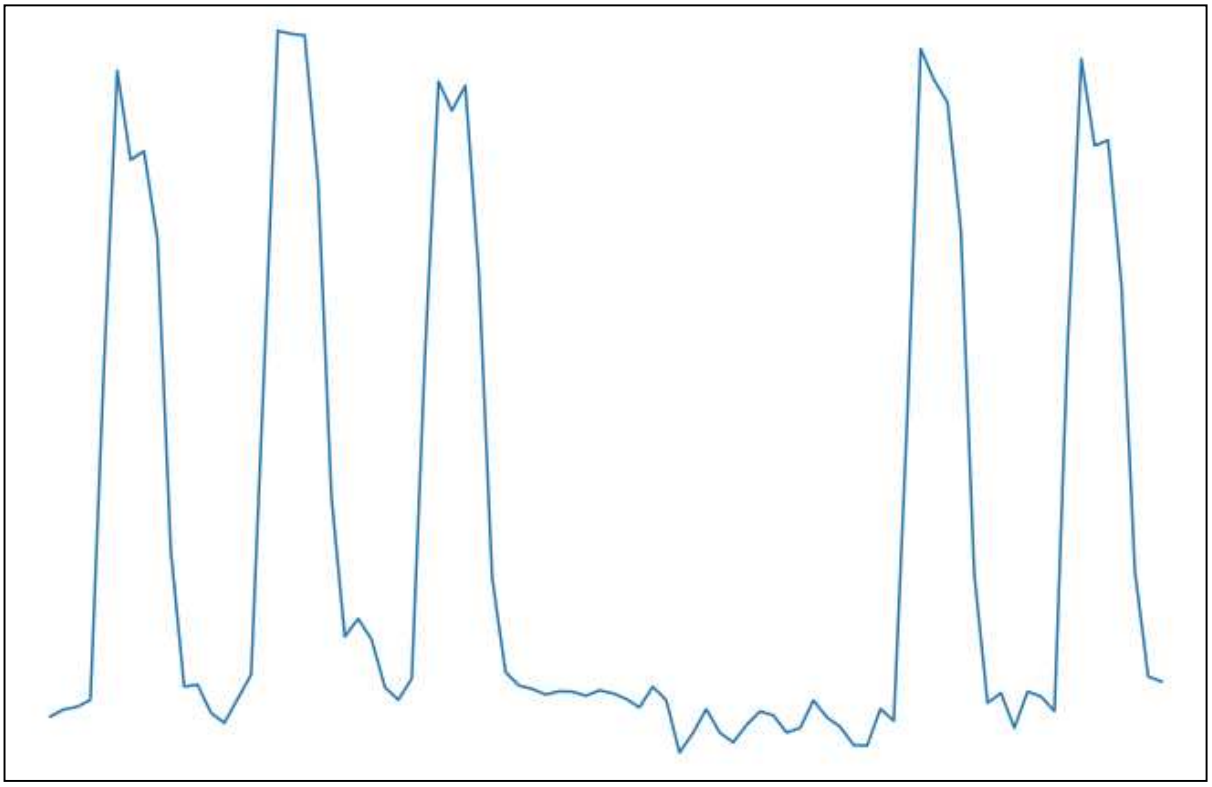

Figura 16 – Uma semana normal de demanda de energia elétrica, iniciando-se na quarta-feira

O conjunto de dados de demanda de energia elétrica fornecido por (KEOGH *et al.*, 2007) apresenta o registro de um ano de coleta de dados sobre a demanda de energia. O comportamento normal dos dados caracteriza-se por alta demanda durante os dias úteis da semana e demanda mais baixa durante os finais de semana. Assim, altas demandas durante os finais de semana e baixas demandas em dias úteis indicam anomalias nos dados que são anotadas.

Com isso, realiza-se um pré-processamento nos dados em que é aplicada uma subamostragem com fator 8, ou seja, cada 8 pontos da série são convertidos em um, utilizando um algoritmo que mantém o formato da série o mais próximo possível do original. Em seguida, a série é dividida em janelas sem interseções com 84 pontos cada, representando exatamente

uma semana de demanda, reproduzindo o mesmo processamento realizado por (MALHOTRA *et al.*, 2016). A figura 16 apresenta uma semana de comportamento normal da série, iniciando na quarta-feira.

Para a experimentação, os conjuntos s_n e v_a foram construídos a partir dos primeiros 40% dos pontos da série anual, e o restante constituiu o conjunto t_a. Assim, uma parte maior da série foi utilizada para a etapa de testes, pois, como anomalias são eventos relativamente raros, necessitamos de mais pontos anômalos no conjunto de teste para obter um score mais representativo.

4.4.1.2 MIT Electrocardiogram Dataset

O conjunto de dados de eletrocardiogramas do MIT é um conjunto de dados reais obtidos a partir de múltiplas horas de registros dos batimentos cardíacos por sensores, em diversos pacientes, como representado na Figura 17. Cada eletrocardiograma é composto por dois canais, caracterizando o dado como uma série temporal multivariada. Esses dados também incluem anotações de tipos de eventos que podem ocorrer nos batimentos, as quais serão utilizadas para definir as anomalias nos conjuntos de validação e teste dos experimentos.

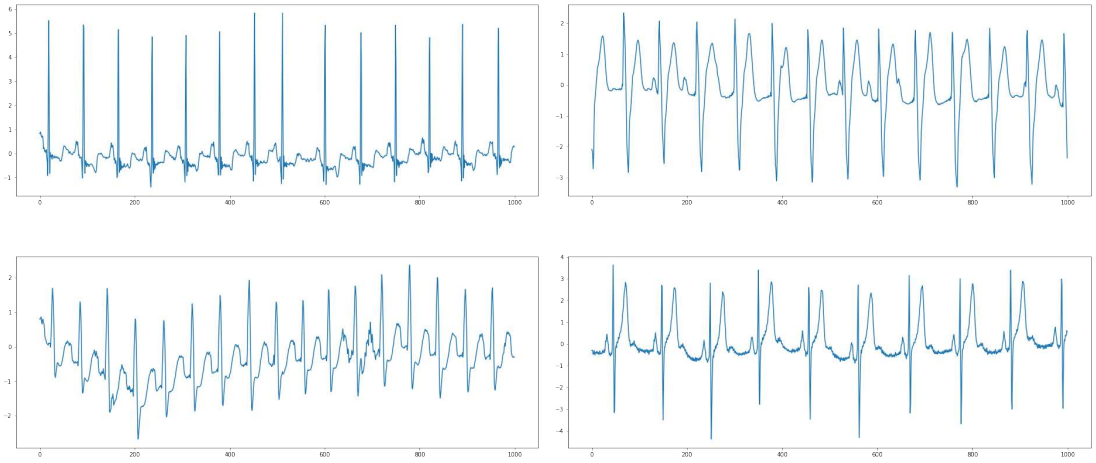

Figura 17 – Exemplos de partes dos dados dos electrocardiogramas dos dados do MIT

Para os experimentos deste capítulo será utilizado um dos eletrocardiogramas dos dados do MIT, o *mitdbx_108*, que também é fornecido e utilizado em (KEOGH *et al.*, 2007), por ser o mesmo conjunto de dados empregado para o teste dos modelos de (MALHOTRA *et al.*, 2015; MALHOTRA *et al.*, 2016) que servirão como base para a construção do ensemble. Esse eletrocardiograma apresenta três anomalias distintas e, diferentemente dos dados de demanda

