Relatório Final de Iniciação Científica

Programa Unificado de Bolsas

Universidade de São Paulo

mHealth: da coleta de dados à regressão de parâmetros clínicos por séries temporais e imagens obtidas por smartphone

Orientador: Diego Furtado Silva

Orientando: Isabela Guarnier De Mitri

Agosto/2024

Resumo

Com o crescente acesso a tecnologias móveis, a coleta e armazenamento de dados tem se tornado uma tarefa cada vez mais explorada pelas comunidades de diferentes domínios. Na área da saúde, esse progresso têm viabilizado a aquisição de dados variados, com aplicações em diversas especialidades. Entretanto, a utilização de dispositivos convencionais para coletar dados pode ficar limitada a ambientes hospitalares. Nesse contexto, surgem as ferramentas de *mobile health* (mHealth) que aproveitam das tecnologias móveis para aprimorar as interações entre as populações e os sistemas de saúde. Visando ampliar as possibilidades de utilização da mHealth e aproximá-las ainda mais da prática clínica, este projeto utiliza a aplicação de técnicas de Aprendizado de Máquina para manipular dados provenientes de dispositivos móveis.

1 Introdução

Com o avanço das tecnologias, houve o surgimento de algoritmos eficientes prontos para lidar com grandes conjuntos de dados. Essa capacidade tem viabilizado a coleta contínua de informações ao longo do tempo em diversos setores, tais como saúde, economia e monitoramento ambiental. Junto a isso, sensores móveis se tornaram cada vez mais acessíveis e precisos em termos de capacidade de processamento. Essa tendência tem resultado no surgimento de aplicações que coletam extensos volumes de dados temporais utilizando sensores [1].

Uma das aplicações que tem se destacado na área da Saúde é a análise de dados temporais em diversos contextos. Sinais relacionados à atividade cardiorrespiratória, por exemplo, podem ser utilizados na prevenção de ataques cardíacos [2], enquanto sinais provenientes de dispositivos de medição inercial, como acelerômetros, podem contribuir para a detecção de condições de saúde, como o mal de Parkinson [3].

Dados temporais podem ser extraídos por meio de alguns equipamentos, como oxímetro e eletrocardiógrafos. Contudo, sua utilização pode ser limitada ao ambiente hospitalar, e nesse contexto, vemos a importância de ferramentas de *mobile health* (mHealth). Em um relatório de 2018 [4], a OMS enfatiza que tecnologias móveis podem revolucionar as interações entre a população e os sistemas de saúde, tendo em vista o alcance e ampla aceitação dessa tecnologia.

No entanto, há vários obstáculos que precisam ser superados para viabilizar essa revolução. A padronização de tecnologias de mHealth e o acesso à informação ainda são desafios pertinentes.

Nesse cenário, o propósito desta pesquisa consiste em expandir as possibilidades de aplicação da mHealth através do emprego de Aprendizado de Máquina. Nesse contexto, experimentos preliminares mostraram que o algoritmo ROCKET [5] (do inglês *RandOm Convolutional KErnel Transform*), que utiliza convoluções aleatórias para extrair características de séries temporais, tem obtido bons resultados. No entanto, há um fenômeno relacionado a esse algoritmo que pode ser observado em diversos domínios de aplicação que envolvem classificação e regressão extrínseca de séries temporais: o caráter aleatório do ROCKET pode levar a resultados muito diferentes a cada execução. Soma-se a isso a falta de estudos sobre os efeitos da aleatoriedade neste algoritmo. Assim, o foco principal desta pesquisa é estudar o algoritmo ROCKET, com ênfase em propor técnicas para diminuir a instabilidade do algoritmo para dados do domínio da Saúde.

2 Realizações do período

As próximas seções descrevem as atividades realizadas ao longo do período. A Seção 3 apresenta uma introdução ao algoritmo ROCKET, enquanto as seguintes detalham as abordagens adotadas para otimizar o desempenho desse algoritmo. Entre as estratégias abordadas estão a modificação da distribuição probabilística dos kernels, análise da relação entre número de kernels e acurácia, customização de kernels e *bias*, e a aplicação do UMAP (Uniform Manifold Approximation and Projection) nas features extraídas e nos kernels. Finalmente, a Seção 7 discute os resultados obtidos, os desafios enfrentados e as reflexões sobre o desenvolvimento da pesquisa.

