REPRESENTAÇÃO SIMBÓLICA DE SÉRIES TEMPORAIS PARA RECONHECIMENTO DE ATIVIDADES HUMANAS NO SMARTPHONE

KEVIN GUSTAVO MONTERO QUISPE

REPRESENTAÇÃO SIMBÓLICA DE SÉRIES TEMPORAIS PARA RECONHECIMENTO DE ATIVIDADES HUMANAS NO SMARTPHONE

Dissertação apresentada ao Programa de Pós-Graduação em Informática do Instituto de Computação da Universidade Federal do Amazonas como requisito parcial para a obtenção do grau de Mestre em Informática.

Orientador: D.Sc. Eduardo James Pereira Souto

Manaus

Maio de 2018

Ficha Catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

Quispe, Kevin Gustavo Montero
Q8r Representação simbólica de séries temporais para reconhecimento de atividades humanas no smartphone / Kevin Gustavo Montero Quispe. 2018
102 f.: il. color; 31 cm.

Orientador: Eduardo James Pereira Souto
Dissertação (Mestrado em Informática) - Universidade Federal do Amazonas.

1. reconhecimento de atividades humanas. 2. representação simbólica. 3. sensores inerciais. 4. smartphone. I. Souto, Eduardo James Pereira II. Universidade Federal do Amazonas III. Título

FOLHA DE APROVAÇÃO

"Representação Simbólica de Séries Temporais para Reconhecimento de Atividades Humanas no Smartphone"

KEVIN GUSTAVO MONTERO QUISPE

Dissertação de Mestrado defendida e aprovada pela banca examinadora constituída pelos Professores:

Prof. Eduardo James Pereira Souto - PRESIDENTE

Prof. Marco Antonio Pinheiro de Cristo - MEMBRO INTERNO

Prof. Daniel Macêdo Batista - MEMBRO EXTERNO

Manaus, 14 de Agosto de 2018

Agradecimentos

Os meus agradecimentos a todos aqueles que, de alguma forma, permitiram que esta dissertação se concretizasse. Agradeço, primeiramente, as pessoas mais importantes da minha vida, minha família. Aos meus pais Juan Carlos e Maria Mirian pela educação, apoio e motivação recebida por eles. A minha irmã, Yasmin, pela forma amiga e generosa com que sempre me ajudou e incentivou durante todos os períodos de dificuldades. O meu muito obrigado.

Agradeço ao Prof. Dr. Eduardo Souto, do Instituto de Computação da UFAM, que me aceitou como orientando, acreditando em min e nas minhas capacidades. Não menos importantes, agradeço aos meus amigos e membros do grupo de pesquisa — *Emergency Tecnologies and Security Systems* (ETSS). Agradeço pelos momentos compartilhados, convívio e boas discussões que obtivemos. Em especial a dupla Wesllen Lima e Hendrio Luis, que participaram fortemente no desenvolvimento desta pesquisa com críticas, ideias e sugestões.

Por fim, gostaria de agradecer as instituições de fomento a pesquisa e educação que apoiaram de maneira econômica o desenvolvimento deste trabalho.

"Quando penso que já cheguei ao meu limite, descubro que tenho forças para ir além."
(Ayrton Senna)

Resumo

O reconhecimento de atividade humanas (RAH) por meio de sensores embutidos em dispositivos vestíveis como, por exemplo, smartphones tem permitido o desenvolvimento de soluções capazes de monitorar o comportamento humano. No entanto, tais soluções têm apresentado limitações em termos de eficiência no consumo dos recursos computacionais e na generalização para diferentes configurações de aplicação ou domínio de dados. Essas limitações são exploradas neste trabalho no processo de extração de características, na qual as soluções existentes utilizam uma abordagem manual para extrair as características dos dados de sensores. Para superar o problema, este trabalho apresenta uma abordagem automática de extração de características baseada na representação simbólica de séries temporais — representação definida por conjuntos de símbolos discretos (palavras). Nesse contexto, este trabalho apresenta uma extensão do método de representação simbólica Bag-Of-SFA-Symbols (BOSS) para lidar com o processamento de múltiplas séries temporais, reduzir a dimensionalidade dos dados e gerar modelos de classificação compactos e eficiêntes. O método proposto, denominado Multivariate Bag-Of-SFA-Symbols (MBOSS), é avaliado para a classificação de atividades físicas a partir de dados de sensores inerciais. Experimentos são conduzidos em três bases de dados públicas e para diferentes configurações experimentais. Além disso, avalia-se a eficiência do método em aspectos como tempo de computação e espaço de dados. Os resultados, em geral, demostram uma eficácia de classificação equivalente as soluções baseadas na abordagem comun de extração manual de características, destacando os resultados obtidos na base de dados com nove classes de atividades (UniMib SHAR), onde o MBOSS obteve uma acurácia de 99% e 87% para o modelo personalizado e generalizado, respectivamente. Os resultados de eficiência do MBOSS demostram o baixo custo computacional da solução e mostram a viabilidade de aplicação em smartphones.

Palavras-chave: *reconhecimento de atividades humanas, representação simbólica, sensores inerciais, smartphone.*

Abstract

Human activity recognition (RAH) through sensors embedded in wearable devices such as smartphones has allowed the development of solutions capable of monitoring human behavior. However, such solutions have presented limitations in terms of efficiency in the consumption of computational resources and generalization for different application or data domain configurations. These limitations are explored in this work in the feature extraction process, in which existing solutions use a manual approach to extract the characteristics of the sensor data. To overcome the problem, this work presents an automatic approach to feature extraction based on the symbolic representation of time series — representation defined by sets of discrete symbols (words). In this context, this work presents an extension of the symbolic representation of the Bag-Of-SFA-Symbols (BOSS) method to handle the processing of multiple time series, reduce data dimensionality and generate compact and efficient classification models. The proposed method, called Multivariate Bag-Of-SFA-Symbols (MBOSS), is evaluated for the classification of physical activities from data of inertial sensors. Experiments are conducted in three public databases and for different experimental configurations. In addition, the efficiency of the method is evaluated in aspects such as computing time and data space. The results, in general, show an efficiency of classification equivalent to the solutions based on the traditional approach of manual extraction, highlighting the results obtained in the database with nine classes of activities (UniMib SHAR), where MBOSS obtained an accuracy of 99% and 87% for the custom and generalized template, respectively. The efficiency results of MBOSS demonstrate the low computational cost of the solution and show the feasibility of application in smartphones.

Keywords: *human activity recognition, symbolic representation, inertial sensors, smartphone.*

Lista de Figuras

Lista de Tabelas

Sumário

Capítulo 1

Introdução

Nos últimos anos, a comunidade científica tem buscado aprofundar os conhecimentos sobre como os humanos agem, tomam decisões e interagem com outros indivíduos (Cook & Krishnan, 2015). Dentre as áreas de pequisa, o reconhecimento de atividades humanas (RAH) tem avançado no desenvolvimento de sistemas capazes de identificar padrões de comportamento humano a partir de dados de sensores. Esses padrões, relacionados a atividades ou hábitos, possibilitam que aplicações inovadoras surjam para atender as necessidades dos usuários, dar suporte a tomada de decisão e expandir os conhecimentos da área.

Em especial, a tarefa de reconhecer atividades tem se destacado em aplicações nas áreas da saúde, segurança e indústria. Na saúde, reconhecer atividades permite monitorar o desempenho funcional e as condições de saúde de indivíduos, detectar comportamentos anormais (e.g., quedas em idosos) e hábitos irregulares (e.g., sedentarismo e tabagismo) (Mubashir et al., 2013; Sazonov et al., 2011). Na segurança, mecanismos de autenticação contínua em *smartphones* podem ser desenvolvidos utilizando as informações das atividades reconhecidas (Ehatisham-ul Haq et al., 2017). E na indústria, as atividades dos trabalhadores podem ser monitoradas para extrair informações de produtividades e avaliar o progresso em, por exemplo, construções civis (Akhavian & Behzadan, 2016; Maekawa et al., 2016).

Motivado pelas oportunas aplicações, este trabalho investiga o desenvolvimento de sistemas RAH utilizando *smartphones*. Os *smartphones* são ferramentas cada vez mais oportunas, flexíveis e prontamente disponíveis para monitorar automaticamente as atividades do dia a dia dos usuários. Comercializados em massa, os *smartphones* facilitam que cada vez mais pessoas tenham acesso à tecnologia desenvolvida neste trabalho. Além disso, esses dispositivos portáteis podem monitorar continuamente os usuários, sem impor nenhum inconveniente.

1.1 Problema

Na literatura, o desenvolvimento de sistemas RAH ainda apresenta varias questões em aberto (Shoaib et al., 2015). Uma questão é o monitoramento invasivo e contínuo de atividades, na qual a utilização de dispositivos móveis tem apresentado vantagens em relação ao uso de vídeo câmeras. Existem, também, as dificuldades de execução em tempo real desses sistemas, limitações de processamento e bateria em dispositivos móveis, e como lidar com a extração de conhecimento a partir de dados de sensores com múltiplos sinais de informação. Nesse contexto, este trabalho explora a viabilidade de usar *smartphones* para realizar o reconhecimento automático de atividades, abordando também as limitações atuais dos sistemas RAH.

O desenvolvimento de sistemas RAH em *smartphones* introduz novos desafios relacionados à implementação de um sistema de reconhecimento de padrões em um dispositivo de uso diário. O primeiro trata-se de escolher quais dos sensores serão utilizados para obter informações das atividades. Para isto, este trabalho propõe reconhecer atividades físicas a partir dos dados coletados dos sensores inerciais, em particular, do acelerômetro. Esses sensores são pequenos, confiáveis, e de baixo consumo de energia. Além disso, devido à popularização da tecnologia MEMS (do Inglês, *Microelectromechanical Systems*), esses sensores são frequentemente encontrados nos *smartphones* comercializados nos dias atuais.

O segundo desafio trata-se da forma que o sistema processará as informações dos sensores e determinará a atividade física que se está praticando. Neste caso, a solução deve ser capaz de processar as informações atendendo as limitações de recursos dos dispositivos, como processamento e memória. Para isto, este trabalho propõe a implementação de algoritmos supervisionados de aprendizagem de máquina para a tarefa e de métodos de extração de conhecimentos eficientes e viáveis para o processamento em dispositivos portáteis.

1.2 Abordagem

Em geral, o desenvolvimento de um sistema RAH baseado em dados de sensores é divido em cinco etapas: sensoriamento, pré-processamento, segmentação, extração de características e classificação (Bulling et al., 2014). Na primeira etapa, os sensores selecionados são iniciados e amostras são obtidas e armazenadas em memória continuamente ao longo do tempo. Na segunda etapa, os dados coletados são organizados, limpos e filtrados se necessário. Na terceira etapa, os dados são formatados e divididos em segmentos de tamanho fixo. Na quarta etapa, os segmentos são submetidos aos

métodos de extração de características. Por ultimo, na quinta etapa, os dados obtidos da etapa anterior são utilizados para inferir a atividade ou para auxiliar na criação de um modelo de classificação.

Com base nessas cinco etapas, muitos trabalhos na literatura são propostos para reconhecer atividades físicas a partir dos dados do acelerômetro. Uma revisão da literatura é apresentada no Capítulo 3. Em síntese, grande parte dos trabalhos implementa uma abordagem manual de extração de características discriminativas dos sinais dos sensores. Isto é, um especialista determina quais características serão utilizadas a partir de conhecimento prévio do problema ou de conhecimento adquirido explorando os dados. As características selecionadas, em geral, são obtidas a partir de dois domínios de análise: domínio do tempo (e.g., média, desvio padrão e variança) e domínio da frequência (e.g., frequência média, energia espectral e entropia espectral) (Sousa et al., 2017).

Outra abordagem, recentemente, encontrada na literatura é a utilização de algoritmos de aprendizagem de máquina profunda, ou também, conhecidos como rede neurais profundas (Plötz & Guan, 2018). Nesses algoritmos, o especialista não necessita selecionar as características manualmente e a rede neural profunda é encarregado de aprender automaticamente quais características devem ser extraídas. Essa abordagem torna o projeto de sistemas RAH automatizado e reduz os erros que podem ocorrer com o especialista.

Apesar da bons resultados obtidos nas abordagens apresentadas anteriormente, este trabalho explora lacunas deixas por trabalhos anteriores. A primeira consiste em viabilizar a extração de características no domínio discreto. Nesse domínio, os dados são discretizados e reduzidos a manipulação de um conjunto limitado de símbolos discretos, o que possibiliza uma eficiência no uso dos recursos de processamento e memória (Figo et al., 2010).

Por ultimo, este trabalho explora a abordagem de extrair características automaticamente, com o mínimo de interação do especialista. Neste caso, técnicas de processamento de texto, como *Bag-Of-Words*, são utilizadas em contrapartidas as soluções com redes neurais profundas. As redes neurais não são utilizados, devido a que esses algoritmos requerem a introdução de múltiplas camadas ocultas e neurônios para gerar modelos de classificação, o que aumenta a complexidade da solução e exige recursos como GPUs (do Inglês, *Graphics Processing Unit*). Embora não seja um problema para plataformas de alto desempenho, essa prática torna os algoritmos de aprendizagem profunda desfavoráveis para dispositivos com recursos limitados como *smartphones* (Ravi et al., 2016).

1.3 Objetivos

O objetivo deste trabalho é desenvolver um método capaz de reconhecer atividades humanas baseado na representação simbólica dos dados extraídos de sensores inercias embutidos nos *smartphones*. O método deve ser projetado para manipular quantidade massiva de dados a baixo custo computacional, extrair características de forma automática e gera uma representação compacta dos dados de multiplas séries temporais para classificação de padrões de atividades.

Para alcançar esse objetivo, os seguintes objetivos específicos devem ser atingidos:

- Definir e implementar o método de transformação simbólica e extração automática de características na perspectiva do domínio discreto tendo como dados de entrada uma série temporal;
- Definir, estender e implementar a estratégia de estruturação e representação dos dados baseada na construção de histogramas a partir de conjuntos de palavras (do Inglês, *Bag-Of-Words*). Esses histogramas são adaptados para processar séries temporais de múltiplas variáveis de entrada e utilizados para gerar modelos de classificação;
- Definir e implementar a estratégia de representação dos dados no modelo espaço vetorial para obter modelos de classificação compactos e eficientes para implementação em dispositivos móveis;

1.4 Abordagem Proposta

Esta pesquisa propõe uma solução RAH de baixo custo capaz de extrair características relevantes dos dados de atividades de forma automática. O método desenvolvido é capaz de processar grandes quantidades de dados em dispositivos móveis com poucos recursos computacionais e gerar modelos de classificação generalizados para RAH. Além disso, o método introduz as mais recentes técnicas de representação simbólica utilizados na área de classificação de séries temporais.

As principais vantagens das técnicas de representação simbólica são o processamento de baixo custo e a capacidade de redução da dimensionalidade e numerosidade dos dados (Bagnall et al., 2016). Na representação simbólica, uma série temporal é representada por uma cadeia de símbolos ou palavras. Cada palavra representa as propriedades estruturais da série temporal e podem ser utilizadas como características para a tarefa de classificação. Na literatura, poucos trabalhos utilizam a representação

simbólica no problema RAH (Figo et al., 2010; Siirtola et al., 2011; Terzi et al., 2017). Dentre os trabalhos analisados, dois aspectos importantes são destacados: os trabalhos aplicam o método *Symbolic Aggregate approXimation* (SAX) na transformação simbólica e nenhum dos trabalhos utiliza a abordagem proposta por este trabalho no que diz respeito a etapa de extração automática de características.

Nesse sentido, esta pesquisa apresenta o método *Multivariate Bag-Of-SFA-Symbols* (MBOSS) capaz de extrair características de forma automática com o mínimo de intervenção do especialista e utilizá-las para gerar classificadores de atividades humanas. Para isto, três técnicas são utilizadas:

- Representação simbólica de séries temporais: os dados dos sensores são submetidos à redução de ruído e dimensionalidade por meio de uma transformação de valores numéricos para palavras. Essa transformação é aplicada utilizando o método *Symbolic Fourier Approximation* (SFA) (Schäfer & Högqvist, 2012). Ao contrário do método SAX utilizado em trabalhos da literatura, o SFA se apresenta como um novo método para a área de RAH. A Seção 4.3 apresenta detalhes desse método;

- Construção da distribuição de frequência das palavras por meio de histogramas: os dados obtidos da representação simbólica são utilizados para construir uma representação baseada na contagem das repetições das palavras. Esse processo é igual ao proposto pelo método *Bag-Of-SFA-Symbols* (BOSS) (Schäfer, 2016), que é uma extensão do método SFA para classificação de séries temporais.

- Fusão de histogramas: os dados obtidos pelos sensores inerciais são representados em três sinais de entrada (eixos x, y e z), portanto, os histogramas obtidos para cada sinal de entrada são processados para compor uma única representação de dados. Esse procedimento é feito a nível de característica e por meio de uma concatenação de histogramas. Consideramos está técnica com uma extensão do BOSS para lidar com séries temporais de múltiplas variáveis;

- Método de classificação baseado no modelo *term frequency-inverse document frequency* (TF-IDF): modelos de classificação compactos e eficientes são gerados para RAH utilizando as tradicionais técnicas da área de recuperação de informação. A Seção 4.5 apresenta mais detalhes desse modelo;

Para a avaliação da abordagem proposta são utilizadas três bases de dados de atividades: WISDM (Kwapisz et al., 2011), UCI HAR (Anguita et al., 2013b) e UniMib SHAR (Micucci et al., 2017). Essas bases são frequentemente aplicadas na avaliação

de trabalhos RAH da literatura. Cada base de dados impõe diferentes configurações e atributos que ajudam a evidenciar com maior profundidade o desempenho do método. Também é realizado um comparativo entre o método proposto e classificadores baseados na extração manual de características. Por fim, os classificadores são submetidos a duas estratégias de avaliação: modelo personalizado, onde se verifica o desempenho de modelos de RAH construídos a partir dos dados de um único usuário; e modelo generalizado, onde se verifica o desempenho de modelos de RAH construídos a partir de um conjunto de usuários.

Para avaliar a eficiência do método proposto, experimentos são conduzidos para avaliar os tempos de computação para extrair as características dos dados, treinar os modelos de classificação e testar os modelos. Além disso, expermentos são realizados para verificar o consumo de recursos de mémoria, na qual se comparou o tamanho de espaço de dados antes a após a representação símbolica.