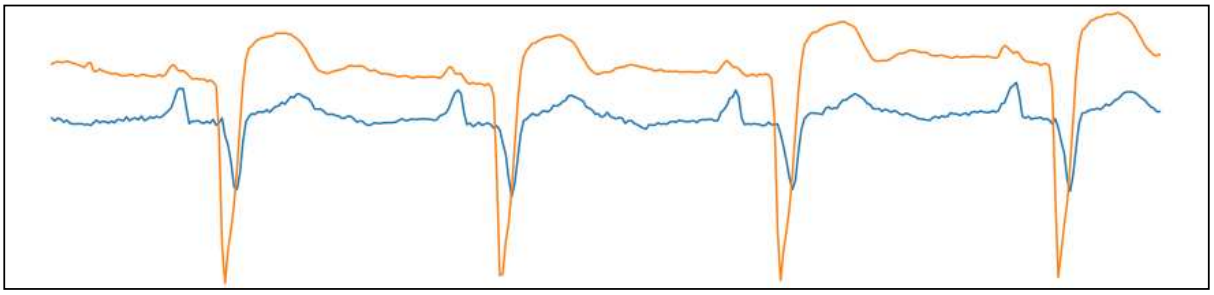

Figura 18 – Quatro batimentos cardíacos de um dos eletrocardiogramas do conjunto de dados

de energia, não é tão bem comportado. Enquanto os dados de energia apresentam um ciclo bem definido de semanas com uma quantidade exata de pontos entre os ciclos, as séries dos eletrocardiogramas podem apresentar batimentos espaçados de forma irregular, pois o coração pode acelerar ou desacelerar normalmente sem apresentar anomalias, e possíveis anomalias podem ocorrer em pontos aleatórios da série.

No pré-processamento, assim como nos dados de demanda de energia, a série passará por um processo de subamostragem com fator de 4 e será dividida em janelas com 93 pontos cada, o que representa da melhor forma possível um ciclo de batimento cardíaco do paciente. Por exemplo, a Figura 18 mostra quatro ciclos de batimentos cardíacos, ou quatro janelas em sequência da série. Por fim, também como realizado no conjunto de dados anterior, o eletrocardiograma será dividido de forma que os primeiros 40% da série gerarão os conjuntos s_n, v_a para treinamento dos modelos individuais, distribuição normal dos vetores de erro e pesos com *threshold* do ensemble, e o restante da série formará o conjunto t_a para teste e comparação dos escores.

4.4.2 Resultados

Nesta subseção, serão apresentados os resultados obtidos pelo TSPME-AD, suas variantes e seus modelos individuais. O modelo proposto será comparado com os resultados de variações nos modelos de LSTM empilhada ou *Stacked LSTM (SL)* e do *Encoder-Decoder (ED)* propostos por (MALHOTRA *et al.*, 2015; MALHOTRA *et al.*, 2016) respectivamente. Também serão avaliados os resultados provenientes da combinação dessas técnicas utilizando as funções de combinação:

- Média Simples (SA);
- Média atenuada (DA);
- Média ponderada simples (SWA);

- Média ponderada atenuada (TSPME-AD).

Como mencionado anteriormente, para a avaliação dessas técnicas de detecção de anomalias, serão utilizados os conjuntos de dados de demanda de energia elétrica e de um eletrocardiograma. Os desempenhos serão medidos em termos do escore F_β com $\beta \geq 0$, tal que:

$$F_\beta = (1+\beta^2)\frac{precision \times recall}{(\beta^2 precision) + recall} \tag{4.2}$$

em que o valor de β define o peso dado à proporção de falsos positivos em comparação com a proporção de falsos negativos, sendo que um $\beta > 1$ tende a dar mais peso ao *recall*, enfatizando a minimização do número de falsos negativos, e um $\beta < 1$ confere mais ênfase na minimização de falsos positivos. Nos resultados apresentados, serão comparados tanto os valores de *precision* e *recall*, quanto os escores F_1 e $F_{0.1}$, em que a utilização do score F_1 ocorre por ser mais comum na literatura como forma de comparação de modelos de classificação, e o $F_{0.1}$ traz uma ênfase significativa na precisão, pois como os "positivos"são anomalias nas séries, e anomalias tendem a ser relativamente raras, um alto número de falsos positivos acaba por descredibilizar o modelo de detecção, mas encontrar uma anomalia dentre algumas que possam existir já traz uma informação importante do comportamento da série, como argumentado por (MALHOTRA *et al.*, 2015), que utiliza o score $F_{0.1}$ para realizar suas comparações.

4.4.2.1 Resultados sobre os dados de demanda de energia elétrica

O conjunto de dados de demanda de energia é uma série temporal periódica, ou seja, o número de pontos por ciclo é constante durante toda a série. Isso ajuda no processo de reconhecimento de padrão e detecção de anomalias por permitir a quebra da série em janelas de tamanho constante que representem bem o padrão normal de um ciclo da série.

A tabela 1 apresenta o *precision*, *recall*, e os escores $F_{0.1}$ e F_1 para todos os modelos base e variações em seus hiperparâmetros, além do modelo de ensemble com as diferentes funções de combinação, treinados e testados com os dados de demanda de energia. Como podemos observar, os ensembles que utilizam as funções de combinação Média Simples (SA) e Média Atenuada (DA) alcançaram os melhores resultados em termos dos escores $F_{0.1}$ e F_1, respectivamente, com o modelo proposto do TSPME-AD obtendo o segundo melhor resultado em ambos os escores.

O experimento mostra que a estratégia do TSPME-AD com a função de combinação

Tabela 1 – Resultados do teste sobre os dados de demanda de energia elétrica

MODELO[a]	Precision	Recall	$F_{0.1}$	F_1
SL [K = 2]	4.42%	77.78%	0.04	0.08
SL [K = 4]	5.49%	77.78%	0.05	0.10
SL [K = 8]	22.86%	44.44%	0.22	0.30
SL [K = 16]	12.77%	66.67%	0.12	0.21
ED [H = 16]	47.06%	44.44%	0.47	0.45
ED [H = 32]	3.39%	22.22%	0.03	0.05
ED [H = 64]	59.09%	72.22%	0.59	0.65
ED [H = 128]	18.52%	27.78%	0.18	0.22
SA	100.0%	44.44%	0.98	0.61
DA	76.19%	88.89%	0.76	0.82
SWA	25.00%	72.22%	0.25	0.37
TSPME-AD	76.47%	72.22%	0.76	0.74

[a] **SL**: Stacked LSTM, **ED**: Encoder Decoder, **SA**: Simple Average Ensemble, **DA**: Damped Average ensemble, **SWA**: Simple Weighted Average Ensemble, **TSPME-AD**: Time Series Prediction Model Ensemble for Anomaly Detection.

de média ponderada atenuada, apesar de não apresentar os melhores escores em todos os cenários, ainda consegue obter bons resultados, sendo superior a todos os modelos individuais que o compõem em ambos os escores $F_{0.1}$ e F_1. Também podemos observar a diferença que a função de atenuação proporciona no resultado final quando comparamos os escores da média simples (SA) e da média atenuada (DA), que, apesar de introduzir mais falsos positivos na detecção, compensa na significativa redução de falsos negativos. Já os ensembles que utilizaram funções de combinação ponderadas apresentaram um resultado um pouco inferior, provavelmente devido às poucas instâncias de anomalias no conjunto de validação que é utilizado para treinar os pesos, aumentando assim a possibilidade de *overfitting*.