3 O que é o ROCKET?

RandOm Convolutional KErnel Transform, sigla em inglês para Transformação Convolucional Aleatória do Kernel, é um algoritmo que tem se destacado para a classificação de séries temporais, além de possuir algumas evidências de bons resultados na regressão extrínseca. Utilizando uma abordagem única, o ROCKET transforma séries temporais por meio de uma grande quantidade de kernels convolucionais aleatórios [5]

a partir de uma função de distribuição normal, gerando um mapa de recursos que é, posteriormente, alimentado a um classificador ou regressor linear.

O processo de transformação das séries temporais pelo ROCKET envolve a convolução de cada série com 10.000 kernels convolucionais aleatórios. Estes kernels possuem características como comprimento, peso, viés, dilatação e preenchimento, todas configuradas de forma aleatória. A aplicação subsequente do *pool* máximo global (max) e do *pool* de "proporção de valores positivos" (ppv) gera dois atributos por kernel, resultando em um total de 20.000 características por padrão [6].

Embora o ROCKET tenha demonstrado excelentes resultados em tarefas de classificação de séries temporais, há estudos que apontam a possibilidade de melhorias significativas, especialmente na combinação de múltiplos ROCKETs [7]. A natureza estocástica do algoritmo pode levar a uma grande variância nos resultados, motivando a investigação sobre como aprimorar seu desempenho, em especial buscando um desempenho mais estável.

Dessa forma, em vez de apenas avaliar o resultado final para uma dada tarefa, serão explorados os comportamentos das características extraídas, investigando a dependência entre essas características e suas correlações com o atributo-alvo.

4 Distribuição de Probabilidade

Um dos métodos explorados para tentar melhorar o desempenho do algoritmo foi a alteração da distribuição de probabilidade utilizada na geração dos kernels.

Originalmente, o ROCKET emprega uma distribuição normal padrão para gerar os valores dos kernels. Decidimos investigar o impacto de utilizar diferentes distribuições, como a T-Student, com o objetivo de avaliar se essas variações poderiam capturar melhor as características dos dados, proporcionando uma variabilidade melhor na geração dos kernels e, consequentemente, melhorar a acurácia do modelo. Ao testar essas diferentes distribuições, analisamos cuidadosamente os resultados obtidos para identificar possíveis ganhos em desempenho.

O experimento foi realizado utilizando 16 datasets[1], sendo alguns escolhidos manualmente, enquanto outros foram escolhidos randômicamente. O processo do experimento envolveu as seguintes etapas principais:

- **Divisão dos Dados:** Cada dataset foi dividido em conjuntos de treinamento e teste. A divisão foi feita utilizando o método *load classification*, com a opção

[1] `https://www.timeseriesclassification.com/dataset.php`

split configurada para "train"e "test", assegurando que o modelo fosse treinado em uma parte dos dados e avaliado em outra.

- **Transformação dos Dados:** O algoritmo ROCKET foi utilizado para transformar os dados brutos de séries temporais em representações adequadas para análise. Após o treinamento do ROCKET com o conjunto de treinamento, as transformações foram aplicadas tanto ao conjunto de treinamento quanto ao de teste.

- **Treinamento do Classificador:** Utilizamos o RandomForestClassifier para treinar o modelo com as features extraídas pelo ROCKET. Esse classificador foi escolhido por sua robustez e eficácia em problemas de classificação.

- **Avaliação do Modelo:** O desempenho do modelo foi avaliado no conjunto de teste, e as previsões foram comparadas com os rótulos reais para calcular a acurácia. Essa métrica fornece uma visão abrangente da eficácia do classificador.

- **Iterações e Resultados:** Foram realizadas 50 iterações para cada dataset, e os resultados foram agregados para obter uma visão consistente da performance do modelo. A acurácia foi calculada para cada iteração e os resultados finais foram obtidos como a média das 50 iterações.

De acordo com a tabela mostrada abaixo (figura 1), as melhorias na acurácia dos diferentes conjuntos de dados foram mínimas. Embora esperássemos que a aplicação das metodologias propostas resultasse em um aumento significativo na acurácia, os resultados obtidos não apresentaram mudanças substanciais em comparação com as abordagens tradicionais. Essa falta de progresso pode indicar que os dados utilizados já estavam suficientemente bem modelados, ou que as melhorias propostas não são tão eficazes para os tipos de dados ou padrões específicos presentes nos conjuntos analisados.