1.5 Contribuições

As contribuições deste trabalho são:

- Introdução de uma nova solução para RAH baseada na estratégia de representar os dados numéricos em conjunto de palavras. A solução apresenta vantagens relacionadas a redução de dimensionalidade e numerosidade dos dados, extração de características automáticas e uso de algoritmos de baixo custo computacional que manipulam uma grande quantidade de dados. Além disso, a solução demostra um desempenho equivalente de generalização em comparação a tradicional abordagem de extrair e selecionar manualmente as características para gerar modelos de RAH;

- Uma extensão dos métodos de representação simbólica, denominado de MBOSS, é apresentada para viabilizar o processamento de séries temporais de múltiplas variáveis. Essa extensão é avaliada para o problema de RAH, entretanto, ela pode ser generalizada para outras áreas que necessitam processar dados de múltiplas fontes de dados;

- Com a implementação do método de transformação simbólica SFA, novos conhecimentos e resultados são introduzidos a área de RAH, visto que no processo de revisão da literatura somente encontramos trabalhos relacionados com a implementação do método SAX;

As contribuições deste trabalho foram publicadas em formato de artigo no XXIV Simpósio Brasileiro de Sistemas Multimídia e Web (WEBMEDIA 2018), sob o título de "*Human Activity Recognition on Smartphone using Symbolic Data Representation*".

1.6 Organização do Texto

O restante desta dissertação está organizado da seguinte forma:

- Capítulo 2 aborda os conceitos teóricos sobre os fundamentos que norteiam o desenvolvimento de sistemas RAH. Entre os conceitos apresentados estão os tipos de atividades que podem ser reconhecida por esses sistemas, os dispositivos e sensores utilizados para a tarefa e metodologia tipica aplicada ao processamento dos dados.

- Capítulo 3 apresenta os trabalhos relacionados. Neste capítulo as diferentes abordagens de extração de características e métodos de classificação são citadas. Os trabalhos são categorizados em dois grupos: extração manual de características (do Inglês, *Handcraft Features*) e extração automática de características (do Inglês, *Non-Handcrafted Features*). Ao final, um resumo dos desempenhos obtidos por esses métodos é apresentado para diferentes estratégias de avaliação e base de dados.

- Capítulo 4 apresenta os métodos de representação simbólica e classificação de séries temporais utilizados com base para o desenvolvimento deste trabalho. Os métodos de representação simbólica apresentados são: *Symbolic Fourier Approximation* (SFA), *Bag-of-SFA-Symbols* (BOSS) e *Bag-Of-SFA-Symbols in Vector Space* (BOSS VS).

- Capítulo 5 apresenta uma discussão sobre como a fusão de dados de sensores pode ser aplicada para adaptar os métodos de representação simbólica para lidar com séries temporais multivariadas. Em seguida, o método Multivariate Bag-Of-SFA-Symbols (MBOSS) é apresentado e detalhado sua implementação para RAH. Por último, o capítulo encerra apresentando uma discussão das vantagens e desvantagens do método proposto.

- Capítulo 6 apresenta o protocolo experimental proposto para avaliar o desempenho do MBOSS para o problema de classificação de atividades humanas. O protocolo divide os experimentos em duas partes: A primeira parte, os experimentos são projetados para avaliar a eficácia de classificação para diferente bases

de dados de atividades e configuraçoes experimentais. A segunda parte, os experimentos são projetados para avaliar a eficiência do método em termos de tempo de computação e espaço de dados. O capítulo encerra apresentado os resultados e as análises para cada uma das partes do protocolo.

- Capítulo 7 apresenta as conclusões obtidas neste trabalho, incluindo uma síntese dos resultados, contribuições e trabalhos futuros sugeridos.

Capítulo 2

Reconhecimento de Atividades Humanas

Este capítulo descreve os principais conceitos relacionados ao desenvolvimento de sistemas de Reconhecimento de Atividades Humanas (RAH) baseado em sensores inerciais. Os conceitos ajudam a compreender os tipos de atividades que são reconhecidas nesta pesquisa, as tecnologias de sensoriamento e a metodologia comum no desenvolvimento de sistemas RAH.

2.1 Atividades Humanas

Atividades humanas podem ser definidas como sendo um conjunto de ações executadas pelo usuário em um determinado ambiente (e.g. caminhar, sentar, ficar em pé, cozinhar e escovar os dentes) (Cook & Krishnan, 2015). Computacionalmente, as atividades são definidas como sendo um conjunto de padrões reconhecidos por meio da análise de dados de sensores disponíveis no ambiente ou fixados ao usuário. Assim, os padrões identificados são traduzidos como atividades. Formalmente, essas atividades são definidas como uma sequência de eventos, $E = \{e_1, e_2, \ldots, e_i, \ldots, e_n\}$, capturados ao longo do tempo. Cada evento e_i possui a forma $e_i = (t, s, m)$, onde t representa o tempo, s o sensor e m a mensagem do sensor.

Na literatura, as atividades humanas podem ser categorizadas em simples e complexas (Shoaib et al., 2016). As atividades simples são definidas a partir da análise de dados de um ou mais sensores em um curto período de tempo como, por exemplo, caminhar, correr e sentar. Atividades complexas são definidas a partir de uma sequência de atividades simples que são executadas durante um longo período de tempo como, por exemplo, cozinhar, trabalhar e fazer compras.

O presente trabalho investiga as atividades simples relacionadas ao movimento do corpo humano (atividades físicas) que são executadas diariamente pelos usuários como, por exemplo, andar, correr, subir e descer escadas, deitar, entre outras. Esse grupo de atividades é frequentemente estudado por sistemas de RAH baseado em sensores inerciais (Lara & Labrador, 2013).

2.2 Sensores

Na literatura dois tipos de sensores são aplicados para RAH: sensores de ambiente e sensores vestíveis (Lara & Labrador, 2013). Os sensores de ambiente são aqueles instalados em um ponto fixo do ambiente ou em objetos. Exemplos de sensores desse tipo incluem câmeras, microfones, sensores de presença, sensores de proximidade, sensores em objetos como portas, janelas, aparelhos domésticos, chuveiros e torneiras. Por outro lado, os sensores vestíveis são aqueles instalados próximos ao corpo do indivíduo, inseridos em acessórios ou em dispositivos móveis. Exemplos de sensores dessa categoria incluem sensores inerciais (acelerômetros e giroscópios), sensores atmosféricos, fisiológicos e de localização. A Figura 2.1 ilustra um exemplo prático de como são utilizados os dois tipos de sensores em um ambiente físico.

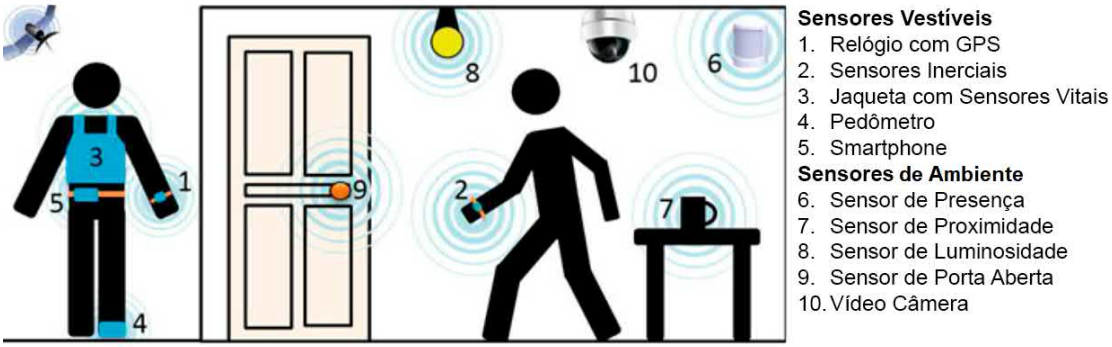


Figura 2.1. Exemplos de sensores de ambiente e sensores vestíveis para o problema RAH. Fonte: Adaptada de Ortiz (2015).

Dentre os dois tipos de sensores, os sensores vestíveis têm apresentado uma opção abrangente no desenvolvimento de soluções de RAH (Ortiz, 2015). Esses sensores podem ser utilizados para explorar informações relacionadas à movimentação do usuário (e.g., velocidade e aceleração), variáveis de ambiente (e.g., temperatura, umidade e pressão atmosférica) e sinais fisiológicos (e.g. batimento cardíaco e pressão sanguínea). Além disso, esses sensores podem ser fixados em diferentes partes do corpo como a

cintura, pulso, peito, pernas e cabeça, ou em acessórios como roupas, óculos, capacetes, calçados, relógios e *smartphones* (Sztyler & Stuckenschmidt, 2016; Lin et al., 2016; Weiss et al., 2016a,b).

2.3 Smartphones

Diferentes tipos de sensores podem ser encontrados nos *smartphones* recentes. A Tabela 2.1 apresenta uma classificação desses sensores de acordo com seus respectivos propósitos (Lane et al., 2010). Por exemplo, os sensores inerciais são utilizados para capturar os movimentos dos usuários. Os sensores acústicos medem a vibração dos sinais sonoros permitindo o reconhecimento de vozes e identificação de ruídos no ambiente. Os sensores magnéticos podem ser utilizados para medir a intensidade, direção e sentido do campo magnético terrestre. Os sensores de localização podem ser utilizados para informar a posição geográfica do usuário.

Dentre esses tipos, os sensores inerciais se destacam por serem pequenos, baratos e confiáveis. Devido aos avanços da tecnologia MEMS (*MicroElectroMechanical Systems*), esses sensores podem ser encontrados em diversas aplicações como, por exemplo, em *smartphones*, carros com sistemas de *air-bags* e sistemas de navegação em aviões (Tian & Chen, 2016).

Tabela 2.1. Tipos de sensores presentes em *smartphones*.

Tipos	Exemplos de sensores
inerciais	acelerômetro e giroscópio
magnéticos	magnetômetro
acústicos	microfone
óticos	infravermelho
atmosféricos	temperatura, umidade e pressão
ambiente	luz e proximidade
localização	GPS
vídeo e imagem	câmeras
rádio	GSN, WiFi e Bluetooth
fisiológicos	frequência cardíaca

2.4 Sensores Inerciais para RAH

Nos *smartphones* os sensores inerciais capturam os movimentos do dispositivo em um sistema de coordenadas espaciais como exemplificado na Figura 2.2. Para o acelerômetro, os dados capturados são representados em três coordenadas, ou também chamadas de direções x, y e z. Igualmente para o giroscópio, na qual a coordenada *roll* mede a rotação do dispositivo nas direções norte e sul (frente-atrás), a coordenada *pitch* mede a rotação do dispositivo para leste e oeste (esquerda-direita) e, por fim, a coordenada *yaw* mede a rotação para cima e para baixo.

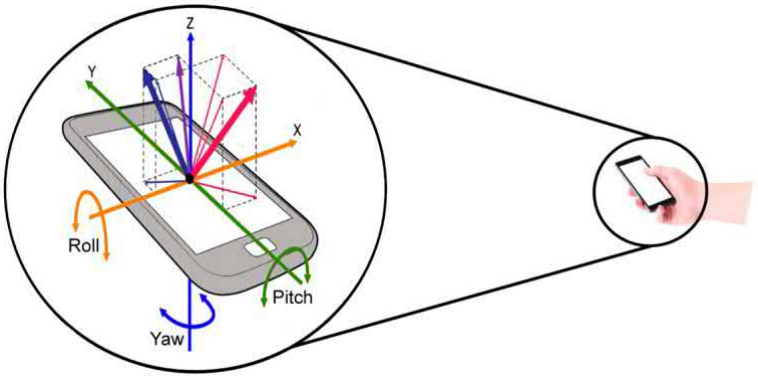


Figura 2.2. Sensores inerciais inseridos nos *smartphones*. Fonte: Tian & Chen (2016).

Uma vez que o usuário está carregando o *smartphone* consigo, os movimentos realizados pelo individuo são transferidos ao dispositivo que consequentemente são capturados pelos sensores inerciais. Os movimentos do usuário têm tendência de repetição em pequenos períodos de tempos, logo, os dados coletados auxiliam na identificação de padrões nas diferentes atividades. A Figura 2.3 exemplifica os dados coletados do acelerômetro para seis diferentes atividades. Os dados foram obtidos de um usuário utilizando um smartphone no bolso frontal da calça (Anguita et al., 2013b).

De acordo com a figura, as atividades *Ficar em pé* (A) e *Caminhar* (D) apresentam diferenças nos padrões de aceleração, na qual é clara a diferença de movimento de uma atividade considerada estacionária e uma atividade dinâmica. Por outro lado, determinar uma diferença entre padrões similares de movimento é um desafio. No caso das atividades *Subir escadas* (E) e *Descer escadas* (F), duas atividades dinâmicas, os padrões não apresenta uma diferença tão clara visualmente. Logo, processos computacionais podem ser aplicados para determinar essa diferença.

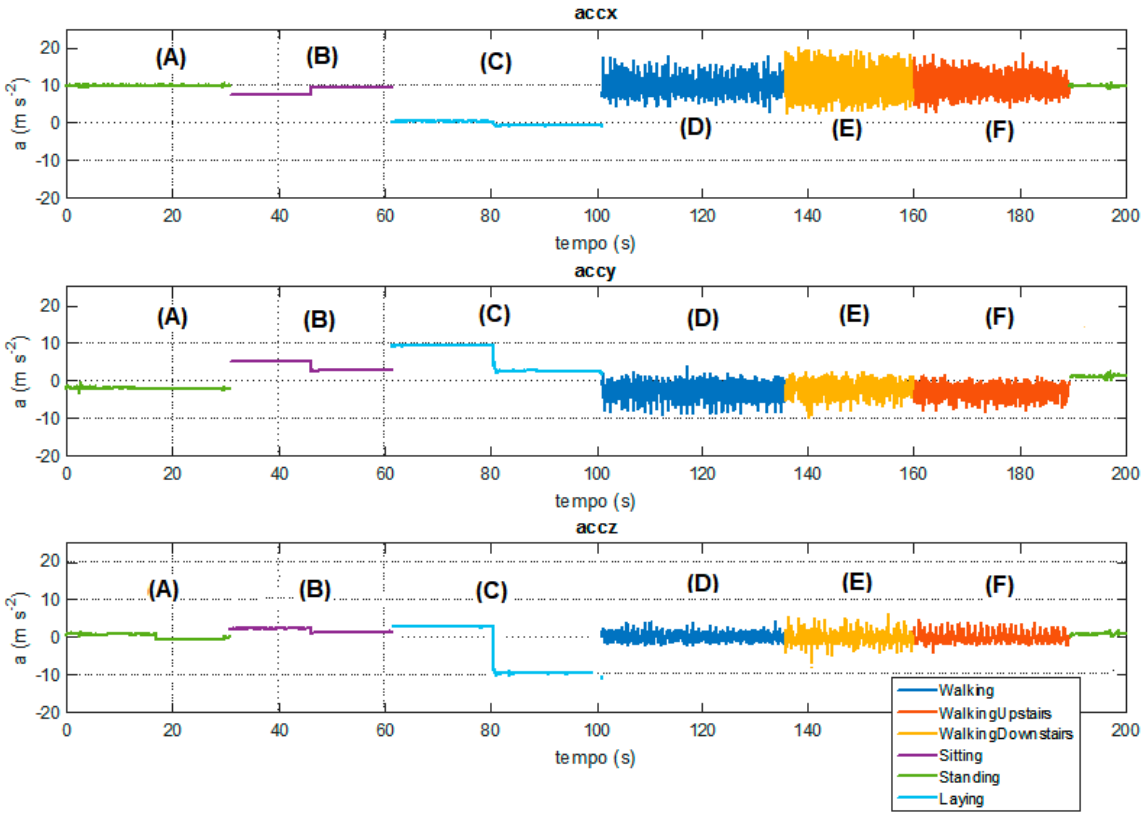


Figura 2.3. Formação de padrões nos sinais do acelerômetro para diferentes atividades: (A) Ficar em pé, (B) Sentar, (C) Deitar, (D) Caminhar, (E) Subir escadas, (F) Descer Escadas. Base de dados: UCI HAR (Anguita et al., 2013b). Fonte: Elaborado pelo próprio autor.

A implementação de sistemas computacionais para reconhecer e distinguir atividades humanas é conteúdo deste trabalho. Com os conceitos base esclarecidos, em sequencia é apresentado a metodologia comum aplicado no desenvolvimento de sistemas para RAH.

2.5 Metodologia para RAH

A metodologia aplicada para reconhecer as atividades humanas baseada em sensores é similar a um processo de reconhecimento de padrões de propósito geral (Ortiz, 2015). O processo abrange etapas que vão desde a tarefa de sensoriamento até a etapa de classificação das atividades. Entretanto, algumas etapas podem sofrer variações dependendo da aplicação de interesse e dos sensores utilizados. Além disso, para algoritmos de classificação supervisionados, a metodologia pode ser executada de dois modos: de treino (construção do modelo) e de classificação.

A Figura 2.4 apresenta esquematicamente a metodologia aplicada neste trabalho para RAH, chamada na literatura como ARC (do inglês, *Activity Recognition Chain*) (Bulling et al., 2014).

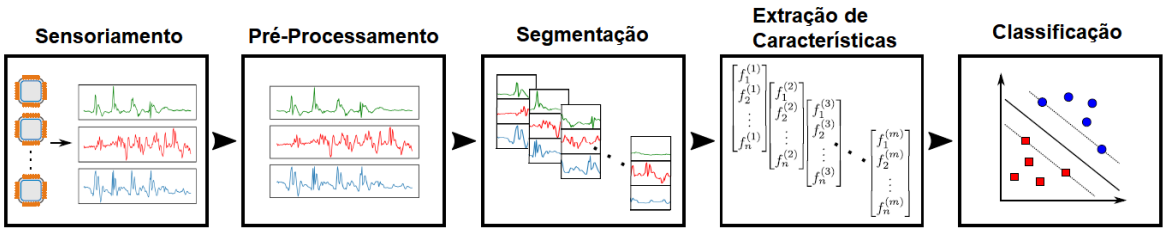


Figura 2.4. Etapas comuns para o desenvolvimento de um sistema RAH baseado em sensores. Fonte: Adaptada de Bulling et al. (2014).

A metodologia é composta pelas seguintes etapas que serão descritas a seguir.

Sensoriamento: esta etapa inicial seleciona e configura os sensores para coleta dos dados. Os sensores são escolhidos de acordo com o tipo de atividade que se deseja identificar. Os dados coletados consistem em séries temporais armazenadas em memória. Uma vez que alguns sensores fornecem múltiplas variáveis de dados (e.g., acelerômetro tri-axial), as séries temporais são coletadas e armazenadas individualmente.

Pré-processamento: esta etapa é responsável pela adequação dos dados em formatos específicos com o propósito de preparar os dados para serem manipulados nas demais etapas de processamento. Exemplos de técnicas que podem ser utilizadas nessa etapa são: calibração de sensores, conversão de unidades, normalização e limpeza de dados corrompidos causados por possíveis falhas de hardware. Nos casos que envolvem o uso de vários sensores, a sincronização desses dados também é considerada.