4.4.2.2 Resultados sobre o conjunto de dados de eletrocardiogramas

Como a duração de um ciclo em um eletrocardiograma, que representa um ciclo completo de batimento cardíaco, varia de uma instância de electrocardiograma para outras, e até dentro do mesmo eletrocardiograma, esse tipo de série temporal é chamada de *quasi-periódica*. Essa classe de séries temporais é mais difícil de lidar e dificulta a construção de modelos preditivos, pois, além de apresentar um padrão mais difícil de identificar, como os modelos esperam receber janelas de tamanho igual como entrada (*input*), é necessário definir um tamanho de janela que consiga representar da melhor forma possível os ciclos da série em cada janela, conforme realizado em (MALHOTRA *et al.*, 2016).

Como na seção anterior, a Tabela 2 apresenta o *precision*, *recall*, e os escores $F_{0.1}$ e F_1 para todos os modelos base e para o modelo de ensemble com as mesmas funções de agregação utilizadas na subseção 4.4.2.1. Diferentemente da experimentação no conjunto de dados de demanda de energia, o TSPME-AD conseguiu atingir os melhores resultados em relação aos escores de *precision*, $F_{0.1}$ e F_1, com uma melhoria sobre o escore F_1 de aproximadamente 12.8% quando comparado ao segundo melhor resultado, obtido utilizando o ensemble com a função de agregação de média simples ponderada (SWA), e um resultado 22.2% superior quando comparado com o melhor dos modelos individuais.

Neste experimento, as outras funções de combinação, como a **SA**, **DA** e **SWA**, não conseguiram um resultado tão bom na combinação dos modelos de (MALHOTRA *et al.*, 2015; MALHOTRA *et al.*, 2016). Isso pode ser explicado devido ao baixo desempenho dos modelos baseados em *encoder-decoder*, pois tanto quando não se utilizam pesos para a combinação dos modelos quanto quando não se utiliza uma função de amortecimento para estabilizar escores muito altos, modelos de qualidade inferior, como todas as variações do modelo de *encoder-decoder* no caso deste experimento, podem afetar negativamente o resultado do ensemble.

O motivo para a baixa efetividade do modelo de *encoder-decoder* deve-se provavelmente ao fato de o eletrocardiograma ser uma série quasi-periódica, ou seja, com a maior variabilidade dos ciclos representados pelas janelas, o modelo apresenta uma dificuldade significativamente maior de reconstruí-las. No caso do LSTM empilhado, como ele processa ponto a ponto para fazer as predições, é mais fácil adaptar-se a mudanças sutis na série. Já o *encoder-decoder* lê a janela por completo e a representa com seu estado interno ao fim da leitura, tornando assim o processo de aprendizado do padrão muito mais complexo diante dessa variabilidade da janela.

4.5 Conclusão

Até o momento, não se tem conhecimento de trabalhos anteriores que tenham proposto ensembles baseados em modelos para detecção de anomalias em séries temporais que utilizassem essas funções de combinação ou modelos baseados em LSTMs (AGGARWAL, 2013; LIU *et al.*, 2012). Existem algumas técnicas similares que propõem modelos de ensemble para detecção de anomalias (AGGARWAL, 2013), como (LIU *et al.*, 2012; GAO; TAN, 2006), mas nenhuma dessas técnicas modela o comportamento padrão de conjuntos de dados utilizando LSTMs e seus benefícios na modelagem de séries temporais multidimensionais.

Tabela 2 – Resultados dos testes no electrocardiograma

MODELS[a]	Precision	Recall	$F_{0.1}$	F_1
SL [K = 2]	22.37%	47.80%	0.22	0.30
SL [K = 4]	18.37%	57.07%	0.18	0.28
SL [K = 8]	20.97%	48.29%	0.21	0.29
SL [K = 16]	42.28%	30.73%	0.42	0.36
ED [H = 16]	7.02%	100%	0.07	0.13
ED [H = 32]	7.36%	100%	0.07	0.14
ED [H = 64]	7.37%	100%	0.07	0.14
ED [H = 128]	7.37%	100%	0.07	0.14
SA	30.84%	48.29%	0.31	0.38
DA	11.33%	60.00%	0.11	0.19
SWA	34.05%	46.34%	0.34	0.39
TSPME-AD	41.00%	47.80%	0.41	0.44

[a] **SL**: LSTM empilhada, **ED**: Encoder-Decoder, **SA**: Ensemble de Média Simples, **DA**: Ensemble de Média Amortecida, **SWA**: Ensemble de Média Ponderada

Podemos concluir, principalmente com base nos resultados apresentados na seção 4.4.2.2, que a técnica de combinação utilizada pelo TSPME-AD (que utiliza tanto o amortecimento dos escores de anomalias quanto uma média ponderada para os escores dos modelos) consegue compensar resultados inferiores de alguns modelos base e pode produzir um ensemble com resultados de qualidade superior quando comparado tanto com a utilização de outras funções de agregação quanto com os modelos base utilizados.

Também é válido mencionar que o TSPME-AD, em geral, obtém melhores resultados que os dois modelos do estado da arte apresentados por Malhotra *et al.* (2015), Malhotra *et al.* (2016), o que é esperado, já que está comprovado, como mencionado em (OPITZ; MACLIN, 1999), que ensembles de modelos de classificação que obtêm bons resultados individualmente e possuem discordâncias nos locais de seus erros em relação ao mesmo dado de entrada sempre obtêm melhores resultados que os modelos que os compõem. Os modelos individuais do **TSPME-AD** possuem certas diferenças nos locais de erros, pois o *LSTM empilhado* apresenta erros mais altos após a ocorrência de uma anomalia, e o *Encoder-Decoder* apresenta erros mais altos em locais anteriores às anomalias devido à sua característica de reconstrução de trás para frente. Com isso e os bons resultados obtidos por esses modelos, as premissas para um bom ensemble são atendidas, e os resultados são verificados neste trabalho.

Portanto, os resultados mostram que modelos de ensemble podem ser boas alternativas na detecção de anomalias em séries temporais, pois três modelos simples de ensemble

já obtêm resultados superiores aos modelos base utilizados sob o mesmo conjunto de dados, como demonstrado na Tabela 2. Também é demonstrado que diferentes funções de combinação podem apresentar resultados completamente distintos, como a Média Ponderada, que obteve um resultado inferior aos outros modelos nos dados de demanda elétrica, e a Média Amortecida, com um resultado inferior no conjunto de dados de eletrocardiogramas. O TSPME-AD obteve bons resultados em comparação com os outros em ambos os cenários, tornando-o uma boa escolha de modelo de forma geral.