	NORMAL			TSTUDENT	
Dataset	Acurácia	Desvio-padrão	Dataset	Acurácia	Desvio-padrão
ArrowHead	0.7699428571	0.0219092064	ArrowHead	0.7672	0.024045216
DistalPhalanxTW	0.679856115	0.012439587	DistalPhalanxTW	0.6735251798	0.0139327319
FaceAll	0.7556449704	0.010280119	FaceAll	0.7790532544	0.025914187
FiftyWords	0.721098901098	0.0090263463	FiftyWords	0.72632967032	0.00871507739
GunPoint	0.97986666	0.0054693539	GunPoint	0.975466666666	0.00493770719
Ham	0.7540952380	0.029224160	Ham	0.762857142	0.026885813395

Figura 1: Tabela comparativa das distribuições Normal e T-Student

5 Análise da Relação entre Kernels, Bias e Acurácia

Nesta seção, abordamos a relação entre a quantidade de kernels utilizados pelo algoritmo ROCKET, a customização desses kernels, e o efeito do bias na acurácia do modelo. Os experimentos realizados investigaram como essas variáveis influenciam o desempenho do algoritmo em diferentes configurações e datasets.

5.1 Quantidade de Kernels e Acurácia

Investigamos a hipótese de que a quantidade de kernels gerados pelo algoritmo pode impactar a acurácia do modelo. Para testar essa hipótese, variamos o número de kernels num intervalo de 0 a 2000, analisando o impacto dessas variações na performance do modelo. Avaliamos se um aumento no número de kernels resultaria em melhorias significativas ou se haveria um ponto de saturação, além do qual a acurácia não melhoraria ou até mesmo diminuiria devido à inclusão de kernels redundantes.

Para cada configuração de quantidade de kernels, realizamos o treinamento e a avaliação do modelo ROCKET. Utilizamos as mesmas divisões de dados e procedimentos de transformação e classificação descritos anteriormente, garantindo a consistência nas condições de teste.

5.2 Efeito do bias

O bias, no contexto do ROCKET, é um valor adicionado às operações de convolução para ajustar a saída dos kernels antes da aplicação da função de ativação. Este experimento analisou como a introdução de um bias customizado poderia influenciar a capacidade do modelo de capturar padrões específicos nos dados.

5.3 Customização de Kernels

Exploramos a customização dos kernels como uma tentativa de aprimorar o desempenho do algoritmo. Ao contrário dos kernels gerados aleatoriamente pela distribuição normal padrão, os kernels customizados foram desenvolvidos para introduzir uma maior variedade e aleatoriedade nas análises. Esses kernels foram projetados para simular diferentes padrões e formas, permitindo capturar uma ampla gama de características dos datasets em análise. Isso inclui a criação de kernels que não seguem um padrão fixo, mas que variam de forma a explorar diferentes aspectos dos dados, potencialmente revelando informações que poderiam ser perdidas com kernels mais convencionais.