Segmentação: esta etapa é responsável por separar a série temporal em segmentos de tamanhos específicos. A intenção é que cada segmento contenha informações sobre uma determinada atividade. Nessa fase, os segmentos podem ser filtrados com o propósito de eliminar possíveis segmentos sem informações de atividades.

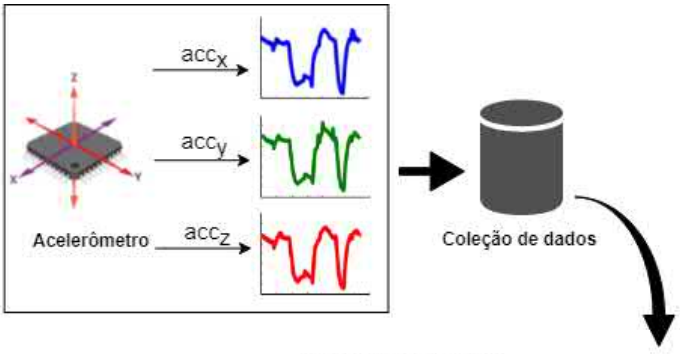
Extração de características: esta etapa é responsável pela transformação dos dados brutos e segmentados em uma representação de alto nível com as características relevantes para o problema. Encontrar uma representação ótima é importante para assegurar a capacidade de generalização dos sistemas de RAH. A representação pode ser obtida de forma manual com base em um conhecimento prévio sobre os dados ou de forma automática com auxílio algoritmos computacionais. Os procedimentos realiados para obter essas representações são os objetos de estudo desta pesquisa e são apresentados com mais detalhes nas próximas seções.

Classificação: esta etapa corresponde às tarefas de treino e classificação dos modelos de classificação de atividades. Na primeira, uma base de dados de atividades é utilizada para treinar e avaliar modelos de classificação. Esses modelos são gerados a partir da aplicação de algoritmos de classificação de aprendizagem de máquina. Na segunda tarefa, com o modelo treinado, a classificação de novas instâncias pode ser executada.

A seguir esta seção apresenta os principais conceitos e técnicas utilizadas para cada uma das etapas da metodologia ARC.

2.5.1 Sensoriamento e Pré-processamento

Para desenvolver um sistema RAH, as primeiras tarefas a serem realizadas estão relacionadas às definições de quais atividades se desejam reconhecer e quais sensores serão utilizados. Como apresentado anteriormente, atividades de movimentos podem ser reconhecidas a partir dos dados do acelerômetro. Entretanto, na literatura os trabalhos existentes não se restringem a utilização de um único sensor, mas exploram a combinação com outros como o microfone (Lu et al., 2009), giroscópio (Shoaib et al., 2014) ou GPS (Mousavi et al., 2017).



Coleta de dados

Tempo	Acc$_x$	Acc$_y$	Acc$_z$	Atividade (Rótulo)	ID Usuário
00:00:000	0.692	10.832	-2.038	ANDANDO	1
00:00:050	6.853	7.443	-0.598	ANDANDO	1
00:00:100	0.934	5.636	-0.597	ANDANDO	1
...	...	...	...	...	

Figura 2.5. Formato dos dados coletados do acelerômetro. Fonte: Elaborado pelo próprio autor.

Uma vez definidos os sensores, a próxima tarefa consiste da coleta de dados e pré-processamento. Os dados são organizados e armazenados cronologicamente em forma de séries temporais em memória. Além dos dados dos sensores, informações extras são coletadas como instante de tempo, identificação do usuário e o rótulo da atividade. A

Figura 2.5 exemplifica a coleta de dados, na qual uma estrutura em formato de tabela é utilizada para representação. São coletados os dados do acelerômetro para cada direção (acc_x, acc_y , acc_z), a atividade executada no instante de tempo e o tempo. Após a coleta, o pré-processamento é opcional. Alguns trabalhos aplicam algumas técnicas de filtragem para eliminação de ruídos e normalização dos dados (Anguita et al., 2013b).

Na literatura são encontrados trabalhos que na execução da etapa de sensoriamento geraram bases de dados e as publicaram para o desenvolvimento de novas soluções para RAH. As bases são construídas a partir de múltiplos indivíduos e utilizando diferentes tipos de sensores. Dentre elas, citamos a WISDM (Kwapisz et al., 2011) e UniMib SHAR (Micucci et al., 2017) com dados obtidos do acelerômetro e a UCI HAR (Anguita et al., 2013b) com dados obtidos do acelerômetro e giroscópios. Essas três bases são frequentemente utilizadas para avaliar sistemas RAH e são utilizadas neste trabalho. Uma descrisção detalhada das bases é apresentada na Seção 6.2.1.

2.5.2 Segmentação

A segmentação é responsável por dividir as séries temporais em blocos menores definidos como segmentos ou janelas de tempo. Cada segmento contém uma quantidade limitada de amostras que é definida pelo seu tamanho. Um segmento de dados $w_i = (t_i, t_f)$ possui um tempo inicial t_i e um tempo final t_f. Logo, a segmentação consiste em obter um conjunto de segmentos $W = \{w_1, ..., w_m\}$. Esse processo é realizado considerando também as múltiplas séries temporais disponíveis na base de dados.

A Figura 2.6 mostra um exemplo da formatação dos dados após o processo de segmentação das séries temporais obtidas de um acelerômetro tri-axial. Temos as seguinte variáveis:

- "w_i", que representa o identificador do segmento, variando de 1 até m;
- "m" corresponde a quantidade total de segmentos obtidos;
- "s" corresponde ao tamanho de cada segmento.

As principais técnicas de segmentação podem ser categorizadas em três grupos: janela definida por atividade, janela definida por evento e janela definida por tempo (Banos et al., 2014).

A janela definida pela atividade consiste em um particionamento do fluxo de dados do sensor com base na detecção de mudanças de atividades. Essa detecção pode ser realizada aplicado algum tipo de processamento prévio. Um exemplo poderia ser a análise das variações das características de frequência dos sinais. Utilizando essa

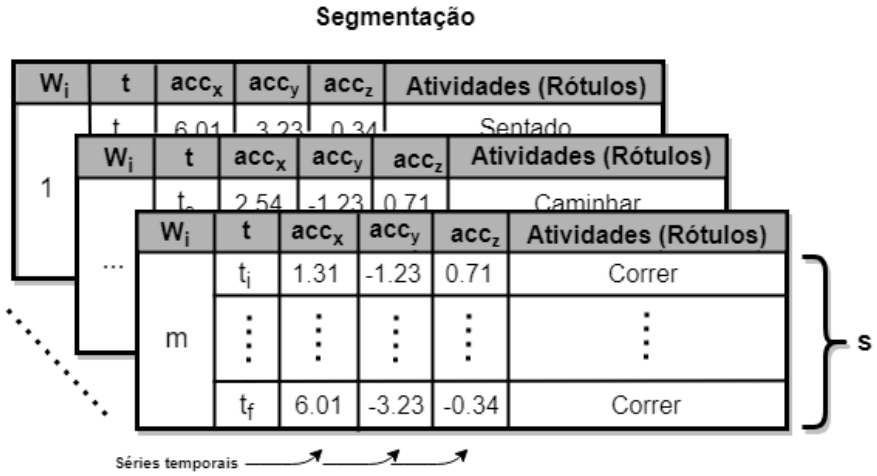


Figura 2.6. Formato dos dados após o processo de segmentação. Fonte: Elaborado pelo próprio autor.

abordagem se encontram os trabalhos de Sekine et al. (2000) e Nyan et al. (2006). Ambos utilizam um modelo baseado na decomposição de Wavelets para detectar as mudanças entre três tipos de atividades: caminhar, subir e descer escadas.

A janela definida por evento consiste em detectar eventos específicos para então utilizá-los para particionar o fluxo de dados do sensor. Por exemplo, na análise da marcha (do Inglês, *Gait analysis*), os impactos do calcanhar no solo (evento) são identificados analisando a aceleração linear do pé (Benocci et al., 2010). Esse evento detectado indica um ponto de segmentação e dados são segmentados com base nesse evento. Em certos casos, a segmentação não é continua, mas somente quando um evento especifico é detectado.

Ao contrário das abordagens anteriores, a janela definida por tempo não requer nenhum pré-processamento nos dados e os dados são particionados continuamente. Também chamada de "janela deslizante" ou "janelamento", este procedimento é o mais utilizado para RAH, principalmente, nas aplicações que buscam processamento em tempo real (Bersch et al., 2014). Nesta abordagem, o fluxo de dados é particionado em janelas de tamanho fixo e sem intervalos entre janelas. Uma sobreposição entre janelas adjacentes é tolerada para certas aplicações, e chama-se de janela deslizante com sobreposição. Nas janelas sobrepostas, a quantidade de informação compartilhada é definida por uma porcentagem de sobreposição. Por exemplo, 50% de sobreposição significa que 50% dos dados da janela anterior pertencerão, também, aos 50% dos valores da janela posterior. A Figura 2.7 mostra um exemplo do processo de segmentação para janelas de tamanho fixo igual a 128 amostras com 50% de sobreposição. Neste caso, uma série temporal de tamanho 450 é segmentada em 6 janelas de 128 amostras.

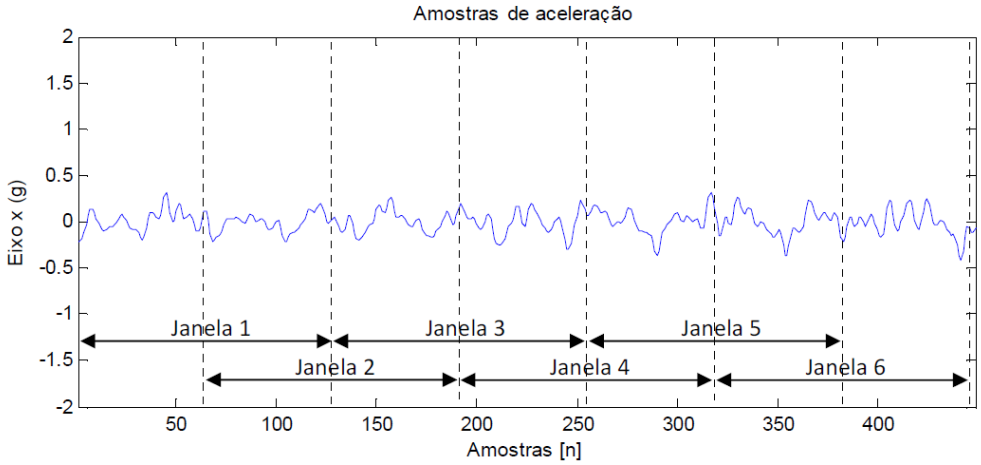


Figura 2.7. Segmentação: Janela deslizante de tamanho fixo igual a 128 com 50% de sobreposição entre janelas vizinhas. Fonte: Adaptado de Silva (2013).

2.5.3 Extração de Características

O processo de extração de características tem como objetivo transformar os dados brutos em dados compactos e com informações relevantes para representar bem determinada atividade e auxiliar na discriminação das atividades na etapa de classificação (Ignatov, 2017). Para isto, características são selecionadas para sumarizar e descrever os segmentos de atividades. Essas características são descritores ou métricas aplicadas aos dados e podem ser utilizadas para representar as diferentes propriedades, padrões e tendências encontradas nos dados dos sensores.

Na literatura, as soluções para RAH baseado em sensores inercias têm apresentado uma gama grande de características que podem ser aplicados para representar padrões de atividades. Essas características podem ser categorizadas de acordo com domínio de análise a qual é aplicada a característica. Os domínios apresentados são: o domínio de tempo, o domínio da frequência e o domínio discreto (Figo et al., 2010). Exemplos de características nessas três categorias são apresentados na Figura 2.8.

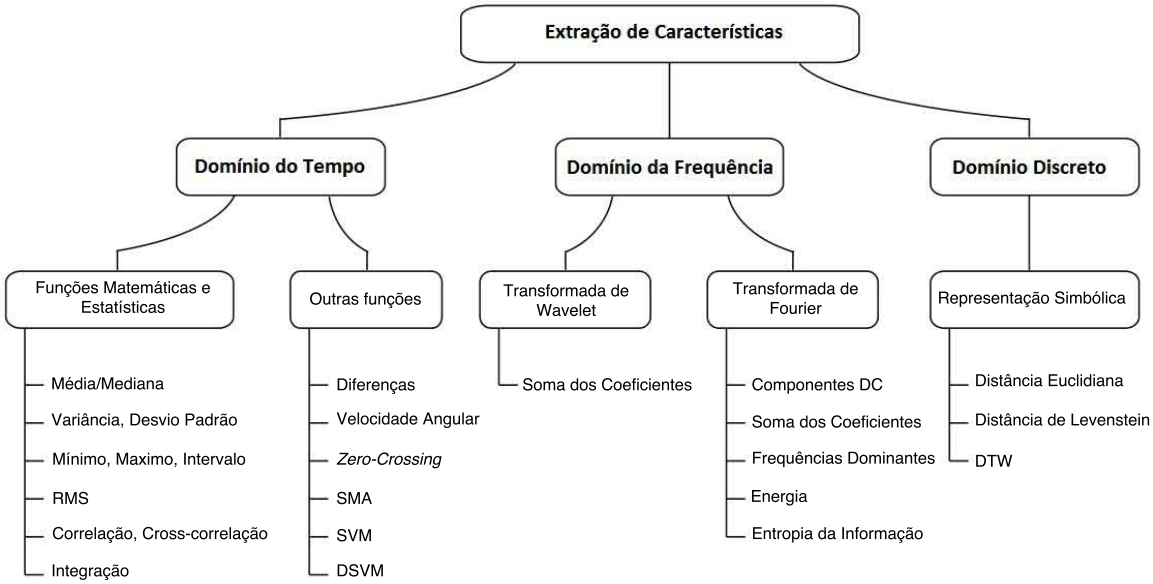


Figura 2.8. Exemplos de características utilizadas para RAH para diferentes domínios de análise. Fonte: Adaptada de Figo et al. (2010).

As categorias apresentam as seguintes características:

Domínio do Tempo: As características neste domínio são baseadas em cálculos matemáticos e estatísticos realizadas sobre um conjunto de amostras ao longo do tempo (série temporal), neste caso o conjunto é um segmento de dados do sensor. Um segmento pode, por exemplo, ser representado pelas seguintes características: média dos valores, variância e desvio-padrão. Essas características e outras mais são selecionadas de acordo com os dados observados e utilizados para treinar classificadores em grande parte dos trabalhos para RAH.

Domínio da Frequência: As características neste domínio apresentam uma alternativa à opção de análise no domínio do tempo. Elas são baseadas no espectro da frequência do conjunto de amostras (e.g. Frequências dominantes, Energia). Métodos como a transformada discreta de Fourier (DFT) ou transformada de Wavelet são frequentemente utilizadas para transformar os dados brutos para uma representação no domínio da frequência. Deste modo, novas características sobre essa representação podem ser obtidos para representar padrões de atividades. As características no domínio da frequência são ótimas para representar a natureza repetitiva dos sinais, na qual repetições se correlacionam com a natureza periódica de atividades como, por exemplo, caminhar e correr (Figo et al., 2010).

Domínio Discreto: As características neste domínio têm como base um processo de aproximação e discretização de um conjunto de dados contínuos em um conjunto de dados discretos (Terzi et al., 2017). Esse processo, chamado de representação sim-

bólica, possibilita que os dados numéricos sejam tratados agora por meio de símbolos ou palavras e que novas características no domínio discreto sejam obtidas para representar padrões de atividades. Essas características, em geral, são cálculos de distância e similaridade de par de palavras, como distância mínima entre palavras, distância Euclidiana e distância de Levenstein (Figo et al., 2010).

Para o processo de representação simbólica, o algoritmo, denominado *Symbolic Aggregate Approximation* (SAX), é repetidamente aplicado no processo de extração de características no domínio discreto. Baseada na aproximação do Piecewise Aggregate Approximation (PAA), esse método é capaz de descrever de maneira compacta a série temporal e discretizar os dados em um conjunto limitado de símbolos e tamanho de palavra fixo, reduzindo os custo de processamento e armazenamento das soluções que as utilziam (Lin et al., 2007). Além disso, palavras representando atividades podem ser obtidas como protótipos de comparação. Logo, esses protótipos são utilizados para obter as características e auxiliar a discriminação de padrões de atividades (Figo et al., 2010). A Figura 2.9 exemplifica o processo de representação simbólica, onde são extraídas palavras de segmentos de uma série temporal, com um conjunto de 4 símbolos e tamanho de palavra igual a 8.

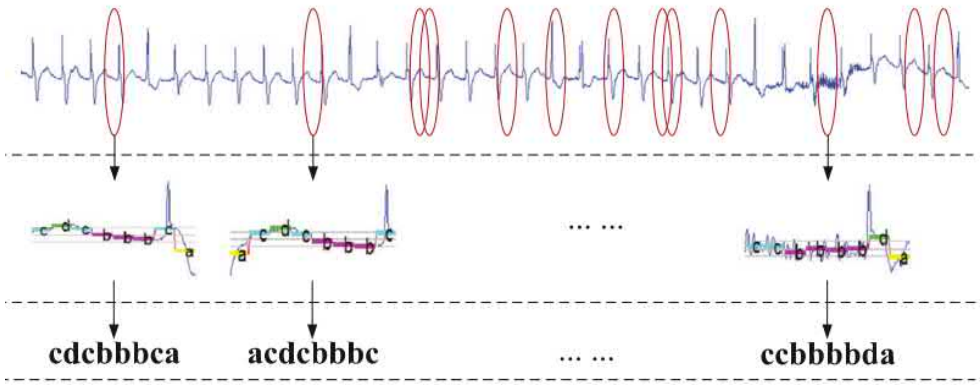


Figura 2.9. Extração de características no domínio discreto. Palavras obtidas através da representação simbólica com o SAX. Fonte: Adaptada de Lin & Li (2009).

2.5.4 Classificação

A etapa de classificação, por sua vez, consiste na aplicação de algoritmos de aprendizagem de máquina capazes de aprender os padrões dos dados e inferir as atividades humanas. Em geral, as características obtidas na etapa de extração são utilizadas para gerar os modelos de classificação com a aplicação de algoritmos de aprendizagem de máquina como Árvores de Decisão (DT), Naive Bayes (NB), k-Vizinhos-Próximos

(KNN), Máquinas de Vetores de Suporte (SVM) e Redes Neurais Artificiais (RNAs) (Shoaib et al., 2015).