5 ENCODER-DECODER DE REPRESENTAÇÃO CONCRETA E NOVAS TÉCNICAS PARA SEGMENTAÇÃO DE JANELAS

5.1 Introdução

No capítulo anterior, foram realizadas análises de anomalias em séries temporais utilizando ensembles, que aplicam modelos já propostos e os combinam para obter melhores resultados. Neste capítulo, a análise será estendida pela apresentação de um novo modelo de detecção de anomalias em séries temporais utilizando LSTMs, que modifica a estrutura do modelo proposto em (MALHOTRA *et al.*, 2016), no qual o modelo continua sendo um *auto-encoder*, mas a representação interna da série analisada será representada por uma camada densa em vez do estado interno final do *encoder*.

Como demonstrado nos resultados do modelo de *encoder-decoder* na Tabela 2, o *encoder-decoder* apresenta resultados inferiores em séries quasi-periódicas devido ao problema de representação dos ciclos na segmentação da série em janelas. Por isso, este capítulo também apresentará técnicas de segmentação de janelas que utilizam características específicas da série para definir a melhor separação entre os ciclos, permitindo segmentar as janelas de acordo com eles.

Com isso, como as técnicas de segmentação de janelas serão avaliadas nesse capítulo, não é adequado avaliá-las no conjunto de dados de demanda de energia elétrica, visto que se trata de uma série periódica com um ciclo bem definido de uma semana, com uma quantidade exata de pontos em todos os ciclos. Apesar disso, será utilizado todo o conjunto de dados de eletrocardiogramas do MIT apresentado na seção 4.4.1.2, o que servirá tanto para comparar os modelos de predição em um cenário mais complexo com vários eletrocardiogramas diferentes, quanto para verificar o impacto de uma técnica eficiente de segmentação de janelas ao auxiliar os modelos preditivos.

Este capítulo será dividido da seguinte maneira: a Seção 5.2 apresentará o novo modelo baseado em encoder-decoder para detecção de anomalias; na Seção 5.3 serão apresentadas algumas técnicas de segmentação de janelas mais simples e será proposta uma segmentação de janela dinâmica; a Seção 5.4 apresenta a experimentação e os resultados comparados dos modelos e técnicas de segmentação de janelas; e, por fim, a Seção 5.5 apresenta as conclusões sobre as técnicas e os resultados.

5.2 Encoder-Decoder de Representação Concreta (CRED)

Esta proposta traz uma variação sobre o modelo de *Encoder-Decoder* proposto por Malhotra *et al.* (2016). Essa técnica anterior inicializa os pesos do estado interno do *decoder* com os pesos do estado interno do *encoder* após ler completamente uma janela da série temporal e, a partir disso, reconstrói a série fornecida como *input* ponto a ponto de trás para a frente, conforme (SUTSKEVER *et al.*, 2014), utilizando cada ponto reconstruído como *input* para a reconstrução do ponto seguinte, como apresentado na Figura 10.

O modelo proposto, Concrete Representation Encoder-Decoder (CRED), utiliza em sua arquitetura uma camada densa com número de nós menor que o tamanho da janela fornecida como *input*, a qual será a camada de *output* para o *encoder*. O número menor de nós na camada interna tem por objetivo obter uma representação mais generalizada da série passada como *input*, como uma análise de componentes principais ou *PCA*, para que anomalias em uma série passada como *input* tenham menos impacto na representação da série na camada interna. Com isso, o *decoder* inicia seus estados iniciais com zeros para manter a consistência das reconstruções e iniciará a reconstrução da série recebendo como *input* a camada densa, utilizando-a para reconstruir cada ponto na ordem normal da série, até que toda a janela seja reconstruída, como apresentado na Figura 19.

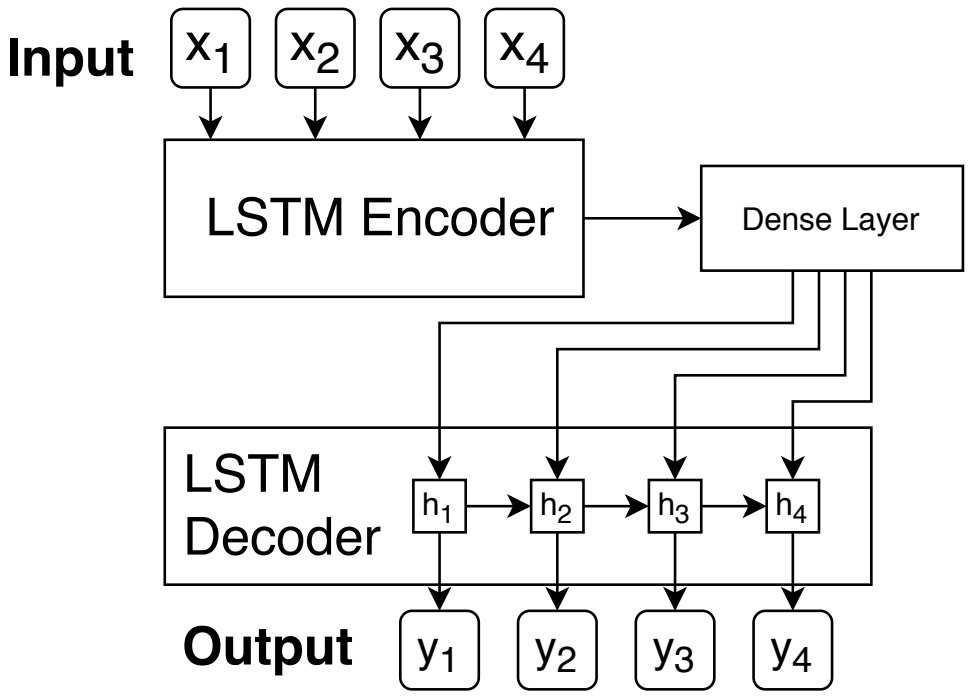

Figura 19 – Arquitetura do modelo proposto (CRED)

Seja $X = [x_1, ..., x_n]$ uma janela de tamanho n de uma série temporal, *ENC* a camada

LSTM de *encoder* do modelo, D a camada densa de tamanho k no centro do modelo, e DEC a camada LSTM de *decoder* do modelo. O primeiro passo é calcular $I = D(ENC(X))$, onde $I \in \mathbb{R}^k$ é a representação intermediária da série de *input*. Então, o vetor I é repetido n vezes, como $I_r = [I, ..., I]$ onde $||I_r|| = n$. Essa repetição acontece para que todos os pontos que forem ser reconstruídos pelo *decoder* tenham como *input* o vetor intermediário, evitando assim o "esquecimento"dessa representação intermediária pelos nós de LSTM, que pode acontecer no modelo proposto por (MALHOTRA *et al.*, 2016), onde o único momento em que o modelo possui a representação da série gerada pelo *encoder* é no passo inicial do processo de *decoding*, podendo assim dificultar a convergência do modelo no treinamento e afetar o resultado da reconstrução.

O score de anomalia é calculado exatamente como nos outros modelos apresentados, computando-se a matriz de erros $E = [e_1, ..., e_n]$ entre a janela original $X = [x_1, ..., x_n]$ e a janela reconstruída $Y = DEC(D(ENC(X))) = [y_1, ..., y_n]$, com $e_i = y_i - x_i$. Com isso, cria-se uma distribuição gaussiana multivariada $\mathcal{N}(\mu, \Sigma)$ sobre todos os vetores de erros individuais e calcula-se os scores de anomalia a partir da equação $\alpha_i = (e_i - \mu)^T \Sigma^{(-1)} (e_i - \mu)$ para cada ponto na janela da série temporal.