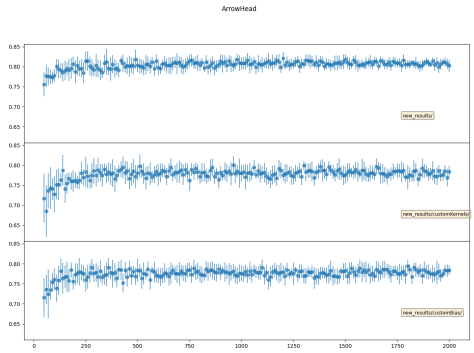

Figura 2: Arrow Head

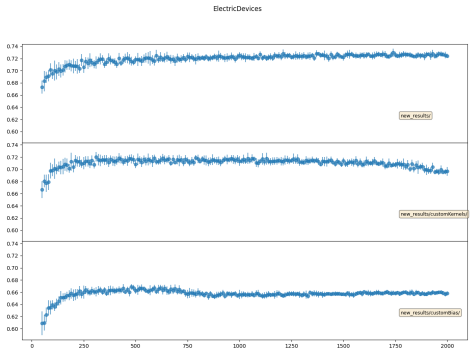

Figura 3: Distal Phalanx TW

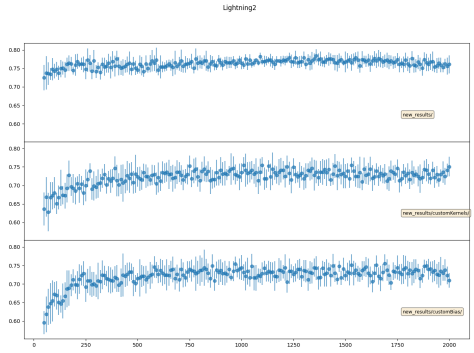

Figura 4: Lightning2

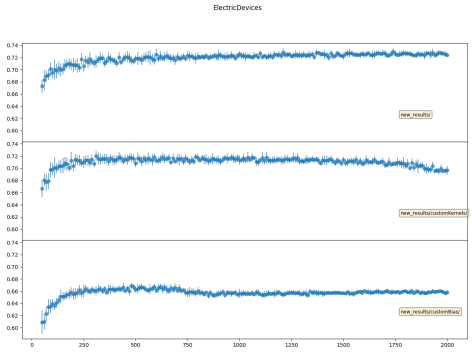

Figura 5: ElectricDevices

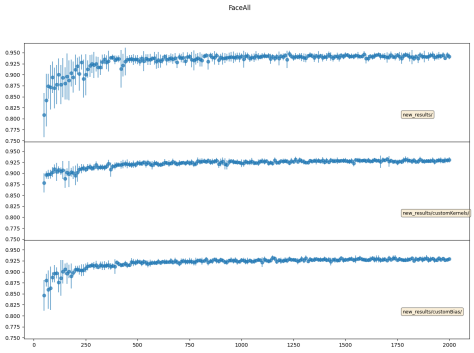

Figura 6: FaceAl

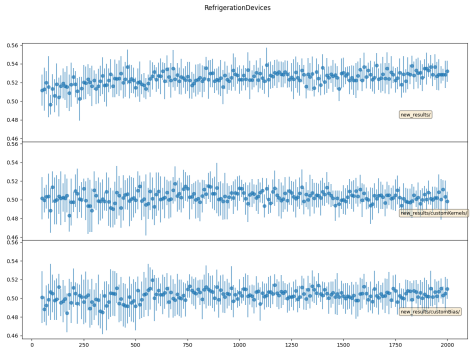

Figura 7: RefrigerationDevices

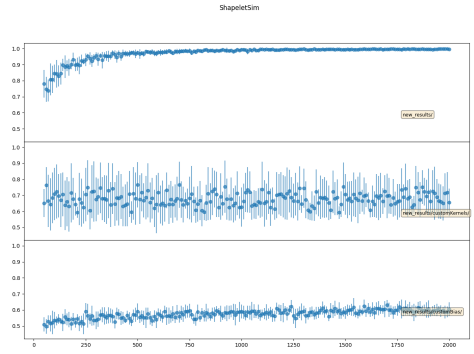

Figura 8: ShapeletSim

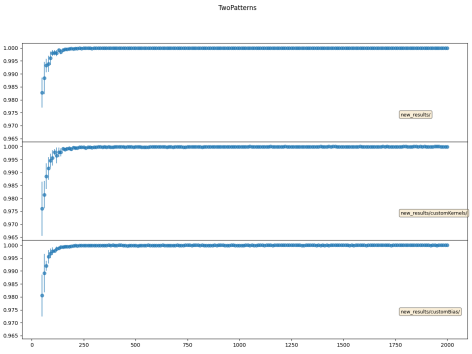

Figura 9: TwoPatterns

Para avaliar o impacto das diferentes customizações no desempenho do modelo, foram gerados gráficos (Figuras 2-9) que comparam três cenários distintos: configuração padrão (sem customização), kernels customizados e bias customizado. Em cada gráfico, a comparação é feita de cima para baixo, com o gráfico superior representando a configuração padrão, o gráfico do meio representando os kernels customizados, e o gráfico inferior mostrando o cenário de bias customizado, nos 8 datasets escolhidos.

No eixo horizontal, é representado o número de kernels utilizados, enquanto o eixo vertical mostra a acurácia do modelo. Essas visualizações permitem uma análise detalhada sobre como as customizações influenciam o desempenho do modelo. Através dessas comparações, é possível identificar quais abordagens proporcionam melhorias significativas e em quais contextos específicos as customizações se mostram mais eficazes.

Em alguns datasets, observamos que a acurácia do algoritmo se estabiliza após um certo número de kernels. Isso é evidente em datasets como ElectricDevices, FaceAll e TwoPatterns. No entanto, em outros casos, a adição de kernels customizados (incluindo o bias) resultou em uma redução significativa da acurácia, como observado nos datasets OSULeaf e Shapelet Slim. Além disso, enquanto alguns datasets mostram uma melhoria consistente com o aumento do número de kernels, outros exibem variações consideráveis na acurácia, como é o caso dos datasets RefrigerationDevices, Lightning2 e ArrowHead.