Com os avanços na área de aprendizagem de máquina, principalmente, com o desenvolvimento da aprendizagem de máquina profunda, a etapa de extração de características e classificação tem sido implementadas a partir de diferentes tipos de arquiteturas de redes neurais profundas como Deep Fully-Connected Network (Lane et al., 2015), Convolutional Neural Network (Chen & Xue, 2015), Recurrent Neural Network (Hammerla et al., 2016), Long Short-Term Memory, Stacked Autoencoder (Ravi et al., 2016) e Restricted Boltzmann Machine (Ronao & Cho, 2016). Essas redes têm apresentado ganhos nas capacidades de representação e generalização dos dados e são uma alternativa a tarefa de seleção manual de características usualmente aplicada nas soluções de RAH (Nweke et al., 2018). Os resultados de desempenho de classificação dessas redes em comparação com os algoritmos de aprendizagem de máquina tradicionais são apresentados no capítulo seguinte.

2.6 Considerações Finais

Ao longo deste capítulo foram apresentados diversos conceitos e propostas relacionadas a tarefa de desenvolver sistema para RAH. Foram apresentados os tipos de atividades que podem ser reconhecidas por esses sistemas e como diferentes sensores podem ser utilizados para esse problema.

Dentre os diferentes tipos de sensores, foram destacados os sensores inerciais presentes em *smartphones*, como o acelerômetro e o giroscópio, e como eles podem ser utilizados para reconhecer atividades físicas. Foi apresentada também a metodologia predominante na literatura para desenvolver sistema de RAH. Um destaque maior foi apresentado às etapas de extração de características e classificação.

O próximo capítulo apresentará os resultados obtidos de um processo de revisão da literatura sobre o tema proposto nesta dissertação. Diversos trabalhos relacionados foram investigados, categorizados e descritos segundo a maneira que as características são extraídas dos dados.

Capítulo 3

Trabalhos Relacionados

Este capítulo apresenta uma síntese dos trabalhos da literatura analisados nesta pesquisa. Os trabalhos, em geral, são descritos com foco na abordagem realizada para extrair características e classificar os dados de atividades. Além disso, um resumo dos resultados experimentais obtidos por cada trabalho avaliado é exposto de acordo com as categorias propostas neste capítulo.

3.1 Revisão da Literatura

Ao longo do processo de revisão da literatura, alguns atributos comuns foram identificados nos trabalhos de RAH e utilizados para organizar esta seção. Esses são:

- Bases de dados públicas utilizadas para avaliar os métodos, como WISDM (Kwapisz et al., 2011), UCI HAR (Anguita et al., 2013a) e UniMib SHAR (Micucci et al., 2017).
- Abordagens utilizadas para extrair características, nas quais os trabalhos podem ser categorizados em dois grupos. O primeiro grupo trata dos trabalhos baseados na extração manual de características — *Handcraft Features* (HF) — e o segundo grupo trata dos trabalhos baseados na extração automática de características — *Non-Handcraft Features* (NHF).
- Estratégias de avaliação dos classificadores. Nesta última, diferentes procedimentos de avaliação são utilizados para avaliar os sistemas de RAH como *5-fold cross validation* ou *5-fold cv*, *leave-one-subject out* e 21-9 (dados de 21 usuários para treino e dados de 9 usuários para teste).

Uma síntese dos trabalhos relacionados é apresentado a seguir, na qual são descritas as técnicas utilizadas, os principais resultados e contribuições para a área RAH. Ao final, são apresentados os pontos de melhoria encontrados que justificam as direções assumidas neste trabalho.

3.1.1 Trabalhos Baseados em Handcraft Features

Esta subseção apresenta os trabalhos baseados em uma abordagem de extração manual de características. De maneira geral, os trabalhos se caracterizam por selecionar e avaliar as melhores características para o problema RAH e na aplicação de algoritmos de aprendizagem de máquina tradicionais, ou também chamados de rasos (do Inglês *shallow*). Um fator importante nesta abordagem é a necessidade de um especialista para determinar quais características devem ser utilizadas. Esta tarefa pode ser realizada de maneira exploratória ou baseada em resultados obtidos por trabalhos relacionados. Os trabalhos categorizados são apresentados em ordem cronológica a seguir.

Figo et al. (2010) realizaram uma avaliação exploratória abrangendo um total de quinze tipos de características no domínio do tempo (média, desvio padrão, mediana, mínimo, máximo, diferença entre mínimo e o máximo, *root mean square*, *integration*, correlação, correlação cruzada, diferenças entre amostras, *zero-crossings*, *signal magnitude area*, *signal vector magnitude*, *differential signal vector magnitude*), seis no domínio da frequência (energia espectral, entropia espectral, soma dos coeficientes de Fourier, frequência dominante, soma dos coeficientes de Wavelet) e três no domínio discreto (distância mínima entre palavras, *Dynamic Time Wrapping*, *Lavenstein*). A avaliação destaca as melhores características para o desenvolvimento de sistemas para RAH. Além disso, informações a respeito do custo de processamento e a viabilidade de implementação em dispositivos móveis para cada uma das características é apresentada. Experimentos foram realizados com dados obtidos do acelerômetro para três atividades físicas (pular, correr e caminhar). Os resultados obtidos mostram que as características estatísticas simples pertencentes ao domínio do tempo têm bom desempenho e são viáveis para execução em dispositivos móveis. As características no domínio da frequência derivadas da Transformadas Rápida de Fourier (FFT) e Waveles apresentam bons resultados de classificação, entretanto, não são recomendadas para dispositivos móveis devido ao alto custo computacional. As características no domínio discreto obtidas por meio da aplicação do SAX são apresentadas como viáveis para aplicações em dispositivos móveis. Entretanto, esse tipo de característica não apresenta desempenho de classificação superior em comparação com as demais características do estudo.

Com relação a trabalhos posteriores ao descrito anteriormente por Figo et al. (2010), uma maior adoção das características nos domínios do tempo e frequência foi observado para desenvolver soluções RAH. Alguns desses trabalhos são descritos a seguir.

Kwapisz et al. (2011) descrevem um processo de extração manual com seis tipos de características no domínio do tempo. Essas características são: média da aceleração, desvio padrão, diferença média absoluta, média da resultante da aceleração, tempo entre picos e a distribuição binária. As características selecionadas foram utilizadas para avaliar três algoritmos de classificação: Árvore de decisão, Regressor logístico e Rede Perceptron Multicamadas (MLP). Os experimentos foram executados em uma base de dados denominada WISDM, contendo informações de 36 usuários e seis atividades de movimento: caminhar, correr, subir escadas, descer escadas, sentar e ficar em pé. Nessa base, dados do acelerômetro foram coletados de diferentes *smartphones* Android. O procedimento de avaliação consistiu na execução da validação cruzada com particionamento de todas as instâncias de atividades em dez partes — *10-fold cross-validation*. Os resultados demostram a superioridade do classificador MLP em relação aos demais avaliados, com 91,7% de acurácia.

Siirtola et al. (2011) apresentam um método que combina características nos três diferentes domínios (tempo, frequência e discreto). A pesquisa teve como objetivo melhorar a acurácia dos sistemas de RAH com a inclusão de características no domínio discreto. Essas características são chamadas de SAXS (Symbolic Aggregate approXimation Similarity) e consistem no cálculo da similaridade entre palavras. As palavras são obtidas com a transformação simbólica do SAX e palavras de referência (*templates*) são selecionadas a partir das instâncias de treino e para cada classe de atividade. Deste modo, a similaridade é calculada entre as palavras de referência e a palavra da instância que se deseja classificar. Os experimentos foram conduzidos em cinco bases de dados. Cada base contém dados de diferentes tipos de aplicações no contexto de RAH. A primeira e a segunda base estão relacionadas à classificação de atividades em uma perspectiva de utilização de ferramentas. A terceira trata de reconhecimento de gestos. A quarta é para reconhecer atividades esportivas. A quinta é para reconhecer estilos de natação. Os classificadores avaliados foram: Árvore de decisão, K-Vizinho mais próximo (KNN), Naive Bayes. As características selecionadas no domínio do tempo são: média, desvio padrão, mediana, quartis, mínimo e máximo. E no domínio da frequência: soma das menores sequências da transformada de Fourier, a quantidade e a taxa de *zero crossings*. Os resultados demostram um aumento significante na acurácia (entre 3% a 10%) com a utilização combinada das características no domínio tempo, frequência e discreto em comparação ao conjunto com somente características

do domínio tempo e frequência. O uso de *smartphones* foi avaliado na base de dados relacionada ao uso de ferramentas. Nessa base de dados, os resultados experimentais demostram que o KNN é melhor entre os demais classificadores avaliados, com 85,3% de acurácia.

Anguita et al. (2013b) propôs um método para RAH com um extenso processo de extração de características, onde 17 características no domínio do tempo e da frequência foram selecionadas. Essas características foram aplicadas sobre os sinais brutos e pré-processados obtidos do acelerômetro e giroscópio. Os sinais pré-processados foram obtidos a partir da aplicação de filtros digitais, como *Butterworth low-pass*, para decomposição da aceleração entre sinais do corpo e da gravidade. Sinais adicionais também foram pré-processados como a magnitude euclidiana e as derivadas no tempo para aceleração linear e aceleração angular. Ao total foram extraídas 561 características para representar cada segmento de atividade. Os experimentos foram conduzidos na base de dados UCI HAR contendo instâncias de 6 atividades físicas como caminhar, subir escadas, descer escadas, sentar e ficar em pé e deitar. O algoritmo de classificação utilizado foi o SVM *One-vs-All*. O procedimento de avaliação consistiu no particionamento dos dados 21-9, ou seja, dados de 21 usuários são utilizados para treino e os dados de 9 usuários são utilizados para teste. Os resultados demostram um desempenho de 96% na acurácia.

Os trabalhos de Catal et al. (2015) e Walse et al. (2017) são extensão dos trabalhos de Kwapisz et al. (2011) e Anguita et al. (2013b), respectivamente. Ambos avaliaram a utilização de comitês de classificadores (*ensemble classifiers*) como o *bagging* e *boosting*. Os resultados demostram um aumento significante na acurácia comparado com as propostas sem a técnica de comitê, 94,3% e 98,06%, respectivamente, para Catal et al. (2015) e Walse et al. (2017).

Ronao & Cho (2017) apresentam um método de classificação baseado em modelos escondidos de Markov (HMM). O método é composto por dois níveis de HMMs. O primeiro nível classifica as instâncias em atividades estáticas (e.g., ficar em pé, sentar e deitar) e atividades dinâmicas (e.g. caminhar, subir e descer escadas), e o segundo nível classifica as instâncias do grupo específico em questão. Neste são utilizados tanto características no domínio do tempo e frequência. Os experimentos foram realizados na base de dados UCI HAR e o procedimento de avaliação foi 21-9. Os resultados demostram um bom desempenho de classificação, com acurácia de 93,18%.

Em seguida, Terzi et al. (2017) apresentam um método que extrai somente características no domínio discreto e três cálculos de similaridade são propostos para lidar com séries temporais multivariadas: distância média, mínima e geométrica. Da mesma maneira que Siirtola et al. (2011), o método proposto extrai templates com aplicação

da transformada simbólica do SAX e o cálculo da similaridade é aplicado para pares de palavras. Experimentos foram realizados na base de dados UCI HAR, mas somente para duas atividades. Os resultados demostram que a distância média tem um desempenho melhor que os demais cálculos de similaridade, com acurácias de 99,54% e 97,07%, respectivamente para caminhar e a combinação subir e descer escadas. Apesar do bom desempenho apresentado pelas características no domínio discreto, poucas são as atividades reconhecidas, uma vez que a base de dados original contém dados de seis classes de atividades.

Por último, Micucci et al. (2017) apresentam uma avaliação comparativa de classificadores como K-NN, SVM, Rede Neural Artificial e Random Forest (RF) para RAR em uma base de dados com nove atividades. A base de dados utilizada nos experimentos explora uma extensa diversidade de usuários com diferentes características como, por exemplo, idade, peso, altura e gênero. Os classificadores foram submetidos aos dados em modo crú (Raw), isto é, sem extração de características. Na avaliação foi estabelecido dois tipos de avaliação: 5-fold cross validation e leave-one-subject-out. Os resultados demostram que o classificador RF obteve a melhor acurácia entre os demais, com 88,41% e 73,17%, respectivamente, para o primeiro e segundo tipo de avaliação.

3.1.2 Trabalhos Baseados em Non-Handcraft Features

Nesta subseção são apresentados os trabalhos baseados em uma abordagem de extração automática de características. Esses trabalhos se caracterizam, em geral, por propor uma alternativa a tarefa de extração e seleção manual de características pelo especialista. Esses métodos permitem que os sistema descubram a melhor representação dos dados para a tarefa de classificação. Os trabalhos categorizados são apresentados em ordem cronológica a seguir.

Li et al. (2014) apresentam um framework que explora diferentes métodos para extrair características de forma automática. Os métodos investigados incluem Sparse Auto-Encoder (SAE), Denoising Auto-Encoder (DAE) e Principal Components Analisys (PCA). Experimentos foram realizados na base de dados UCI HAR. As características foram obtidas individualmente para cada fonte de sinal do acelerômetro e giroscópio. Além disso, a magnitude dos sinais foi calculada e utilizada para adicionar mais informações sobre as atividades. A representação final é formada pela concatenação de todas as características obtidas. Os resultados utilizando o classificador SVM demostram que aplicação de SAE atinge os melhores resultados em comparação com os demais métodos. As acurácias apresentadas foram de 92,16%, 90,50% e 91,82% para os classificadores SVM com SAE, DAE e PCA, respectivamente.

Semelhante ao trabalho anterior, Kolosnjaji & Eckert (2015) apresentam um método baseado em Auto-Encoder para extrair características automaticamente. Neste trabalho, dois classificadores são utilizados: Rede Neural com quatro camadas associada à técnica Dropout e *Random Forest*. Os experimentos foram realizados nas bases de dados WISDM e UCI HAR, e o procedimento de avaliação foi baseada na técnica de validação cruzada *leave-one-subject-out*. Isto é, os dados de um usuário são utilizados para compor a partição de teste e os dados dos demais usuários da base de dados para compor a partição de treino. Os resultados mostram desempenho equivalentes entre os dois classificadores, com destaque para a rede neural que obteve acurácia de 85,35% e 76,26% para as bases de dados WISDM e UCI HAR, respectivamente.

Seto et al. (2015) apresentam um método para classificar atividades humanas baseado no algoritmo Dynamic Time Warping (DTW), tipicamente utilizado na área de classificação de séries temporais. O método gera uma representação dos dados dos sensores por meio de um algoritmo modificado, chamado DTWsubseq. Este algoritmo é baseado na comparação de subsequências (Negi & Bansal, 2005) e aplica um procedimento de agrupamento hierárquico (*clustering*) para obter protótipos de comparação (*templates*). Esses protótipos são utilizados para classificação baseada no cálculo da distância, similar aos trabalhos de Siirtola et al. (2011) e Terzi et al. (2017). Experimentos foram realizados em duas bases de dados, uma sintética e uma base real (UCI HAR). Os resultados mostram que o método proposto tem um bom desempenho de classificação, com acurácia de 89% para a base de dados real e 70% para base sintética.

Alsheikh et al. (2016) apresentam um método baseado em *Deep Belief Networks* (DBNs). DBNs são redes neurais compostas de múltiplas camadas ocultas. Essas camadas são formadas por máquinas restritas de Boltzmann (RMBs). Experimentos com esse método foram conduzidos na base de dados WISDM e com estratégia de avaliação *10-fold cross-validation*. Previamente, os dados dos sinais foram pré-processados utilizando a transformada rápida de Fourier. Os resultados demostram um bom desempenho de classificação, com acurácia de 98,23%. Esse resultados evidenciam uma superioridade com relação aos trabalhos de Kwapisz et al. (2011) e Catal et al. (2015), apresentados no grupo dos trabalhos baseados na extração manual de características (HF).

Ronao & Cho (2016) apresentam um método baseado em redes neurais profundas de convolução (CNN) para extrair automaticamente características dos dados brutos dos sensores. O método é aplicado aos dados originais sem a necessidade de etapas de pré-processamento ou extração manual de características. Experimentos foram realizados na base de dados UCI HAR. Os resultados demostram a eficácia do método com acurácia de 94,79%. Além disso, experimentos com adição de informações do sinal

no domínio da frequência (pré-processamento com a transformada rápida de Fourier) mostram um incremento da acurácia para 95,75%.

Ignatov (2017) apresenta, também, um método baseado em CNN. O método tem como estratégia utilizar a CNN para extrair automaticamente características dos dados do acelerômetro. Essas características são combinadas com características selecionadas manualmente (e.g., média, variância) para formar uma representação de alto nível, e consequentemente, utilizar a representação para o problema RAH. Experimentos foram realizados para as bases de dados WISDM e UCI HAR. Os resultados demostram um desempenho de classificação superior aos apresentados por Catal et al. (2015) e Anguita et al. (2013b), com 93,42% e 97,63% de acurácia respectivamente. Além disso, uma avaliação do método com e sem a combinação de características HF demostrou que a utilização de características estatísticas incrementou a acurácia em apenas 2,32% na base de dados UCI HAR.

3.1.3 Discussão

As Tabelas 3.1 e 3.2 apresentam os trabalhos relacionados agrupados de acordo com a base de dados utilizada nos experimentos, tipo de avaliação e desempenho obtido.

Tabela 3.1. Sumário dos trabalhos baseados em HF.

Referência	Métodos	Base de dados	Avaliações	Acurácias (%)
Siirtola et al. (2011)	SAXS + KNN	-	10-fold cv	86,7
Kwapisz et al. (2011)	HF + MLP	WISDM	10-fold cv	91,7
Catal et al. (2015)	HF + Ensemble Vote	WISDM	10-fold cv	94,3
Anguita et al. (2013b)	HF + SVM	UCI HAR	21-9	91,7
Ronao & Cho (2017)	HF + HMM	UCI HAR	21-9	93,18
Walse et al. (2017)	HF + Adaboost com RF	UCI HAR	21-9	98,06
Terzi et al. (2017)	SAX + distância média	UCI HAR	3 usuários e 2 atividades	97-99
(Micucci et al., 2017)	Raw + RF	UniMib SHAR	5-fold cv	88,41
(Micucci et al., 2017)	Raw + RF	UniMib SHAR	leave-one-subject-out	73,17

Tabela 3.2. Sumário dos trabalhos baseados em NHF.