O modelo proposto por Malhotra *et al.* (2016) se aproxima mais de modelos *sequence-to-sequence*, comumente utilizados em predição de texto, mas quando o modelo é utilizado sobre dados de séries temporais, ele passa a ler janelas de input bem maiores. Com isso, conforme mencionado anteriormente, como a saída da camada de *encoder* se torna o estado interno inicial da camada de *decoder*, à medida que o tamanho da janela analisada aumenta, torna-se cada vez mais difícil manter a informação do estado interno inicial, pois este se altera a cada ponto processado.

Quando uma camada densa central é utilizada e repetida como *input* para todas as reconstruções do *decoder*, ela consegue manter a representação da janela obtida pelo *encoder* durante todo o processo de reconstrução, com a desvantagem da necessidade de um tamanho de janela fixo, definido antes do treinamento, mas com a vantagem de uma convergência mais fácil no treinamento e melhores resultados, como apresentados na seção 5.4.

5.3 Estratégias de segmentação de janelas

Nesta seção, serão apresentadas três técnicas de segmentação de janelas, que têm como objetivo dividir a série temporal em janelas semânticas que consigam representar um

ciclo da série e facilitar a descoberta de padrões e anomalias pelos modelos. A primeira técnica, apresentada na seção 5.3.1, é a segmentação por valor constante, que é utilizada pelos trabalhos (MALHOTRA *et al.*, 2015; MALHOTRA *et al.*, 2016), na qual se escolhe arbitrariamente um valor de segmentação para seccionar a série a partir de uma análise manual dos dados. A segunda técnica, apresentada na Seção 5.3.2, é baseada em picos, na qual o tamanho da janela é definido a partir da distância mediana entre máximos locais da série, separados por uma certa distância. A terceira técnica, que é proposta por este trabalho e apresentada na Seção 5.3.3, utiliza um *threshold* de varredura, que secciona a série em todos os pontos de travessia do *threshold* pela série e tenta maximizar a similaridade entre as janelas obtidas.

5.3.1 Segmentação Baseada em Tamanho de Janela Constante

Na segmentação de janelas baseada em tamanho constante, um valor h é escolhido com base na análise manual da periodicidade da série e é utilizado para seccionar a série em janelas de tamanho h não sobrepostas. A ideia é escolher h tal que as janelas sejam o mais semelhantes possível.

Essa técnica é fácil de entender e implementar, mas apresenta algumas desvantagens, como a necessidade de escolher manualmente um valor para a segmentação, o que depende de um conhecimento prévio sobre o conjunto de dados, além de um certo nível de especialização para identificar os ciclos presentes nas séries. Outra desvantagem mais importante é que diferentes séries do mesmo conjunto de dados, ou até mesmo trechos diferentes na mesma série, podem apresentar durações de ciclos com variações que comprometem a similaridade das janelas, mesmo com uma escolha ótima de tamanho constante das janelas.

Existem algumas possibilidades para se enfrentar o problema da variabilidade da duração dos ciclos entre séries diferentes, como, por exemplo, escolher um tamanho de janela específico para cada série do conjunto de dados e, ao final, realizar uma reamostragem das séries para ajustá-las todas ao mesmo tamanho de janela. Contudo, o problema da variabilidade dentro da própria série ainda persiste, e a necessidade de analisar manualmente a duração dos ciclos de cada uma das séries no conjunto de dados torna o problema ainda mais complexo.

5.3.2 Segmentação Baseada em Picos

A técnica apresentada nesta seção visa utilizar a estrutura da própria série para definir um tamanho de janela ótimo, na qual todos os máximos locais da série são encontrados e, dado

um tamanho de vizinhança escolhido arbitrariamente, excluem-se os máximos locais que habitam a mesma vizinhança de outro máximo local com base no tamanho escolhido. Ou seja, dada uma série temporal X e uma distância $h \in \mathbb{N}^+$, esta técnica encontra todos os máximos locais de X de modo que a distância mínima entre qualquer par de máximos locais seja h.

Dada uma série temporal $X = [x_1, ..., x_n]$, dizemos que x_i é um máximo local se $x_i > x_{i\pm1}$, ou se x_i é o ponto central em um platô $P = [x_{i-p}, ..., x_i, ..., x_{i+p}]$ onde $\forall x \in X, x = x_i$, $x_i > x_{i-p-1}$ e $x_i > x_{i+p+1}$. Seja $\vec{M} = [m_1, ..., m_k]$ onde $\forall (1 \leq i \leq k), m_i \leq m_{i+1}$, a lista ordenada de máximos locais de X. Para cada $i \in \{1, ..., k\}$, se existe algum outro ponto de máximo local na vizinhança de distância h de m_i, remove-se m_i. Ao final do algoritmo, teremos todos os máximos locais de X com pelo menos distância h em relação a qualquer outro máximo.

Esta técnica possui vantagens em relação à técnica de segmentação por tamanho constante, pois não requer o conhecimento de um tamanho exato que capture o ciclo das séries, além de permitir uma melhor segmentação de séries quasi-periódicas ao capturar os pequenos desvios que podem ocorrer nos ciclos pela utilização dos picos da série. Apesar disso, ainda é necessária a definição de um hiperparâmetro h, já que um h pequeno pode levar à separação em janelas diferentes de múltiplos picos que pertencem a um mesmo ciclo, e um h muito grande pode agregar vários ciclos dentro de uma única janela.

5.3.3 Segmentação por Similaridade de Janelas através de Varredura de Threshold (WSST)

Como podemos observar nas técnicas anteriores, elas requerem ou algum conhecimento prévio do conjunto de dados, uma análise individual de cada série nos dados, ou são extremamente sensíveis a pequenas variações nos ciclos da série.

Para enfrentar esses problemas, é proposta uma técnica de segmentação dinâmica de janelas, o Window Similarity Scanning Threshold (WSST), que realiza uma varredura de um *threshold* no eixo y da série, com esse *threshold* variando do ponto mínimo da série até o ponto máximo, e a segmentação em janelas nos pontos em que a série atravessa o valor do *threshold*. Com isso, para cada *threshold* é calculada a dissimilaridade média entre todas as janelas criadas, escolhendo-se assim o *threshold* que minimiza essa dissimilaridade.

A importância da escolha do *threshold* é exemplificada nas figuras 20, 21 e 22. Na Figura 20, o *threshold* escolhido divide a série em janelas completamente caóticas, sem um padrão identificável. Na Figura 21, um *threshold* melhor é escolhido, já existe um padrão nas janelas obtidas, mas é possível identificar que o ciclo presente na série é segmentado de forma