6 UMAP

O UMAP (Uniform Manifold Approximation and Projection) é uma técnica de redução de dimensionalidade amplamente utilizada em aprendizado de máquina e análise de dados. Desenvolvido para capturar e visualizar a estrutura de dados complexos, o UMAP é especialmente útil para explorar e interpretar dados de alta dimensionalidade, transformando-os em um espaço de menor dimensão que preserva as relações topológicas e geodésicas dos dados originais.

Uma das principais vantagens do UMAP [8] é sua capacidade de manter a estrutura global dos dados, facilitando a identificação de padrões e agrupamentos naturais. Diferente de outras técnicas de redução de dimensionalidade, como o PCA (Análise de Componentes Principais) ou o t-SNE (t-distributed Stochastic Neighbor Embedding), o UMAP não só preserva as relações locais, mas também as relações globais, proporcionando uma visualização mais intuitiva e informativa dos dados complexos.

Na presente análise, utilizamos o UMAP para visualizar as características (features) e os kernels gerados pelo algoritmo ROCKET, incluindo os kernels customizados (pontos azuis). O objetivo foi observar a distribuição desses elementos em um espaço reduzido, facilitando a interpretação das características e a identificação de padrões ou agrupamentos significativos.

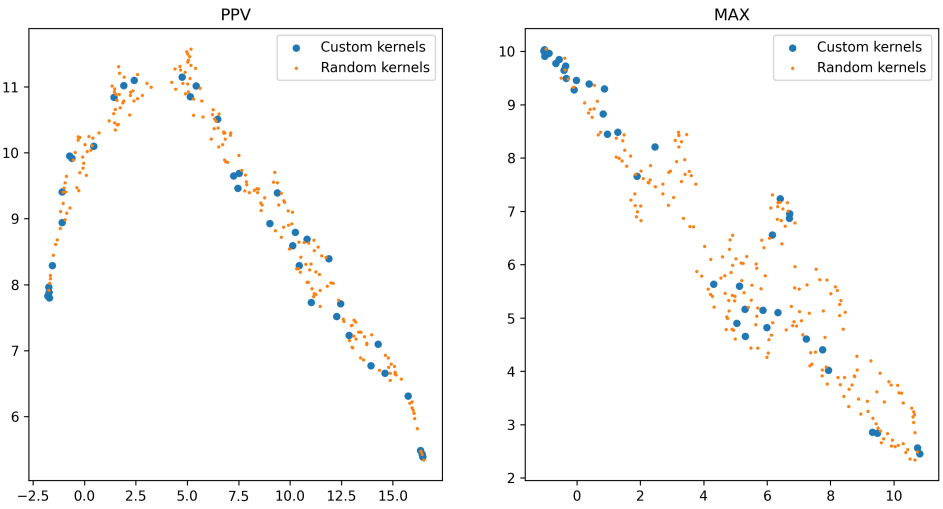

Figura 10: UMAP das features do dataset FaceAll

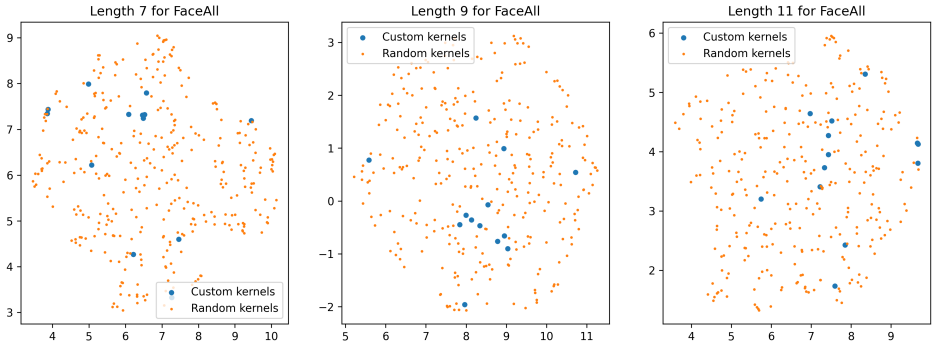

Figura 11: UMAP dos kernels de comprimento 7, 9 e 11 do dataset FaceAll

Para as features (figura 10), o UMAP nos permitiu visualizar como as diferentes características dos datasets se agrupam ou se distribuem no espaço reduzido, oferecendo uma compreensão sobre a semântica e a estrutura dos dados. Esta visualização é crucial para entender como as features interagem e se relacionam entre si.

Da mesma forma, ao aplicar o UMAP aos kernels (figura 11), incluímos tanto os kernels gerados aleatoriamente quanto os customizados. Esta abordagem nos possibilitou observar a distribuição dos kernels e avaliar se os kernels customizados, projetados para capturar padrões específicos dos dados, resultam em uma estrutura diferenciada em comparação com os kernels padrão.