Referência	Métodos	Base de dados	Avaliações	Acurácias (%)
Kolosnjaji & Eckert (2015)	Auto-Encoder	WISDM	leave-one-subject-out	85,35
Ignatov (2017)	CNN + HF	WISDM	26-10	93,42
Alsheikh et al. (2016)	FFT + DBN	WISDM	10-fold cv	98,2
*	MBOSS VS	WISDM	10-fold cv	**94,68**
*	MBOSS VS	WISDM	leave-one-subject-out	**83,35**
Kolosnjaji & Eckert (2015)	Auto-Encoder	UCI	leave-one-subject-out	76,26
Seto et al. (2015)	DTWsubseq	UCI HAR	21-9	89,0
Ronaoo & Cho (2015)	CNN	UCI HAR	21-9	90,89
Li et al. (2014)	PCA + SVM	UCI HAR	21-9	91,82
Li et al. (2014)	Autoencoders + SVM	UCI HAR	21-9	92,16
Ronao & Cho (2016)	CNN	UCI HAR	21-9	94,79
Ronao & Cho (2016)	FFT + CNN	UCI HAR	21-9	95,75
Ignatov (2017)	CNN	UCI HAR	21-9	96,37
Ignatov (2017)	CNN + HF	UCI HAR	21-9	97,63
*	MBOSS VS	UCI HAR	10-fold cv	**98,85**
*	MBOSS VS	UCI HAR	leave-one-subject-out	**72,81**
*	MBOSS VS	UniMib SHAR	10-fold cv	**99,38**
*	MBOSS VS	UniMib SHAR	leave-one-subject-out	**79,60**

A partir dos dados das tabelas[1], é possível verificar que três bases de dados, WISDM, UCI HAR e UniMib SHAR, são utilizadas frequentemente para avaliar os métodos propostos por esses trabalhos relacionados. Uma relação entre base de dados e modelo de avaliação é observada. Para a base WISDM é comum uma avaliação *10-fold cross-validation*, para UCI HAR é comum um particionamento 21-9 e para UniMiB SHAR 5-fold cv e leave-one-subject-out.

[1] Valores com a referência (*) pertencem aos resultados obtidos por este trabalho e são apresentados antecipadamente com objetivo de comparação.

Com relação aos resultados experimentais dos trabalhos, três principais pontos se destacam. O primeiro, os métodos HF e NHF, em geral, são comparáveis em termos de acurácia, mas diferentes estrategias de avaliação e base de dados dificultam um avaliação justa entre os classificadores. O segundo é que grande parte dos trabalhos NHF que atinge altas taxas de acurácia implementa algum tipo de rede neural profunda cuja aplicação em plataformas com recursos limitados é desfavorável. Por último, os trabalhos de RAH, em geral, buscam obter os maiores desempenhos de classificação e não dão uma atenção a realização de uma avaliação aprofundada, na qual os métodos são submetidos a diferentes bases de dados e avaliados para diferentes configurações de aplicação, o que pode omitir problemas de generalização. Dentre os trabalhos categorizados, somente os trabalhos de Kolosnjaji & Eckert (2015) e Ignatov (2017) aprofundam a investigação para mais de uma base de dados de atividades.

Com relação a pontos de melhoria e novas contribuições, observamos dois fatos. O primeiro é que são poucos os trabalhos que exploram a utilização das características no domínio discreto. Além disso, os resultados alcançados por esses trabalhos até o momento não são suficientes em termos de desempenho de classificação. Logo, uma melhoria é oportuna para contribuir com o conhecimento da área. O segundo fato, trata-se de investigar técnicas e algoritmos de classificação de séries temporais. De acordo com os trabalhos de Seto et al. (2015) e Terzi et al. (2017), aplicar esse tipo de algoritmo sugere uma alternativa oportuna a abordagem de extração manual de características. Portanto, o uso das características no domínio discreto combinado com as tecnicas de classificação de séries temporais são as contribuições exploradas neste trabalho.

3.2 Considerações Finais

Ao longo deste capítulo foi apresentado a revisão da literatura executada para analisar e avaliar os trabalhos relacionados na área de RAH. Os trabalhos foram categorizados e descritos segundo a abordagem que as características são extraídas. Ao final, uma síntese dos resultados experimentos dos trabalhos foi apresentado, juntamente, com os resultados deste trabalho.

Por fim, a análise dos trabalhos possibilitou identificar pontos de melhoria e direções oportunas para contribuir com o conhecimento da área. Essas direções sugerem explorar o uso de características no domínio discreto e na investigação de técnicas de análise de séries temporais, assunto do próximo capítulo.

Capítulo 4

Representação Simbólica de Séries Temporais

Este capítulo apresenta como as características no domínio discreto podem ser extraídas com a aplicação de técnicas de análise e classificação de séries temporais. Ao longo deste capítulo os seguintes métodos são apresentados: *Symbolic Fourier Approximation* (SFA), *Bag-Of-SFA-Symbols* (BOSS) e *Bag-Of-SFA-Symbols Vector Space* (BOSS-VS). Esses métodos de representação simbólica de séries temporais são base para o desenvolvimento deste trabalho.

4.1 Motivos para Utilizar o Domínio Discreto

Um importante aspecto das características no domínio discreto é o processo de transformação (discretização) aplicado sobre os dados brutos. Esse processo é capaz de compactar substancialmente os dados de tal maneira, que traz ganhos no processamento e armazenamento dos sistemas que as utilizam. Apesar da potencial perda de informação imputada por esse processo, um conjunto de símbolos limitado e tamanho de palavra podem ser definidos para estabelecer o nível de compressão desejado, dando flexibilidade às aplicações.

Uma vez que os dados brutos são transformados em palavras por meio da discretização, as características podem ser calculadas com técnicas de comparação exata ou aproximada de palavras e métricas de distância ou similaridade (Figo et al., 2010). Logo, as características obtidas podem ser utilizadas para encontrar padrões conhecidos nos dados e classificar atividades humanas.

Como apontado na Seção 3.1.3, poucos são os trabalhos para RAH que se beneficiam das características no domínio discreto. Além de que os trabalhos existentes

executam uma abordagem manual e exploratória para determinar quais características devem ser extraídas dos dados. Dessa maneira, definir um método de discretização, com suas especificidades, e os cálculos necessários para obter as características, e posteriormente, avaliá-los para RAH, torna a adesão desta abordagem complexa e custosa em comparação às demais, como as características do domínio do tempo e frequência.

Entretanto, trabalhos na área de classificação de séries temporais têm apresentado inovações com a manipulação dos dados discretizados. Essas inovações consistem de um processamento automatizado de extração de características e baseado nas técnicas de recuperação de informação em documentos de texto.

4.2 Representação de Séries Temporais

A literatura apresenta diferentes modos de representação de séries temporais, como mostrado na Figura 4.1. Essas são utilizadas com o propósito de reduzir a quantidade de dados a tamanhos manuseáveis e, também, preservar as informações mais relevantes após o processo de redução.

Os métodos de representação podem ser divididos em duas categorias: numéri-

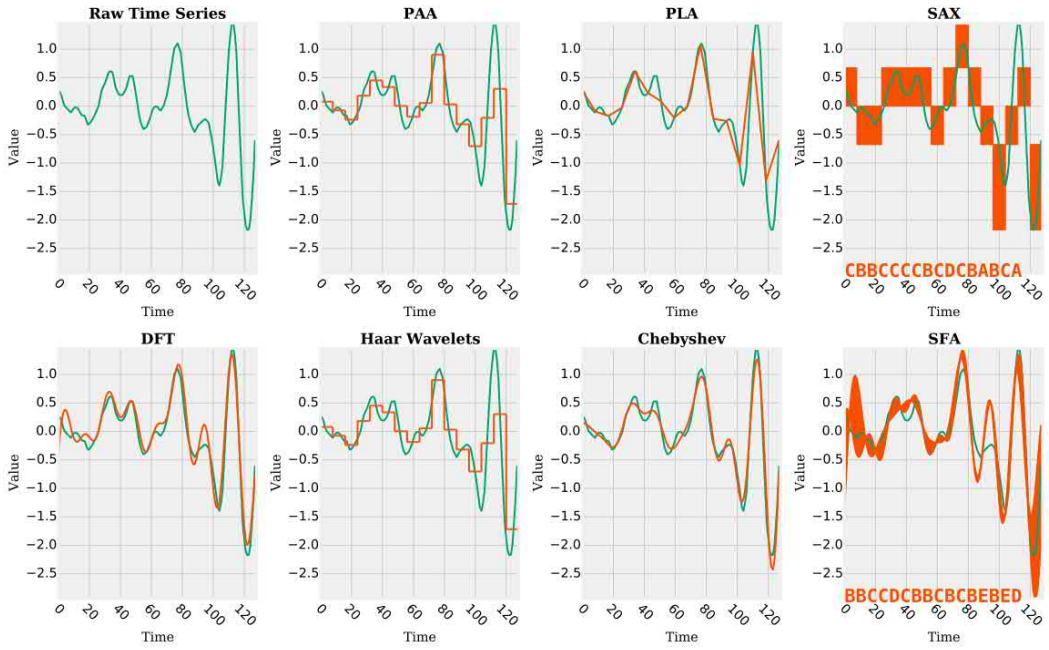


Figura 4.1. Sete exemplos de representação de séries temporais: PAA (Piecewise Aggregate Approximation), PLA (Piecewise Linear Approximation), SAX (Symbolic Aggregation approXimation), DFT (Discrete Fourier Transform), Haar Wavelets, Chebyshev e SFA. Fonte: Adaptada de Schäfer (2016).

cas ou simbólicas. A representação numérica é uma aproximação (ou também chamada de transformação) dos dados originais em dados numéricos mais representativos e compactos. Exemplos desses métodos são: Transformada Discreta de Fourier (DFT), a Transformada Discreta de Wavelet (DWT) e Piecewise Aggregate Approximation (PAA) (Wilson, 2017).

Por outro lado, a representação simbólica é uma aproximação combinada com um processo de discretização (ou também chamada de quantização). A discretização consiste em mapear os dados numéricos a uma sequência de símbolos. Os símbolos representam intervalos numéricos e são utilizados para compactar as informações originais. Exemplos desses métodos são: *Symbolic Aggregation approXimation* (SAX)(Lin et al., 2007) e o *Symbolic Fourier Approximation* (SFA) (Schäfer & Högqvist, 2012).

Dentre os métodos de representação simbólica, este trabalho avalia a utilização do SFA. O SFA é um método mais recente que o SAX e tem apresentado resultados superiores para a tarefa de classificação de séries temporais (Schäfer, 2016). Além disso, após a revisão da literatura, conclui-se que o SFA não foi avaliado em nenhuma solução baseado em sensores inerciais para RAH.

4.3 SFA: Symbolic Fourier Approximation

A representação simbólica obtida pelo SFA se constitui em duas fases: aproximação e discretização. A aproximação consiste na aplicação da DFT sobre a série temporal. O resultado da DFT são os coeficientes reais e imaginários que representam o sinal no domínio da frequência. A aproximação é feita com utilização dos $\frac{l}{2}$ primeiros coeficientes, sendo l o tamanho definido para a palavra SFA. Schäfer & Högqvist (2012) mostram que essa aproximação corresponde a aplicação de um filtro passa baixa com uma frequência de corte definida a partir do tamanho l, e que o filtro é útil para eliminar ruídos nos sinais.

A discretização consiste na aplicação de uma função para mapear um intervalo de valores numéricos para valores discretos (símbolos). A função de mapeamento utilizada pelo SFA é o *Multiple Coefficient Binning* (MCB). O MCB é construído a partir dos dados de várias amostras, e os detalhes da construção são apresentados nas próximas seções. De acordo com Schäfer & Högqvist (2012), o MCB minimiza a perda de informação introduzida no processo de discretização, e o tamanho do conjunto de símbolos define a compactação dos dados originais.

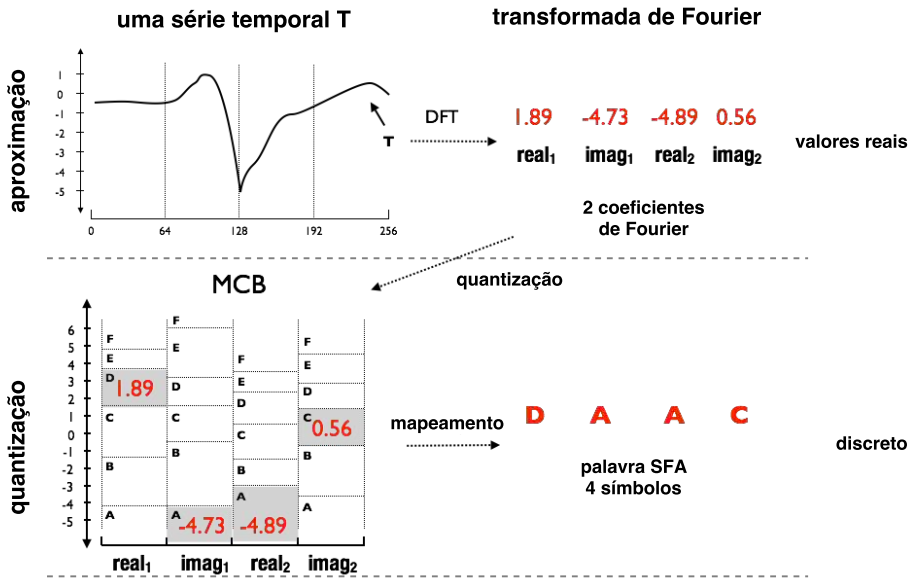


Figura 4.2. Transformação SFA: Uma série temporal T é aproximada usando a DFT e discretizada usando MCB resultando em uma palavra SFA DAAC. Fonte: Adaptado de Schäfer & Högqvist (2012).

A Figura 4.2 exemplifica a aplicação do método SFA para uma série temporal de tamanho $n = 256$, tamanho de palavra $l = 4$ e tamanho do conjunto de símbolos $c = 6$. Neste método uma série temporal é representada por uma sequência de valores discretos (símbolos). Esses valores são na prática, por exemplo, caracteres ou letras de uma palavra, e eles podem ser utilizados em estruturas ou algoritmos de mineração de dados como árvores, *hashing*, modelos de Markov e *matching* de palavras (Schäfer, 2016).

4.3.1 Aproximação

A DFT (Phillips et al., 2013) é um algoritmo que decompõe um sinal T de tamanho n em uma soma de funções ortogonais básicas, por exemplo, ondas senoidais. Cada onda é representada por um número complexo $X_u = (real_u, imag_u)$, para $u = 0, \ldots, n-1$, chamados de coeficientes de Fourier. O n-valor de Fourier de uma série temporal $T(x) = 0, \ldots, n-1$ é dado pela equação:

$$DFT(T) = X_0 \ldots X_{n-1} = (real_0, imag_0, \ldots real_{n-1}, imag_{n-1}), \tag{4.1}$$

com

$$X_u = \frac{1}{n} \sum_{x=0}^{n-1} T(x) \cdot e^{-j2\pi ux/n}, \text{para } u \in [0, n), j = \sqrt{-1}. \tag{4.2}$$

Os primeiros coeficientes de Fourier estão relacionados aos intervalos de frequência mais baixos ou as seções que se alteram lentamente de um sinal. Os coeficientes de ordem mais alta se correlacionam com faixas de frequência mais altas ou seções de mudança rápida de um sinal. Para a aproximação são utilizados os primeiros coeficientes de Fourier. Esses são suficientes para descrever de forma aproximada grande parte dos sinais. A aproximação aplicada é equivalente à aplicação de um filtro tipo passa-baixa para suavização e remoção dos ruídos presentes nos intervalos de frequências mais altas do sinal. O primeiro coeficiente de Fourier $X_0 = \frac{1}{n}\sum_{x=0}^{n-1} T(x) \cdot e^0$ é igual ao valor médio do sinal e pode ser descartado para obter invariância de deslocamento (desvios verticais).

4.3.2 Discretização

Uma série temporal é transformada em uma palavra SFA com aplicação da DFT e uma tabela de pesquisa com intervalos de discretização MCB pré-calculados. Para entender esse processo, esta seção apresenta algumas definições.

Intervalos da discretização (bins): um a-intervalo de discretização ($QI(a)$) é definido por um ponto de quebra (breakpoint) superior $\beta(a)$, um inferior $\beta(a-1)$ e um $a\text{-}symbol_a$ de um alfabeto Σ:

$$QI(a) = [\beta(a-1), \beta(a)) \triangleq symbol_a \in \Sigma. \tag{4.3}$$

A quantidade de bins corresponde à quantidade de valores utilizados na DFT. Um índice j é usado para indicar os correspondentes j-valores da Transformada de Fourier $DFT(T) = t'_0, \ldots, t'_j, \ldots, t'_{l-1}$:

$$QI(a) = [\beta(a-1), \beta(a)) \triangleq symbol_a, j \in [1 \ldots c), |\Sigma| = c. \tag{4.4}$$

Por um momento, é omitido a forma que os intervalos são obtidos (próxima subseção) e se assume que a discretização MCB retorna l conjuntos de c intervalos de quantização para uma palavra SFA. Esses intervalos são apresentados pelo MCB:

$$MCB = \begin{pmatrix} -\infty & -\infty & \ldots & -\infty & -\infty \\ \beta_0(1) & \beta_1(1) & \ldots & \beta_{l-2}(1) & \beta_{l-1}(1) \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \beta_0(c-1) & \beta_1(c-1) & \ldots & \beta_{l-2}(c-1) & \beta_{l-1}(c-1) \\ +\infty & +\infty & \ldots & +\infty & +\infty \end{pmatrix}. \tag{4.5}$$

A tabela de pesquisa mapeia os primeiros $\frac{l}{2}$ coeficientes de Fourier ($\frac{l}{2}$ valores reais e $\frac{l}{2}$ valores imaginários).

Palavra SFA: A representação simbólica $SFA(T) = s_0, ..., s_t - 1$ da série temporal T com aproximação $DFT(T) = real_0, imag_0, \ldots, real_{\frac{l}{2}-1}, imag_{\frac{l}{2}-1} = t'_0, \ldots, t'_{l-1}$ é T mapeada de um número real a um símbolo a partir de um alfabeto $\Sigma = \{symbol_1, \ldots, symbol_c\}$ de tamanho c, $SFA : \mathbb{R}^l \to \Sigma^l$. Especificamente, um j-valor numérico $t'_j \in DFT(T)$ é mapeado para um a-símbolo de um alfabeto Σ, se o valor corresponde ao intervalo $QI_j(c) = [\beta_j(a-1), \beta_j(a))$:

$$s_j = \left\{ symbol_a, se \left(\beta_j(a-1) \leq t'_j < \beta_j(a) \right) \right\}, \text{para} j \in [0 \ldots l) \,. \tag{4.6}$$

No canto inferior direito da Figura 4.2 são ilustrados o mapeamento e os intervalos de discretização MCB. Como resultado, uma palavra SFA igual a "DAAC" é obtida da $DFT(T) = (1.89, -4.73, -4.89, 0.56)$.