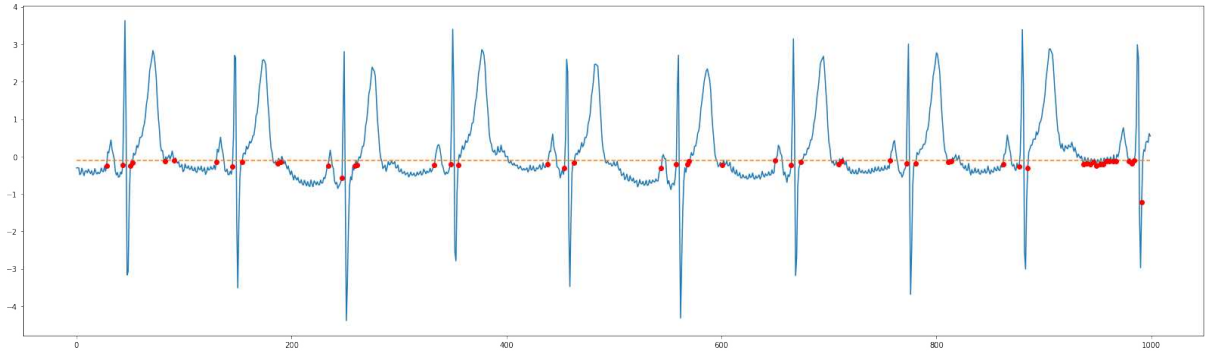

Figura 20 – Uma escolha ruim de *threshold*

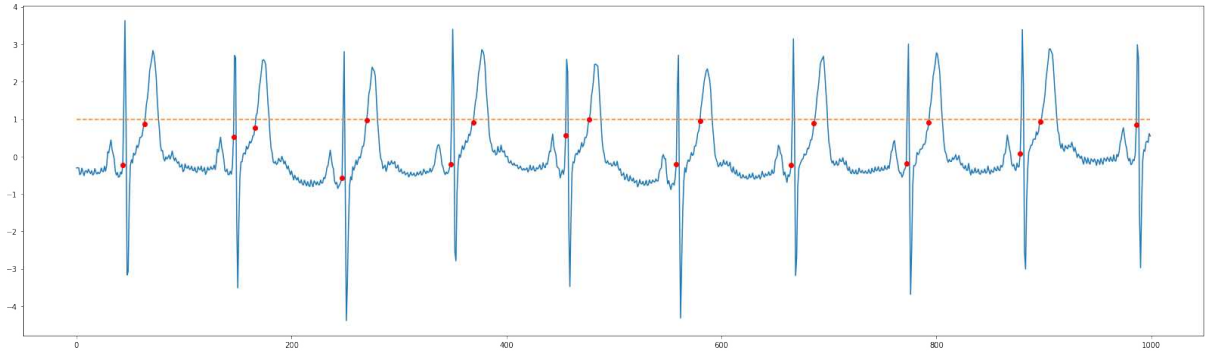

Figura 21 – Uma escolha de threshold aceitável, mas que pode ser melhorada

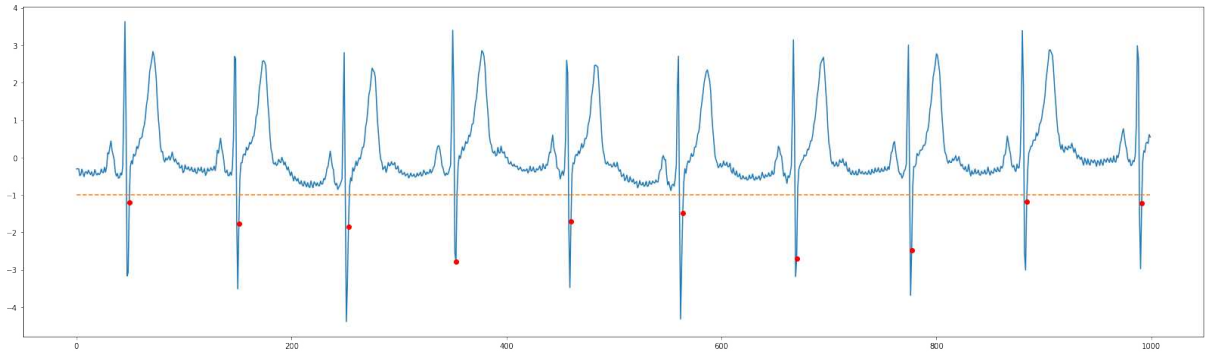

Figura 22 – Uma boa escolha de threshold que divide a série em cada um de seus ciclos

que cada ciclo acaba sendo seccionado em duas janelas separadas. Por fim, para reforçar a importância da escolha de um bom *threshold*, na Figura 22 é possível visualizar a segmentação bem definida dos ciclos, onde cada ciclo será representado em uma janela correspondente.

Seja $X = [x_1, ..., x_n]$ uma série temporal de tamanho n, e $\mathrm{T} = [\tau_1, ..., \tau_l]$, onde $\tau_1 = \min(X)$, $\tau_l = \max(X)$ e $\forall(1 < i < l), \tau_{i+1} - \tau_i = \tau_i - \tau_{i-1}$, uma lista de thresholds ordenada, para cada $\tau \in \mathrm{T}$ é calculada a lista de pontos

$$P = \left\{ p \in \mathbb{N} \;\middle|\; \begin{array}{l} 1 \leq p \leq n \\ x_p \geq \tau \\ x_{p-1} < \tau \end{array} \right\} \tag{5.1}$$

onde P é a lista de pontos em X que atravessam τ de baixo para cima, ou seja, x_i é um ponto escolhido se $x_{i-1} < \tau$ e $x_i \geq \tau$. Com isso, calcula-se $d = median\left\{ p_{i+1} - p_i \mid 1 \leq i < \|P\| \right\}$ como a distância mediana entre os pontos de P e $[J_1, \ldots, J_m]$ a lista de janelas de X tal que $J_i = [x_{p_i}, .., x_{p_i+d}]$. Assim, é calculada a janela mediana J_{med}, onde se $J_i = [J_i^{(1)}, \ldots, J_i^{(d)}], \forall (1 \leq i \leq m)$, a janela mediana é definida como

$$J_{med}^{(j)} = median\left\{ J_i^{(j)} \mid 1 \leq i \leq m \right\} \tag{5.2}$$

$$J_{med} = [J_{med}^{(1)}, \ldots, J_{med}^{(d)}] \tag{5.3}$$

A partir da janela mediana J_{med}, é calculada a matriz de erros $\vec{E} = [e_1, ..., e_m]$, onde e_i é o erro quadrático médio entre a janela J_i e J_{med}, definido por:

$$e_i = \frac{(J_i^{(1)} - J_{med}^{(1)})^2 + \ldots + (J_i^{(d)} - J_{med}^{(d)})^2}{d} \tag{5.4}$$

Por fim, é calculado o erro geral $\varepsilon(\tau) = mean(\vec{E})$, que determina o erro dado pelo threshold τ escolhido. O objetivo da técnica é encontrar

$$\min_{\tau \in \mathrm{T}} \varepsilon(\tau) \tag{5.5}$$

e utilizar as janelas geradas a partir do valor ótimo de τ para treinar e testar os modelos preditivos na detecção de anomalias.

Assim, por utilizar uma varredura de thresholds, essa técnica evita os processos manuais na obtenção de um tamanho de janela, diferentemente das técnicas apresentadas anteriormente. A abordagem consegue adaptar-se à periodicidade das séries temporais ao utilizar tamanhos de janelas variáveis para cada ciclo de cada série, além de ser capaz de segmentar séries em seus ciclos característicos sem a necessidade de um conhecimento prévio sobre o conjunto de dados.

5.4 Experimentação e Resultados

Para os experimentos, utilizou-se o conjunto de dados de eletrocardiogramas do MIT apresentado na Seção 4.4.1.2. O objetivo foi comparar os escores obtidos pelos modelos de *LSTM Empilhada* (MALHOTRA *et al.*, 2015), *Encoder-Decoder* (MALHOTRA *et al.*, 2016) e o modelo proposto na Seção 5.2. Como etapa de pré-processamento das séries temporais, foram aplicadas as técnicas de segmentação de janela descritas na Seção 5.3.