7 Conclusão

Após um período extensivo de pesquisa e experimentação, este estudo proporcionou uma visão aprofundada sobre o algoritmo e suas aplicações para a classificação de séries temporais.

Os experimentos realizados mostraram que o ROCKET possui um desempenho promissor na extração de características a partir de séries temporais, com a capacidade de transformar dados brutos em representações úteis para modelos de classificação. No entanto, a natureza aleatória introduz uma variabilidade significativa nos resultados, o que pode comprometer a estabilidade e a confiabilidade do algoritmo em aplicações práticas.

Diversas estratégias foram exploradas para otimizar o desempenho do algoritmo. A alteração da distribuição de probabilidade utilizada na geração dos kernels foi uma abordagem importante; no entanto, os resultados mostraram que essa modificação não trouxe os benefícios esperados de forma geral. A alteração só demonstrou uma melhora na acurácia para datasets muito específicos, não apresentando uma vantagem significativa para a maioria dos casos. Isso indica que, embora a modificação da distribuição possa ser benéfica em contextos específicos, não é uma solução universal para a melhoria da acurácia do modelo.

A análise da relação entre o número de kernels e a acurácia, bem como a customização dos kernels, também foram abordadas com o objetivo de melhorar a eficácia do modelo. A customização dos kernels revelou que a redundância pode, paradoxalmente, diminuir a acurácia, evidenciando que a introdução de kernels semelhantes pode ser prejudicial ao invés de benéfica.

A aplicação do UMAP foi uma etapa importante na análise das características e kernels gerados pelo ROCKET. Essa técnica revelou como as características extraídas se agrupam e como os kernels se distribuem em um espaço reduzido, fornecendo uma visão clara sobre a estrutura dos dados e a potencial redundância entre kernels customizados. A visualização mostrou que kernels customizados estavam agrupados próximos uns

dos outros, sugerindo que a redundância pode ser um fator limitante na melhoria da acurácia do algoritmo.

Em conclusão, a pesquisa demonstrou que, embora o ROCKET tenha um grande potencial, há desafios significativos a serem superados para aprimorar sua estabilidade e performance, especialmente quando aplicado a dados do domínio da saúde. Futuras pesquisas podem se beneficiar da combinação dessas abordagens com métodos adicionais de validação e otimização para maximizar o desempenho do ROCKET em aplicações práticas.

Referências

[1] C.-C. M. Yeh, Y. Zhu, L. Ulanova, N. Begum, Y. Ding, A. Dau, Z. Zimmerman, D. F. Silva, A. Mueen, and E. Keogh, "Time series joins, motifs, discords and shapelets: a unifying view that exploits the matrix profile," *Data Mining and Knowledge Discovery*, vol. 32, no. 1, pp. 83–123, 2018.

[2] T. Alsuliman, D. Humaidan, and L. Sliman, "Machine learning and artificial intelligence in the service of medicine: Necessity or potentiality?," *Current research in translational medicine*, vol. 68, no. 4, pp. 245–251, 2020.

[3] I. E. Maachi, G.-A. Bilodeau, and W. Bouachir, "Deep 1d-convnet for accurate parkinson disease detection and severity prediction from gait," *Expert Systems with Applications,*, vol. 143, p. 113075, 2020.

[4] W. H. Organization, "mhealth. use of appropriate digital technologies for public health: report by director-general," *71st World Health Assembly provisional agenda item*, vol. 12, p. A71, 2018.

[5] A. Dempster, F. Petitjean, and G. I. Webb, "Rocket: exceptionally fast and accurate time series classification using random convolutional kernels," *Data Mining and Knowledge Discovery*, vol. 34, no. 5, pp. 1454–1495, 2020.

[6] M. Middlehurst, J. Large, M. Flynn, J. Lines, A. Bostrom, and A. Bagnall, "Hivecote 2.0: a new meta ensemble for time series classification," *Machine Learning*, vol. 110, no. 11, pp. 3211–3243, 2021.

[7] C. W. Tan, C. Bergmeir, F. Petitjean, and G. Webb, "Time series extrinsic regression," *Predicting numeric values from time series data*, vol. 35, p. 11, 2021.

[8] J.-Y. Franceschi, A. Dieuleveut, and M. Jaggi, "Unsupervised scalable representation learning for multivariate time series," *arXiv preprint arXiv:1802.03426*, vol. NA, p. NA, 2018.