4.3.3 MCB: Multiple Coefficient Binning

O objetivo do MCB é minimizar a perda de informação introduzida pelo processo de discretização. Os intervalos MCB são calculados a partir de uma matriz ordenada de coeficientes de Fourier obtidos das séries temporais. Os dados de uma coluna da matriz são utilizados para definir os pontos de quebra dos valores numéricos, e as linhas da matriz são os primeiros coeficientes reais e imaginários das séries temporais.

Para entender esse processo, esta seção apresenta definições.

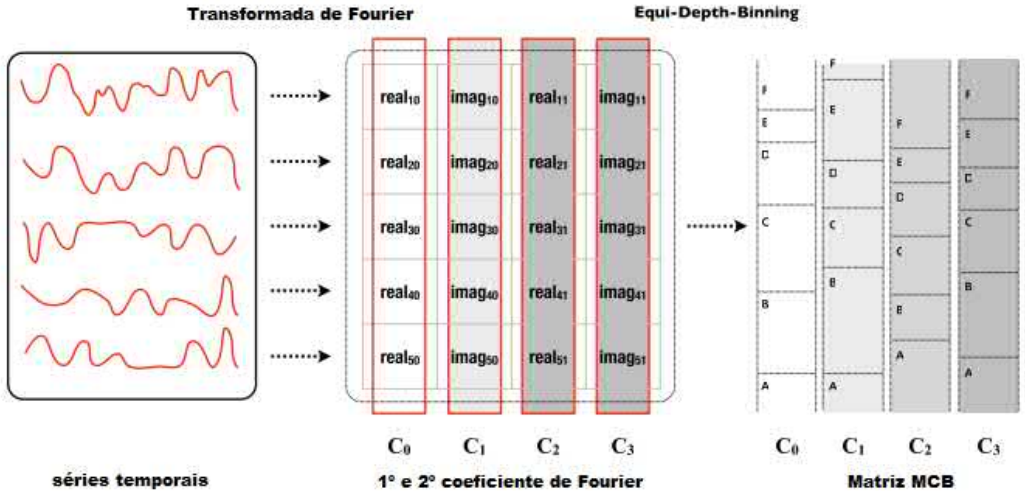


Figura 4.3. Exemplo da aplicação do MCB para cada i-colunas de distribuição. Fonte: Schäfer (2016).

Matriz MCB: A matriz $A = (a_{ij})_{i=1,\dots,N;j=1,\dots,l}$ é construída a partir dos dados de N séries temporais de treino ($T_i ei \in [1 \dots N]$) aproximadas com apenas os $\frac{l}{2}$ coeficientes de Fourier - igual a uma palavra SFA de tamanho l com $\frac{l}{2}$ valores reais e $\frac{l}{2}$ imaginários. A i-linha da matriz A corresponde a transformada de Fourier da i-série temporal T_i (ver Figura 4.3):

$$A = \begin{pmatrix} DFT(T_1) \\ . \\ DFT(T_i) \\ . \\ DFT(T_N) \end{pmatrix} = \begin{pmatrix} real_{1;0} & imag_{1;0} & \dots & real_{1;\frac{l}{2}-1} & imag_{1;\frac{l}{2}-1} \\ . & . & \dots & . & . \\ real_{i;0} & imag_{i;0} & \dots & real_{i;\frac{l}{2}-1} & imag_{i;\frac{l}{2}-1} \\ . & . & \dots & . & . \\ real_{N;0} & imag_{N;0} & \dots & real_{N\frac{l}{2}-1} & imag_{N\frac{l}{2}-1} \end{pmatrix} \tag{4.7}$$

$$= \begin{pmatrix} C_0 & C_1 & \dots & C_{l-2} & C_{l-1} \end{pmatrix}.$$

A j-coluna Cj corresponde a quaisquer valores reais ou imaginários de todas as N séries temporais. Cada coluna é ordenada de acordo com seus valores, e em seguida, a coluna é particionada em c partições. As partições seguem o formado *equi-depth binning*, na qual a quantidade total de valores em um intervalo $\left[\beta_j(a-1) \leq t'_j < \beta_j(a)\right)$ é igual a qualquer outro intervalo.

4.3.4 Complexidade computacional do SFA

A transformação simbólica aplicada pelo SFA tem complexidade computacional dominada pela transformada discreta de Fourier, que pode ser implementada utilizando a transformada rápida de Fourier (Cochran et al., 1967), e portanto, apresenta complexidade computacional igual a:

$$T(SFA) = O(n \log n), \tag{4.8}$$

sendo n o tamanho da série temporal.

4.4 BOSS: Bag-Of-SFA-Symbols

O método BOSS foi desenvolvido baseado na representação simbólica SFA para diferentes tarefas como classificação e agrupamento de séries temporais (Schäfer, 2016). Este método apresenta a estratégia de construir histogramas (distribuição de frequência) utilizando as palavras SFA obtidas de uma série temporal.

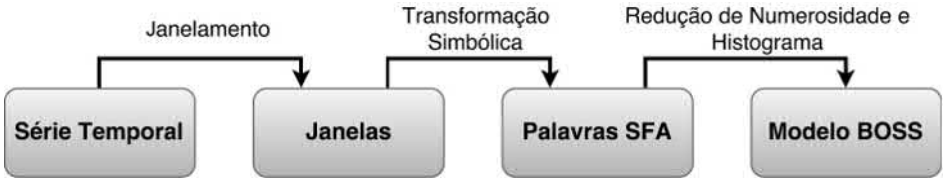


Figura 4.4. Fluxo de tarefas para construção do modelo BOSS. Fonte: Adaptada de Schäfer (2016).

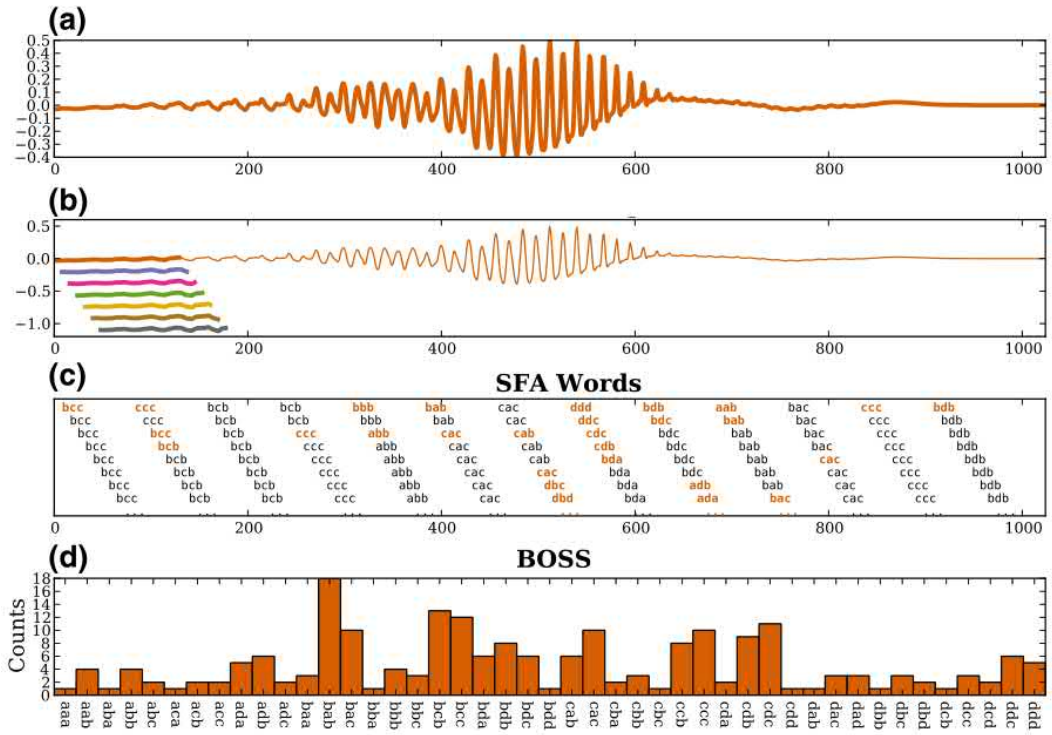


Figura 4.5. Exemplo da construção do histograma BOSS. (a) uma série temporal, (b) processo de janelamento, (c) transformação simbólica para obter as palavras SFA e (d) construção do histograma. Fonte: Adaptado de Schäfer (2016).

Em síntese, essa estratégia constrói uma nova representação a partir de um conjunto de janelas de dados extraídas da série temporal. Essas janelas de tamanho w são transformadas em palavras SFA de tamanho l e alfabeto de tamanho c. Logo, essas palavras SFA são organizadas em suas distribuições de frequência, representadas por histogramas. Esses são chamados de modelo BOSS. Essa nova representação é obtida seguindo um processo de 3 tarefas: janelamento, transformação simbólica e construção do histograma.

As Figuras 4.4 e 4.5 mostram as etapas do processo de criação do modelo BOSS para uma série temporal de tamanho n previamente normalizada (média zero e desvio padrão igual a um). Primeiramente, as janelas são obtidas aplicando um procedimento

de janelamento de tamanho w ao longo da série temporal (Figura 4.5b). Ao total são obtidos $(n - w + 1)$ janelas. Em seguida, cada janela extraída é submetida a transformação simbólica utilizando o algoritmo SFA (Figura 4.5c). Como cada janela corresponde a uma palavra SFA, ao total são obtidas $(n-w+1)$. Por fim é construído o histograma das palavras SFA (Figura 4.5d). Neste ponto, um procedimento de redução de numerosidade é aplicado para reduzir o efeito de repetição de palavras em janelas vizinhas, que ocorre frequentemente em sinais estáveis. Portando, a contagem de uma palavra SFA ocorre na primeira aparição e as próximas repetições não são contabilizadas no histograma até que uma mudança de palavra ocorra (Figura 4.5c).

Segundo Schäfer (2016), os histogramas são úteis para discriminar classes de séries temporais. A Figura 4.6 apresenta um exemplo de um problema de classificação com duas classes e seus respectivos modelos BOSS. Observa-se que diferentes padrões no sinal caracterizam a classe à qual ele pertence. Logo, com transformação para modelo BOSS, esses padrões são extraídos e evidenciados ao longo do histograma. Observando a Figura 4.6, é possível classificar as séries temporais verificando unicamente se a palavra "baa" surge no histograma (classe "Anormal") ou não surge no histograma (classe "Normal").

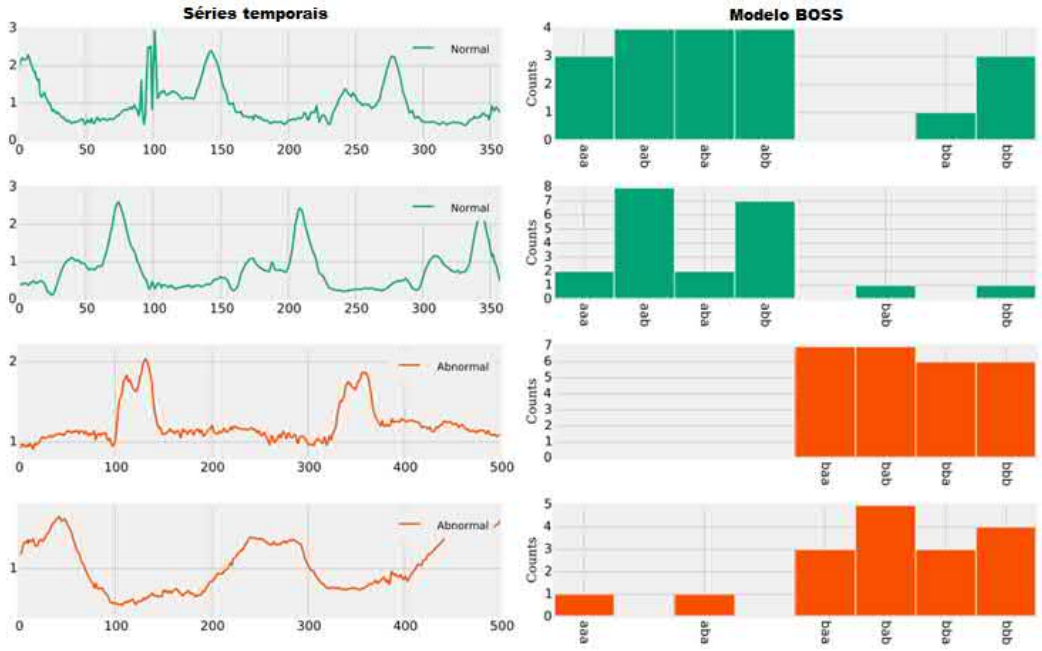


Figura 4.6. Exemplos de modelos BOSS para um problema de classificação de duas classes: movimentos de andar normal (verde) e anormal (laranja). Fonte: Schäfer (2016).

4.4.1 Complexidade Computacional do BOSS

O método BOSS é baseado no SFA, logo a complexidade desse algoritmo é dominante. Entretanto, dada a função de janelamento, Schäfer (2016) propõem a substituição da DFT pelo método *Momentary Fourier Transform* (MFT) para obter uma otimização do método.

A MFT é um caso específico da transformada Fourier, na qual a propriedade recursiva é aplicada por meio da função de janelamento. Dessa maneira, a complexidade computacional do BOSS é linear n, onde existe $n - w + 1$ janelas de tamanho w em uma série temporal de tamanho n. Detalhes da otimização podem ser consultados nos trabalhos de Schäfer (2016). Portando, a complexidade computacional do BOSS é:

$$T(BOSS) = O(n + w \log w), \tag{4.9}$$

sendo n o tamanho da série temporal e w o tamanho da janela.

4.5 BOSS VS: Bag-Of-SFA-Symbols in Vector Space

O modelo BOSS VS é uma combinação do método BOSS e do modelo de espaço vetorial para classificação de séries temporais em grande escala . Isto é, o modelo BOSS-VS estende o modelo BOSS construindo um histograma compacto e generalizado, na qual representa uma classe de séries temporais. Esse processo reduz significativamente a complexidade computacional do modelo de classificação, resultando em tempos menores para a tarefa de classificação.

O modelo espaço vetorial foi apresentado pela comunidade de recuperação de informação para representar documentos de textos através de um vetor de termos (palavras-chave). Esses vetores são utilizados para inferir a similaridade de um determinado texto a uma classe de documentos. Além disso, Salton et al. (1975) propuseram, pela primeira vez, a técnica term-frequency inverse document frequency (TF-IDF), na qual pesos são atribuídos para cada termo no vetor baseado na frequência dele em relação ao documento. Isso proporciona que pesos maiores sejam atribuídos aos termos mais representativos, ao passo que os termos irrelevantes da classe recebem pesos menores.

A Figura 4.7 apresenta o modelo BOSS-VS. Primeiramente, cada série temporal é transformada para o histograma BOSS. Em seguida, a matriz de vetores *tf-idf* é computada utilizando todos os histogramas BOSS. A matriz contém um vetor tf-idf

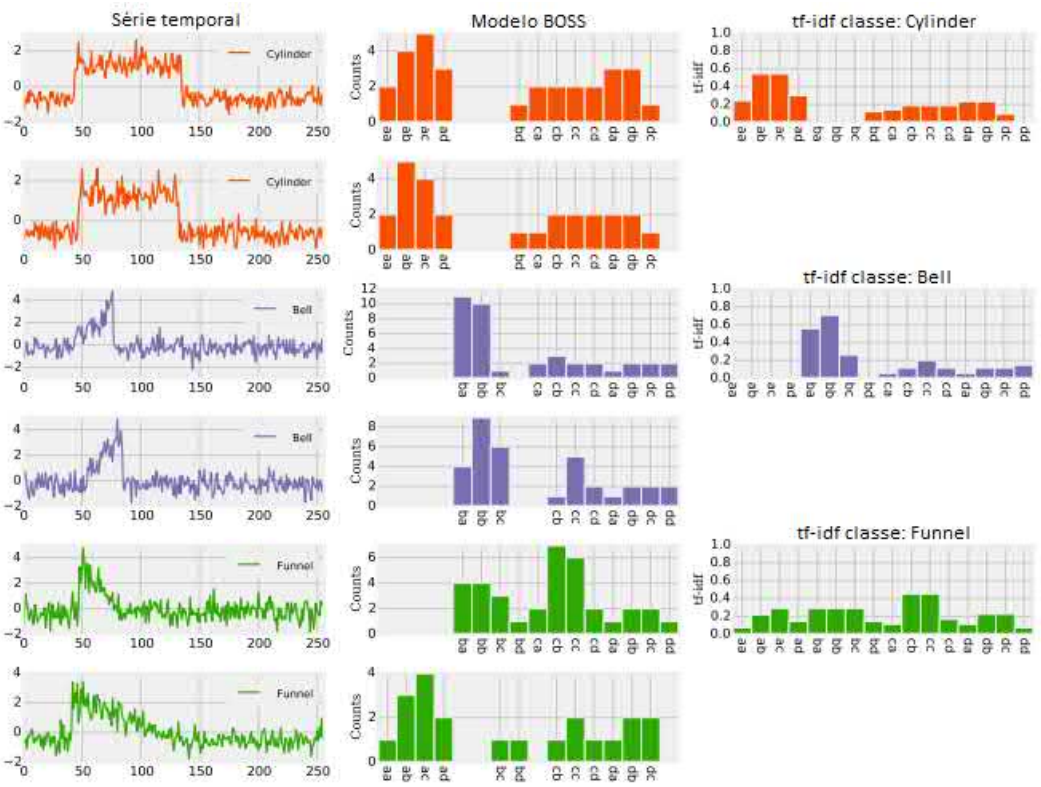


Figura 4.7. Exemplo do modelo BOSS VS para três classes: Cylinder (laranja), Bell (roxo) e Funnel (verde). Fonte: Schäfer (2016).

para cada classe do problema. Essa nova representação proporciona algumas vantagens como velocidade na classificação e reduções de informações irrelevantes. A velocidade é obtida devido ao *tf-idf* que gera uma representação compacta da classe ao todo reduzindo significativamente o custo computacional para classificação. A redução de ruído é obtida em razão dos pesos atribuídos pelo *tf-idf*, na qual palavras SFA presentes em todas as classes do problema recebem pesos baixos de relevância.