Antes da aplicação das técnicas de segmentação de janelas, as séries temporais foram divididas nos conjuntos de treino, validação e teste. Esta divisão prévia garante que, após a segmentação e o embaralhamento das janelas para o treinamento dos modelos, não haja vazamento de informações entre os conjuntos, o que poderia ocasionar viés durante a fase de teste. Tal abordagem previne que os modelos tenham acesso a fragmentos de todas as séries durante o treinamento.

De modo similar à experimentação do capítulo anterior, a Tabela 3 apresenta a comparação dos valores de *precision*, *recall* e das métricas F_1 e $F_{0.1}$, com a inclusão da métrica F_2 nos resultados. Esta última atribui maior ênfase à redução de falsos negativos, podendo constituir outro parâmetro relevante para a comparação dos modelos.

Tabela 3 – Resultados dos testes para os dados de eletrocardiogramas do MIT

W. BREAKING	MODELS	Precision	Recall	F_2	$F_{0.1}$	F_1
Constant	LSTM	6.22%	25.18%	0.1565	0.0627	0.0998
Constant	Enc-Dec	9.28%	22.07%	0.1730	0.0933	0.1306
Constant	CRED	8.58%	22.15%	0.1682	0.0863	0.1237
Peaks	LSTM	6.01%	25.26%	0.1540	0.0606	0.0971
Peaks	Enc-Dec	8.78%	17.15%	0.1441	0.0883	0.1162
Peaks	CRED	9.34%	26.64%	0.1944	0.0941	0.1384
WSST	LSTM	5.53%	20.31%	0.1323	0.0557	0.0869
WSST	Enc-Dec	10.54%	30.89%	0.2229	0.1061	0.1572
WSST	CRED	10.55%	36.42%	0.2444	0.1062	0.1636

A partir dos resultados apresentados na Tabela 3, observa-se que a técnica proposta WSST alcança desempenho superior em todos os aspectos avaliados nos modelos baseados em *encoder-decoder*. Esta técnica proporciona melhorias de até 35% quando comparada às diferentes técnicas de segmentação de janelas aplicadas ao modelo *Encoder-Decoder*, e melhorias de até 32% quando aplicada ao modelo **CRED**.

Além disso, os resultados demonstram que, na maioria dos casos, o modelo preditivo proposto, WSST, apresenta desempenho superior a ambos os modelos do estado da arte, com melhorias de até 19% em relação ao modelo *Encoder-Decoder* e de até 88% quando comparado ao modelo de LSTM Empilhada.

Por fim, destacam-se os resultados obtidos quando combinada a técnica de segmentação de janelas proposta com o modelo preditivo de detecção de anomalias proposto. Esta combinação alcança desempenho superior em todas as métricas de avaliação, com destaque para o *recall* que apresenta uma melhoria de 18% quando comparado ao segundo melhor *recall* obtido

nos experimentos.

5.5 Conclusão

Neste capítulo, foi apresentado um novo modelo preditivo para detecção de anomalias em séries temporais, um *auto-encoder* baseado em LSTM que traz uma variação sobre o modelo anteriormente proposto por (MALHOTRA *et al.*, 2016). O modelo desenvolvido obteve resultados expressivos nos testes, mostrando-se superior aos modelos do estado da arte na maior parte dos cenários avaliados. O desempenho superior em comparação ao modelo de (MALHOTRA *et al.*, 2016) pode ser explicado pelo fato de o modelo proposto manter a representação gerada pelo *encoder* como entrada durante todo o processo de reconstrução da série pelo *decoder*. Esta abordagem previne o problema de "esquecimento"característico das LSTMs no processo de reconstrução, fenômeno que pode ocorrer com o modelo de (MALHOTRA *et al.*, 2016), especialmente na reconstrução de janelas extensas, uma vez que, quanto maior a série de entrada, maior o impacto do "esquecimento"na reconstrução ponto a ponto.

Também foram apresentadas técnicas de segmentação de janelas utilizadas no pré-processamento das séries temporais, as quais possuem menção escassa na literatura, onde predominam processos manuais, frequentemente com auxílio de especialistas. Esse capítulo propõe uma técnica que utiliza a própria estrutura intrínseca da série para determinar os pontos ótimos de segmentação, maximizando a similaridade entre as janelas resultantes. Esta abordagem não apenas reduz a necessidade de intervenção manual sobre os dados, como também aprimora a qualidade das janelas fornecidas como entrada para os modelos preditivos. Tal melhoria é evidenciada pelos resultados apresentados na Seção 5.4, nos quais o **WSST** demonstra desempenho superior em comparação a outras técnicas, especialmente quando aplicado aos modelos de *Auto-Encoder* que são mais sensíveis à qualidade das janelas de entrada.

Por fim, quando combinadas as técnicas propostas nesse capítulo, obtém-se uma abordagem robusta para a análise de anomalias em séries temporais, apresentando desempenho superior a qualquer outra combinação de técnicas avaliadas neste estudo. Tal contribuição abre perspectivas promissoras para aprimoramentos e aplicações em trabalhos futuros.

6 CONCLUSÃO

Nesse trabalho, foram apresentadas técnicas de análise de anomalias em séries temporais, com revisão de abordagens existentes na literatura. A partir desse estudo, foram propostas novas arquiteturas e técnicas de pré-processamento de séries temporais capazes de aprimorar o desempenho tanto de modelos da literatura quanto dos modelos desenvolvidos nesta pesquisa.

Foi proposta uma técnica de ensemble de modelos, denominada **TSPME-AD**, que obtém resultados consistentemente superiores quando comparada aos seus modelos base, mesmo quando alguns dos modelos componentes do ensemble apresentam desempenho inferior aos demais. Adicionalmente, foram apresentadas e analisadas diversas funções de combinação, sendo que a função implementada no **TSPME-AD** demonstrou desempenho superior de forma consistente nos testes realizados.

Também foi proposto um novo modelo preditivo de detecção de anomalias em séries temporais, denominado **CRED**, que utiliza como base o modelo proposto por (MALHOTRA *et al.*, 2016). A arquitetura foi modificada com o objetivo de solucionar o problema do esquecimento característico das LSTMs. Conforme demonstrado nos resultados experimentais, o modelo proposto obtém consistentemente desempenho equivalente ou superior aos demais modelos analisados neste estudo.

Por fim, este trabalho abordou o problema da segmentação de janelas como etapa de pré-processamento das séries temporais, fundamental para viabilizar o treinamento e teste dos modelos preditivos estudados. Foram comparadas diferentes técnicas de segmentação, sendo as mais básicas aquelas utilizadas em trabalhos correlatos, que demandam maior intervenção manual e podem ser prejudicadas pela variabilidade da periodicidade das séries. A técnica de segmentação dinâmica proposta, denominada **WSST**, reduz significativamente a necessidade de processos manuais no pré-processamento, além de gerar janelas alinhadas com a periodicidade intrínseca de cada série. Esta abordagem facilita a identificação de padrões pelos modelos preditivos, resultando em desempenho superior na maioria dos cenários avaliados, conforme evidenciado pelos experimentos realizados.