O método utiliza a abordagem descrita por Senin & Malinchik (2013), na qual é calculado o *idf* (inverse document frequency) para cada classe e não para uma série temporal individual. O tf (term frequency) para uma palavra SFA p de uma série temporal T é dado pela seguinte equação:

$$tf(p,T) = \begin{cases} 1 + \log(B_T(p)) & \text{se } B_T(p) > 0 \\ 0 & \text{senão} \end{cases}, \tag{4.10}$$

sendo $B_T(p)$ o histograma BOSS, que representa a frequência de uma palavra SFA em uma série temporal específica. Logo, o *tf* de uma palavra SFA p em uma classe C é

dado pela seguinte equação:

$$tf(p, C) = \begin{cases} 1 + \log(\Sigma_{T \in C} B_T(p)) & \text{se } \Sigma_{T \in C} B_T(p) > 0 \\ 0 & \text{senão} \end{cases}. \tag{4.11}$$

Como dito anteriormente, a *idf* atribui os pesos de relevância às palavras SFA obtidas da série temporal T de uma classe C e é calculada pela seguinte equação:

$$idf(p, C) = \log \frac{|C|}{\underbrace{|\{C \mid T \in C \wedge B_T(p) > 0\}|}_{\text{numero de classes que contem p}}}. \tag{4.12}$$

O *idf* de uma palavra SFA representa o total de classes dividido pelo número de classes que esta palavra ocorre na classe C. Um valor alto de *idf* é obtido quando a palavra SFA ocorre em uma específica classe, assim significando que é um termo relevante para a discriminação entre classes.

Por fim, o *tf-idf* de uma palavra SFA em uma classe C é calculada pela seguinte equação:

$$tf\text{-}idf(t, C) = tf(t, C) \cdot idf(t, C). \tag{4.13}$$

Valores de pesos *tf-idf* altos são obtidos por palavras SFA muito frequentes, ao mesmo tempo, que ocorrem em uma específica classe. Logo, palavras SFA comuns a todas as classes recebem pesos baixo, consequentemente, são filtrados da representação.

4.5.1 Classificação com o BOSS VS

Figura 4.8. Classificação utilizando 1-NN e o modelo BOSS-VS. Fonte: Schäfer (2016).

A classificação descreve a tarefa de inferir um rótulo para uma série temporal desconhecida (sem rótulo). A Figura 4.8 apresenta um exemplo da tarefa de classificação. Essa requer que os vetores *tf-idf* sejam computados para cada classe do problema. Primeiramente, uma série temporal desconhecida, neste caso chamada de *query* Q, é transformada a um vetor tf utilizando o modelo BOSS com parâmetros estimados (tamanho de janela, tamanho de palavra e alfabeto). Em seguida, a similaridade do cosseno é calculada utilizando a matriz de pesos *tf-idf* para cada classe C. Por fim, o rótulo da *query* Q é inferido atribuindo o rótulo da classe C que maximizou o resultado da similaridade do cosseno. Sendo o cálculo da similaridade do cosseno definido pela seguinte equação:

$$sim(Q, C) = \frac{\vec{Q} \cdot \vec{C}}{\left\|\vec{Q}\right\| \cdot \left\|\vec{C}\right\|} = \frac{\Sigma_{p \in Q}\, tf(p, Q) \cdot tf\text{-}idf(t, C)}{\sqrt{\Sigma_{p \in Q}\, (tf(p, Q))^2}\sqrt{\Sigma_{p \in C}\, (tf\text{-}idf(p, C))^2}}. \tag{4.14}$$

E o rótulo inferido pela seguinte equação:

$$rotulo(Q) = \underset{C \in CLASSES}{\operatorname{argmax}}\ (sim(Q, C)). \tag{4.15}$$

Na prática a tarefa de classificação é implementada utilizando o algoritmo do 1-vizinhos mais próximo (1-NN). A complexidade computacional da classificação dada por uma busca 1-NN ao longo de |CLASSES| classes é:

$$T(\text{BOSS-VS}) = O(|\text{CLASSES}| \cdot n), \tag{4.16}$$

sendo n o tamanho da série temporal.

4.6 Considerações Finais

Este capítulo apresentou os métodos de representação simbólica séries temporais que são utilizados como base para desenvolvimento da solução proposta neste trabalho. Para o desenvolvimento de sistemas RAH, esses métodos são oportunos visto que apresentam vantagens com relação ao processamento de grandes quantidades de dados obtidos de sensores, redução de interferências causadas por ruídos e geração de modelos de classificação eficientes $O(n)$ para aplicação em dispositivos móveis.

O próximo capítulo descreverá detalhadamente a solução proposta por este trabalho e como os métodos de representação simbólica são aplicados para formar um sistema de RAH.

Capítulo 5

MBOSS: Multivariate Bag-Of-SFA-Symbols

Este capítulo apresenta o método de representação simbólica, nomeado MBOSS, concebido para classificar séries temporais com múltiplas variáveis, considerando o desenvolvimento de um sistema para RAH. O método é uma extensão do método BOSS e tem como principal característica a construção de uma representação única das múltiuplas séries temporais mediante a fusão dos histogramas de palavras gerados por cada série temporal univariada de entrada.

5.1 Séries Temporais Multivariadas

Como apresentado no capítulo anterior, os métodos de representação simbólica são projetados para analisar e classificar séries temporais na perspectiva de uma única variável (univariada). Uma vez que soluções para RAH requerem processar múltiplos sinais (Seção 2.3), esses métodos requerem uma adaptação ou modificação na maneira com que os dados de múltiplas fontes sejam processados. Logo, este trabalho descreve e avalia uma solução para viabilizar a aplicação desses métodos.

As séries temporais multivariadas são coleções de séries temporais univariadas que surgem quando múltiplas fontes de dados interconectados capturam amostras ao longo do tempo (Tabachnick & Fidell, 2012). Muito mais que uma combinação de partes individuais, as séries temporais multivariadas devem ser consideradas como um todo, uma vez que as relações e os padrões entre as variáveis aumentam a quantidade de informações sobre o evento observado. Estes são tipicamente produzidos por sensores, como o acelerômetro que captura informações da aceleração a partir de três variáveis (x, y, z), ou por dispositivos e procedimentos com múltiplos sensores, como no exame

de Eletroencefalografia — procedimento que consiste na análise dos sinais elétricos obtidos por múltiplos eletrodos fixados à cabeça para monitorar a atividade elétrica do cérebro — (de Carvalho et al., 2017).

Na literatura, trabalhos com redes de sensores sobre o corpo (do Inglês, *Body Sensor Networks*) têm enfrentado os desafios de manipular séries temporais multivariadas (Gravina et al., 2017). Esses trabalhos capturam informações de inúmeros sensores e aplicam técnicas de fusão de dados em diferentes níveis do processamento. As principais razões dos trabalhos na utilização de fusão de dados são: precisão, confiabilidade e redução da complexidade da aplicação nos aspectos de processamento, comunicação e armazenamento.

De acordo com o problema e a aplicação que se deseja projetar, três abordagens de fusão de dados de sensores podem ser adotadas (Gravina et al., 2017). Estas são:

- **Fusão no nível de dados brutos** (do Inglês, Data-Level fusion): Neste nível em particular, sistemas que abrangem sensores homogêneos coletando informações de um mesmo evento observado podem combinar diretamente os dados no nível mais baixo de abstração (dados brutos). As principais técnicas aplicadas neste nível em grande parte são implementações de algoritmos de pré-processamento como agregação e compressão de dados, remoção de ruídos, calibração e sincronização e procedimentos de extração de características.

- **Fusão no nível de características** (do Inglês, Feature-Level Fusion): Neste nível intermediário, as características extraídas das múltiplas fontes de dados podem ser combinados para criar uma nova representação. As principais técnicas aplicadas neste nível são seleção e *ranking* de características, redução de dimensionalidade e combinação de características.

- **Fusão no nível de decisão** (do Inglês, Decision-Level Fusion): Neste nível mais alto está o processo de decisão (ou seleção de melhor hipótese) baseado em um conjunto de hipóteses geradas por diferentes classificadores. Neste caso, a decisão final é obtida observando as decisões locais de múltiplos sensores e os combinando. As principais técnicas aplicadas neste nível são votação majoritária e suas variantes, e inferência Bayesiana.

Mais detalhes sobre as abordagens e as técnicas aplicadas para cada nível de processamento podem ser obtidas nas publicações de Mangai et al. (2010).

5.2 Representação Simbólica para Classificação de Séries Temporais Multivariadas

Uma etapa de fusão de dados é proposta por este trabalho para possibilitar a classificação de séries temporais multivariadas utilizando os métodos de representação simbólica. Dentre os níveis de fusão descritos anteriormente, a fusão no nível de característica é adequada para processar as representações simbólicas obtidas por cada série temporal de entrada, uma vez que os histogramas de palavras (Seção 4.4) são como vetores de características de uma determinada série temporal de entrada. A Figura 5.1 exibe a visão geral do processo proposto para permitir que os métodos de representação simbólica classifiquem séries temporais multivariadas.

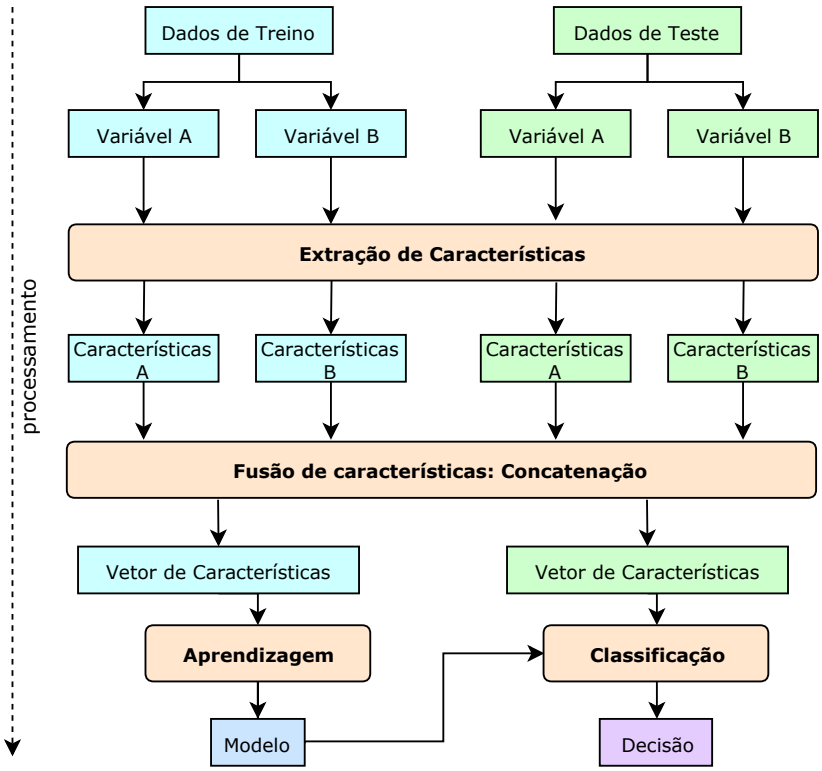


Figura 5.1. Visão geral do processo proposto para classificar séries temporais multivariadas. Fonte: Elaborado pelo Autor.

O processo consiste de basicamente 4 etapas. A primeira consiste em extrair as características dos dados de treino e teste considerando as múltiplas variáveis (no exemplo, Variável A e B). Para isto, os métodos de representação simbólica são aplicados individualmente para cada variável (no exemplo, Característica A e B). A segunda etapa é a fusão de dados a nível de características. Neste caso, a combinação de ca-

racterísticas é realizada aplicando uma função de concatenação. Essa técnica de fusão é simples e não produz nenhum processamento adicional impactante no processo de classificação, visto que esta proposta é projetada para implementação em dispositivos móveis. A terceira etapa consiste na aplicação de algoritmos de aprendizagem para gerar os modelos de classificação. Para isto, utilizaremos a implementação do modelo espaço vetorial (Secão 4.5) adaptado para representação simbólica de séries temporais. Por último, com base no modelo e no vetor de características combinado, a classificação é aplicada para determinar o rótulo da instância de teste.

5.3 MBOSS: Multivariate Bag-Of-SFA-Symbols

De acordo com a proposta de seção anterior, o método de representação simbólica, nomeado Multivariate Bag-Of-SFA-Symbols, é desenvolvido para classificar séries temporais multivariadas. O método é uma extensão do método BOSS (Seção 4.4) cuja implementação aplica a fusão de dados a nível de características.

5.3.1 Extração de Características

A extração de características é executada aplicando o algoritmo de representação simbólica BOSS. Essa tarefa é realizada individualmente para cada variável dos dados de entrada e como resultado são obtidos os histogramas de palavras SFA. Os parâmetros como tamanho de janela w, tamanho de palavra L e tamanho do conjunto de símbolos c serão especificados na Seção 5.5.1.

A Figura 5.2 ilustra um exemplo da representação simbólica de uma série temporal de variável A. No exemplo, a série temporal passa pelas tarefas de janelamento, transformação simbólica com SFA e construção do histograma para obter como resultado sua representação no modelo BOSS.

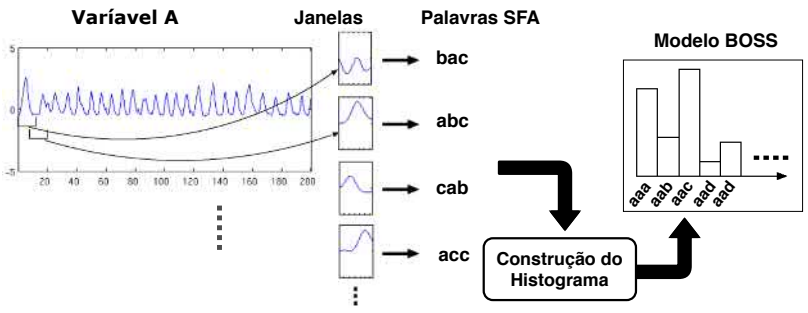


Figura 5.2. Exemplo de uma séries temporal submetida a representação simbólica com o algoritmo do BOSS. Fonte: Elaborado pelo Autor.

5.3.2 Fusão de Múltiplos Histogramas

O modelo MBOSS consiste da combinação dos múltiplos modelos BOSS gerados para cada variável ou série temporal de entrada. A combinação é implementada por uma função de concatenação que combina as características de cada variável e forma um único vetor de características de alta dimensão. Concatenar ou empilhar características (do Inglês *Feature Stacking*) é uma estratégia comum em diversas áreas de pesquisa, um exemplo em recuperação de imagens pode ser encontrado no trabalho de Yu et al. (2013).

A Figura 5.3 exemplifica o processo de fusão de histogramas, na qual a fusão concatena as M características (tamanho do histograma) de dimensão N (número de variáveis) em um único vetor de características de dimensão $M \times N$. Posteriormente, o vetor de caraterísticas final é utilizado para a tarefa de classificação.

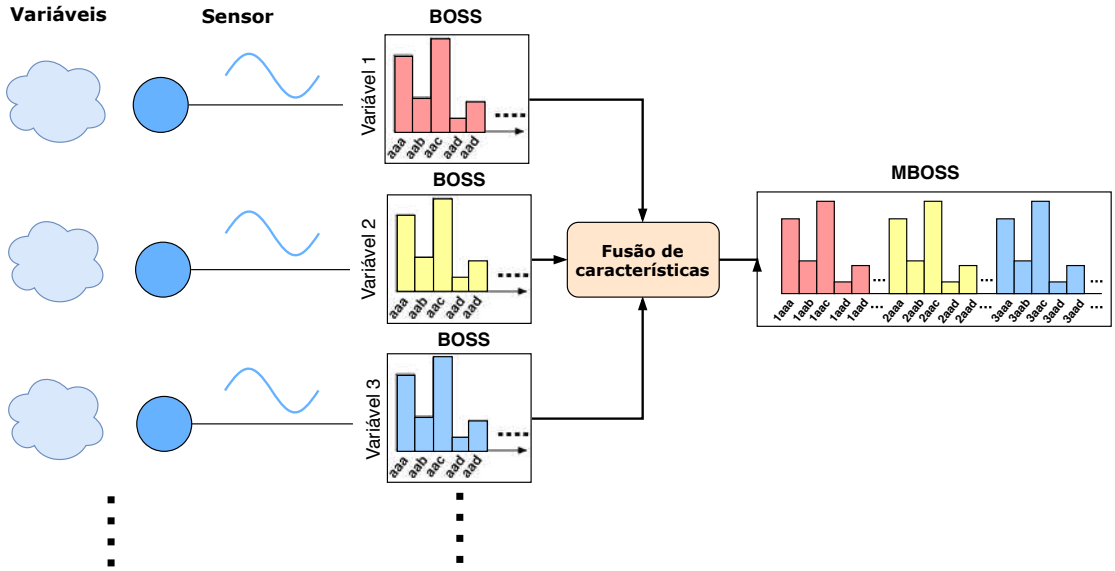


Figura 5.3. Fusão no nível de características: Exemplo da concatenação de histogramas. Fonte: Elaborado pelo Autor.

Na prática, o modelo MBOSS é implementado por meio do Algoritmo 1. O algoritmo considera as séries temporais de entrada como uma única estrutura chamada série temporal multivariada. Essa estrutura é definida como um vetor de séries temporais $s = \{s_1, \ldots, s_i, \ldots, s_v\}$ no qual s_i corresponde a um série temporal de entrada na variável (dimensão) i, e v o tamanho do vetor (linha 1). A saída do algoritmo é definida por uma estrutura que mapeia uma palavra (string) a um valor inteiro (linha 2), chamado de histograma. O algoritmo executa a representação simbólica para cada variável (linha 3) e constrói o histograma a partir das palavras SFA. Para conservar a

Algoritmo 1 Transformação MBOSS.

```
1  function MBOSSTransform( MultivariateTimeSerie s, int v, int w, int l, c)
2      map<String, int > histogram = [], String lastWord = NULL
3      for int i_v in [1..v]
4          for TimeSeries window in sliding_windows(s[i_v], w)  // janelamento
5              String word = toChar(i_v) + SFA(window, l, c)  // palavra SFA
6              if word != lastWord  // redução de numerosidade
7                  histogram[word]++
8              lastword = word
9      return histogram
```

informação do índice da variável da qual a palavra SFA foi obtida, insere-se mais um caractere à palavra SFA (linha 5). Por exemplo, uma palavra "ab" extraída da variável 2 receberá o caractere "2", então, a nova palavra formada será "2ab". Por fim, a redução de numerosidade é aplicada (linhas 6-8), e o resultado do algoritmo é o modelo MBOSS em formato de histograma (linha 8).

Com base na complexidade computacional do BOSS (Seção 4.4), o MBOSS (Algoritmo 1) tem complexidade computacional igual a:

$$T(MBOSS) = v \cdot T(BOSS) \tag{5.1}$$

$$T(MBOSS) = v \cdot O(n)$$

$$T(MBOSS) = O(n), \text{ com } v \ll n,$$

sendo n o tamanho da série temporal, e considerando, que o número de variáveis v é muito menor que o tamanho do segmento, $v \ll n$.