6.1 Trabalhos Futuros

A área de detecção de anomalias em séries temporais é bastante ampla e, apesar das várias análises apresentadas nesse trabalho, a possibilidade de novas investigações é extremamente vasta. Essas possibilidades incluem: novas arquiteturas de modelos de detecção, técnicas aprimoradas de pré-processamento de séries, diferentes abordagens para combinação de modelos e a obtenção de conjuntos de dados mais extensos para melhor avaliação da qualidade das técnicas, entre muitas outras análises possíveis.

Com base nisso, uma possível direção futura é a avaliação dos modelos aqui propostos utilizando novos conjuntos de dados reais, como o do MIT-BIH (MOODY; MARK, 2001), com o objetivo de analisar seus comportamentos em diferentes cenários. Esses cenários incluem séries com ciclos de variabilidade diversa, ciclos de diferentes níveis de complexidade, entre outras características.

Outro ponto para análise futura em relação ao modelo de ensemble proposto é a possibilidade de utilizar outros modelos base para sua composição, bem como implementar novas funções de atenuação e combinação dos escores. Um exemplo seria a aplicação de redes neurais como função de combinação, abordagem já apresentada em outros contextos de ensemble na literatura.

Da mesma forma que este trabalho propôs um novo modelo preditivo de detecção de anomalias, estudos futuros podem propor novos modelos como variações dos existentes ou até arquiteturas completamente novas. A exploração do uso de modelos preditivos para detecção de anomalias em séries temporais tem ganhado relevância mais recentemente, o que abre diversas possibilidades para o desenvolvimento de novos modelos capazes de obter bons resultados nos conjuntos de dados consolidados da literatura.

Por fim, a análise de técnicas de segmentação de janelas em séries temporais é uma área pouco explorada na literatura, mas que, como mostrado nos resultados obtidos neste trabalho, pode apresentar impactos significativos no desempenho dos modelos preditivos. Com isso, como trabalhos futuros, podem ser propostas novas técnicas de segmentação de janelas mais robustas e menos dependentes de interferência manual, potencialmente afetando de forma positiva o desempenho de outros modelos já apresentados anteriormente na literatura.

REFERÊNCIAS

AGGARWAL, C. C. Outlier ensembles: position paper. **ACM SIGKDD Explorations Newsletter**, ACM, v. 14, n. 2, p. 49–58, 2013.

CHANDOLA, V.; BANERJEE, A.; KUMAR, V. Anomaly detection: A survey. **ACM computing surveys (CSUR)**, ACM, v. 41, n. 3, p. 15, 2009.

CHEBOLI, D. Anomaly detection of time series: A thesis submitted to the faculty of the graduate school of the university of minnesota. 2010.

GAO, J.; TAN, P.-N. Converting output scores from outlier detection algorithms into probability estimates. In: IEEE. **Sixth International Conference on Data Mining (ICDM'06)**. [S.l.], 2006. p. 212–221.

GAO, Y.; YANG, T.; XU, M.; XING, N. An unsupervised anomaly detection approach for spacecraft based on normal behavior clustering. In: IEEE. **2012 Fifth International Conference on Intelligent Computation Technology and Automation**. [S.l.], 2012. p. 478–481.

HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. **Neural computation**, MIT Press, v. 9, n. 8, p. 1735–1780, 1997.

IVERSON, D. L. Inductive system health monitoring. 2004.

KENNEDY, J.; EBERHART, R. Particle swarm optimization. In: IEEE. **Proceedings of ICNN'95-international conference on neural networks**. [S.l.], 1995. v. 4, p. 1942–1948.

KEOGH, E.; LIN, J.; LEE, S.-H.; HERLE, H. V. Finding the most unusual time series subsequence: algorithms and applications. **Knowledge and Information Systems**, Springer, v. 11, n. 1, p. 1–27, 2007.

KIEU, T.; YANG, B.; JENSEN, C. S. Outlier detection for multidimensional time series using deep neural networks. In: IEEE. **2018 19th IEEE International Conference on Mobile Data Management (MDM)**. [S.l.], 2018. p. 125–134.

KITTLER, J.; HATER, M.; DUIN, R. P. Combining classifiers. In: IEEE. **Proceedings of 13th international conference on pattern recognition**. [S.l.], 1996. v. 2, p. 897–901.

KONG, X.; SONG, X.; XIA, F.; GUO, H.; WANG, J.; TOLBA, A. Lotad: Long-term traffic anomaly detection based on crowdsourced bus trajectory data. **World Wide Web**, Springer, v. 21, n. 3, p. 825–847, 2018.

LIU, F. T.; TING, K. M.; ZHOU, Z.-H. Isolation-based anomaly detection. **ACM Transactions on Knowledge Discovery from Data (TKDD)**, Acm, v. 6, n. 1, p. 3, 2012.

MAESSCHALCK, R. D.; JOUAN-RIMBAUD, D.; MASSART, D. L. The mahalanobis distance. **Chemometrics and intelligent laboratory systems**, Elsevier, v. 50, n. 1, p. 1–18, 2000.

MALHOTRA, P.; RAMAKRISHNAN, A.; ANAND, G.; VIG, L.; AGARWAL, P.; SHROFF, G. Lstm-based encoder-decoder for multi-sensor anomaly detection. **arXiv preprint arXiv:1607.00148**, 2016.

MALHOTRA, P.; VIG, L.; SHROFF, G.; AGARWAL, P. Long short term memory networks for anomaly detection in time series. In: PRESSES UNIVERSITAIRES DE LOUVAIN. **Proceedings**. [S.l.], 2015. p. 89.

MENG, F.; YUAN, G.; LV, S.; WANG, Z.; XIA, S. An overview on trajectory outlier detection. **Artificial Intelligence Review**, Feb 2018. ISSN 1573-7462. Disponível em: <https://doi.org/10.1007/s10462-018-9619-1>.

MOODY, G. B.; MARK, R. G. The impact of the mit-bih arrhythmia database. **IEEE Engineering in Medicine and Biology Magazine**, IEEE, v. 20, n. 3, p. 45–50, 2001.

OPITZ, D.; MACLIN, R. Popular ensemble methods: An empirical study. **Journal of artificial intelligence research**, v. 11, p. 169–198, 1999.

SEIFFERT, U.; MICHAELIS, B. Directed random search for multiple layer perceptron training. In: IEEE. **Neural Networks for Signal Processing XI: Proceedings of the 2001 IEEE Signal Processing Society Workshop (IEEE Cat. No. 01TH8584)**. [S.l.], 2001. p. 193–202.

SUTSKEVER, I.; VINYALS, O.; LE, Q. V. Sequence to sequence learning with neural networks. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2014. p. 3104–3112.

TARIQ, S.; LEE, S.; SHIN, Y.; LEE, M. S.; JUNG, O.; CHUNG, D.; WOO, S. S. Detecting anomalies in space using multivariate convolutional lstm with mixtures of probabilistic pca. In: ACM. **Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining**. [S.l.], 2019. p. 2123–2133.

WANG, X.; LIN, J.; PATEL, N.; BRAUN, M. Exact variable-length anomaly detection algorithm for univariate and multivariate time series. **Data Mining and Knowledge Discovery**, Springer, v. 32, n. 6, p. 1806–1844, 2018.