5.4 MBOSS VS: Representação no Modelo Espaço Vetorial

A representação no modelo espaço vetorial consiste de uma etapa adicional de transformação que reduz a quantidade de instâncias MBOSS de uma determinada classe para uma única instância, ou também, chamada de protótipo. Essa transformação é realizada para garantir menores tempos de classificação e gerar representações compactas para aplicação em dispositivos móveis. Os protótipos são construídos utilizando a técnica *term-frequency inverse-document-frequency*(TF-IDF), da mesma forma, ao aplicado no BOSS-VS (Seção 4.5).

A Figura 5.4 ilustra o processo de representação no modelo TF-IDF. No exemplo, N séries temporais multivariadas representadas no modelo MBOSS são utilizadas para gerar um único protótipo (MBOSS VS). Importante que as instâncias utilizadas pertençam a uma única classe do problema. Dessa maneira, a quantidade de protótipos gerados será igual a quantidade de classes do problema.

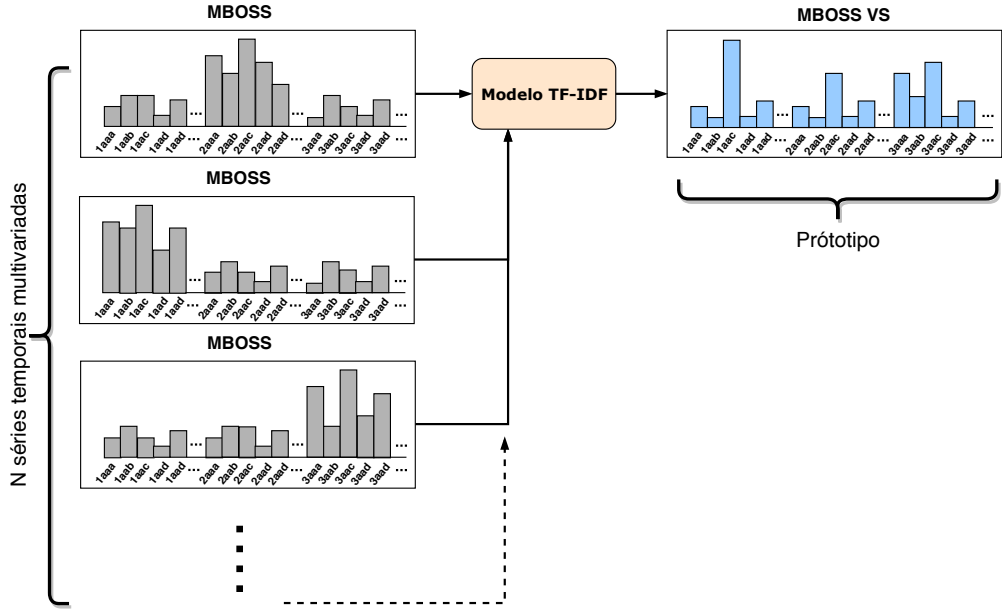


Figura 5.4. Representação de instâncias do modelo MBOSS no modelo *term-frequency inverse-document-frequency*. Fonte: Elaborado pelo Autor.

5.4.1 Classificação Baseada na Comparação de Protótipos

A etapa de classificação consiste na utilização dos protótipos MBOSS VS para a aplicação do algoritmo K-Vizinho mais próximo (K-NN) conjuntamente com o cálculo da similaridade do cosseno (Seção 4.5.1). Neste caso, cada protótipo representa uma classe do problema que pode ser comparado com uma instância desconhecida que se deseja classificar. Desse modo, o cálculo é realizado para cada protótipo e se verifica qual o rótulo do protótipo que maximizou o resultado da similaridade do cosseno. Logo, a instância desconhecida é classificada com o rótulo encontrado.

A Figura 5.5 ilustra esse processo de classificação. No exemplo, três protótipos relacionados às classes A, B e C do problema e uma instância desconhecida são utilizados para calcular o valor da similaridade do cosseno. De acordo com os resultados obtidos, a instância desconhecida é classificada como pertencente a classe B devido ao maior valor obtido na comparação com o protótipo da classe B.

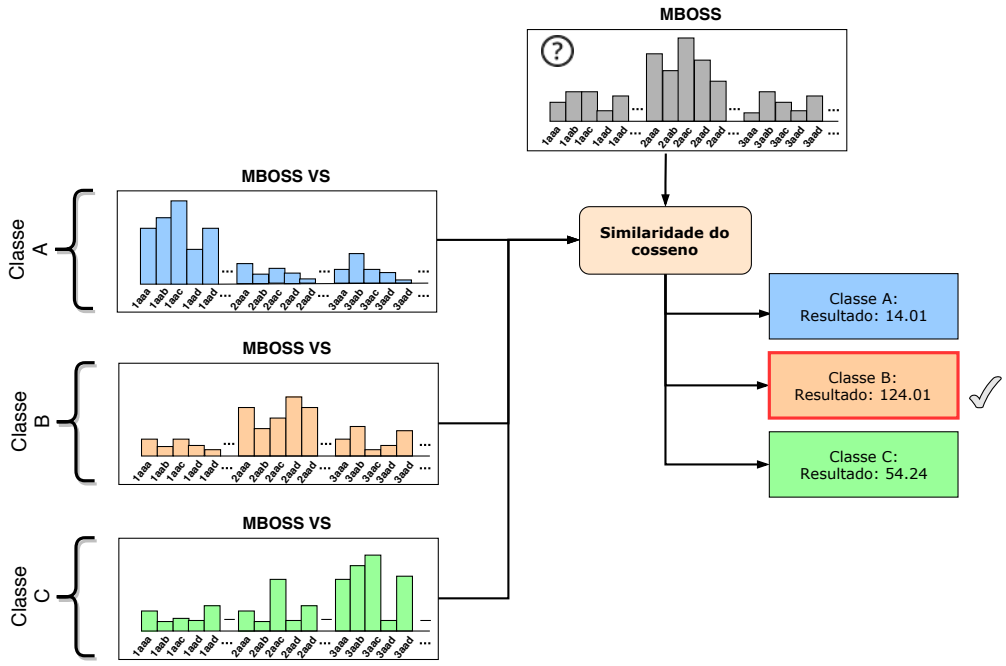


Figura 5.5. Processo de classificação utilizando os protótipos MBOSS VS e o cálculo de similaridade do cosseno. Fonte: Elaborado pelo Autor.

Algoritmo 2 Classificação 1-NN utilizando o modelo MBOSS.

```
1  function predict( map<String, int> tf, map<String, int>[] tfIdfs)
2      double maxSim = 0
3      String bestClass = NULL
4          for int classId in [1..len(tfIdfs)]     // classes
5              double cosSim = dotProduct( tf, tfIdfs[classId])  // similaridade
6              if cosSim > maxSim
7                  maxSim = cosSim
8                  bestClass = classId
9      return label(bestClass)
```

Na prática, a classificação é implementada utilizando o Algoritmo 2. Os protótipos compõem uma estrutura de vetor nomeada de *tfIdfs*, onde cada elemento do vetor é uma estrutura que mapeia uma palavra a um valor inteiro (linha 1). A instância desconhecida é representada pelo vetor de variável *tf* e compõe uma estrutura que mapeia palavras a valores inteiros. A similaridade é calculada pelo produto vetorial entre a instância desconhecida e cada protótipo da estrutura *tfIdfs* (linha 4-5). Por fim, o rótulo de saída é referente à classe que maximizou o resultado da similaridade do cosseno (linhas 6-9).

O produto vetorial utilizado para calcular a similaridade do cosseno tem complexidade computacional igual a $O(n)$, sendo n o tamanho do vetor da instância que foi submetida à classificação. Portanto, a classificação 1-NN MBOSS VS (Algoritmo 2) tem complexidade computacional igual a:

$$T(\text{1-NN MBOSS VS}) = |\text{CLASSES}| \cdot T(\text{Similaridade}) \tag{5.2}$$

$$T(\text{1-NN MBOSS VS}) = |\text{CLASSES}| \cdot O(n),$$

sendo n o tamanho do vetor da instância desconhecida e $|CLASSES|$ a quantidade de classes do problema.

5.5 Classificação de Atividades Humanas Utilizando o MBOSS VS

Este trabalho desenvolve um sistema para reconhecer atividades humanas baseado nos métodos apresentados neste capítulo. O problema consiste de diferentes atividades de movimento (classes) que são submetidas a extração de características e classificação com os métodos de representação simbólica.

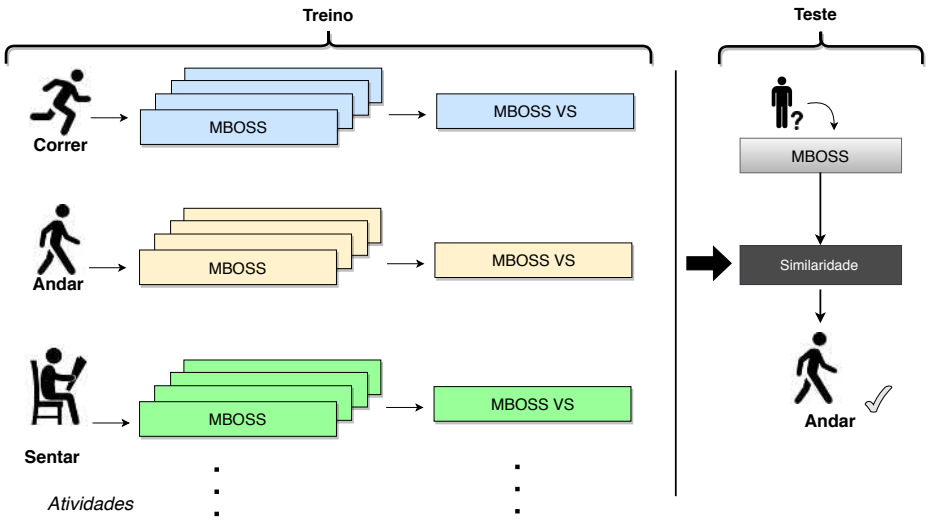


Figura 5.6. Representação de instâncias do modelo MBOSS no modelo *term-frequency inverse-document-frequency*. Fonte: Elaborado pelo Autor.

A Figura 5.6 ilustra, de maneira geral, a proposta deste trabalho para reconhecer atividades. No exemplo, dados de treino fornecem as instâncias necessárias para gerar os modelos MBOSS e MBOSS VS para cada classe de atividade. Os protótipos

gerados são utilizados para classificar instâncias desconhecidas com base no cálculo da similaridade.

Para implementar o sistema e avaliar, uma metodologia baseada em um processo de treino, validação e teste é executado. Ilustrada na Figura 5.7, o processo consiste em particionar a base de dados de atividades em três partes: treino, validação e teste. As partições de treino e validação são utilizadas em um processo repetitivo para otimizar os parâmetros da representação simbólica na etapa de extração de características. Com os melhores parâmetros obtidos, o classificador final é avaliado com a partição de dados de teste.

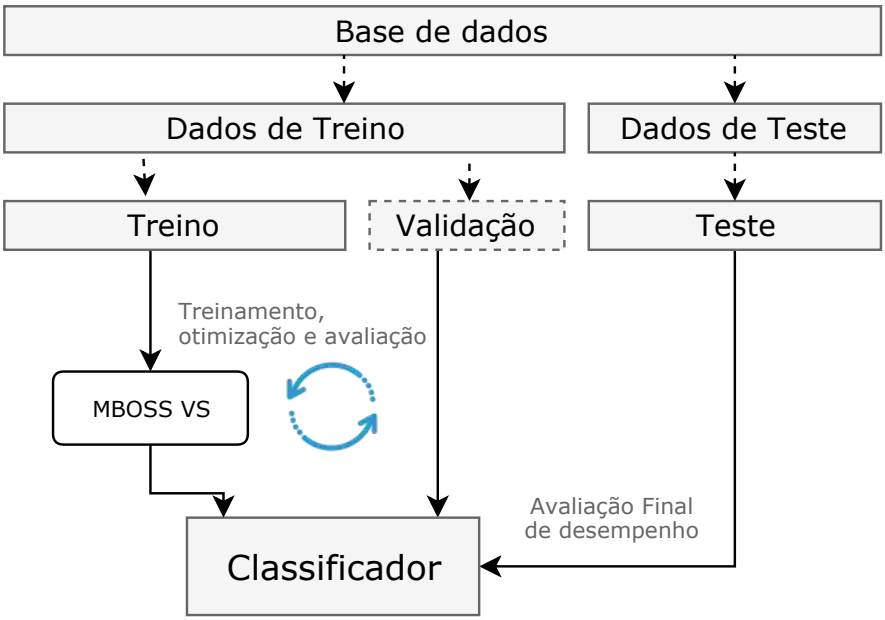


Figura 5.7. Processo executado para gerar o classificador de atividades baseado no MBOSS VS. Fonte: Elaborado pelo Autor.

5.5.1 Otimização dos Parâmetros da Representação Simbólica

Na etapa de extração de características, são três parâmetros que devem ser ajustados para a utilização do MBOSS de maneira eficaz. Esses parâmetros influenciam diretamente no desempenho dos classificadores gerados e devem ser ajustados de acordo com o problema e a aplicação. Dessa maneira, um processo de otimização é implementado com o objetivo de estimar automaticamente os valores mais adequados para esses parâmetros.

Os seguintes parâmetros devem ser otimizados: tamanho de janela w, tamanho de palavra l e tamanho de alfabeto c. A busca pelos parâmetros é realizada fixando o tamanho do alfabeto em $c = 4$ e variando os valores do tamanho da janela e pa-

lavra entre $10 < w < n$, com n sendo o tamanho da série temporal, e $4 < l < 16$, respectivamente. Essas opções de busca são suficientes para a classificação e limitadas com base no projeto de sistemas de RAH em *smartphones*. Por exemplo, com uma plataforma de processamento de 32 bits, os parâmetros $c = 4$ e $l = 16$ equivalem a $4^{16} = 2^{32} = 4.294.967.296$ possíveis palavras, isto é, cada palavra corresponde a um valor inteiro nesta plataforma. Qualquer outra opção deve atender aos requisitos da plataforma de processamento que se deseja implementar, observando a quantidade máxima possível de palavras que podem ser representadas.

Algoritmo 3 Treino dos parâmetros usando *k-fols cross-validation*.

```
1   function fit(MultivariateTimeSerie[] samples, int n, int v, int folds)
2    int c = 4, int maxL = 16
3    int bestScore = 0, int bestL= 0, int bestW = 0
4    map<String, int > bestTfIdfs = []
5        for (int w = 10; w <= in; w += sqrt(n-10))     // itera sqrt(n)
6            map<String, int > sHist = []
7            for int i in [1..len(samples)]  // N
8                sHist[i] = MBOSSTransform(samples[i], v, maxL, c)  // O(n)
9            for int l in [maxL.., 8 , 6 , 4]
10               map<String, int >[] bags = createBags(sHist, l)
11               int score = 0
12               for int i_fold in [1..folds]  // Valiadação cruzada
13                   map<String, int >[] bags_test, bags_train, tfIdfs
14                   bags_test = getFold(i_fold, bags)
15                   bags_train = getOthersFold(i_fold, bags)
16                   tfIdfs = calcTfIdf(bags_train, l)
17                   score += predict_score(bags_test, tfIdfs_train)
18               if score > bestScore
19                   bestScore = score, bestL = l
20                   bestW = w, bestTfIdfs = tfIdfs
21   return (bestScore, bestL, bestW, bestTfIdfs)
```

O processo de otimização de parâmetros é implementado pelo Algoritmo 3 e é baseado na busca por grid (em Inglês, *grid search*) com validação cruzada com $k = 10$ partições para estimar os melhores valores de w e l. O algoritmo itera todas as $\sqrt{n}$ tamanhos de janelas (linha 5). Para cada iteração, é obtido o modelo MBOSS para diferentes tamanhos de janela (linhas 7 e 8). Em seguida, o algoritmo itera sobre as opções de tamanho de palavra, neste caso, $l = [4, 6, 8, 10, 12, 14, 16]$ (linha 9). Para cada tamanho de palavra, são construídas as representações simbólicas de cada amostra de segmento com tamanho de palavra l. Para evitar recalcular várias vezes a transformação MBOSS para diferentes tamanhos de palavras, é calculada primeiramente

a de maior tamanho $l = 16$, que servirá como base para as outras palavras, apenas reconstruindo as representações com tamanhos de palavras menores (linha 10).

Com as representações das amostras, a validação cruzada é aplicada para obter a combinação w e l que obteve a maior taxa de acerto na classificação. Ela é realizada dividindo as amostras em dez partições. Em seguida, é selecionada uma partição para servir de conjunto de teste e as 9 partições restantes são utilizadas para construção do modelo de classificação. Esse processo é repetido até que todas as partições sejam avaliadas e se soma os resultados da classificação de cada partição para obter uma pontuação final. Por fim, a validação cruzada é repetida até que todas as combinações de parâmetros sejam avaliadas, de forma que a saída do algoritmo são os melhores valores estimados para l e w.

A otimização de parâmetros FIT MBOSS (Algoritmo 3) tem complexidade computacional igual a:

$$T(\textit{FIT MBOSS}) = \sqrt{n} \cdot N \cdot [\ T(\textit{MBOSS}) + T(\textit{1-NN MBOSS VS})\ ] \tag{5.3}$$

$$T(\textit{FIT MBOSS}) = \sqrt{n} \cdot N \cdot [\ v \cdot O(n) + |\textit{CLASSES}| \cdot O(n)\ ]$$

$$T(\textit{FIT MBOSS}) = O(N \cdot v \cdot n^{\frac{3}{2}}),$$

sendo n o tamanho da série temporal multivariada de entrada, v o número de variáveis e N a quantidade de séries temporais do problema.

5.5.2 Limitações

Duas principais limitações são observadas no método proposto. A primeira se deve a limitação de espaço necessário para processar as representações simbólicas. Como visto na seção anterior, o projeto e implementação devem levar em consideração a plataforma que se deseja aplicar o sistema e os parâmetros utilizados para a representação simbólica dos dados. Portanto, a limitação se encontra nas opções possíveis de parâmetros que se deseja utilizar e na quantidade de variáveis das séries temporais multivariadas de entrada.

A segunda limitação se encontra no processo de otimização de parâmetros, cujo custo computacional tem ordem polinomial. Isto é, grandes quantidades de dados de treino podem afetar o tempo necessário para produzir a otimização. Trabalhos futuros podem ser direcionados a avaliar outras maneiras de otimizar os parâmetros para afrontar essa limitação.

5.6 Considerações Finais

Este capítulo apresentou o método proposto de representação simbólica desenvolvido para classificar séries temporais multivariadas, nomeado MBOSS. O desenvolvimento do método teve como principal motivação o desenvolvimento de sistemas de RAH com baixo custo computacional. Para isto, uma fusão de dados a nível de características foi implementada estendendo os métodos de representação simbólica, BOSS e BOSS VS, para lidar com séries temporais multivariadas.

Para o desenvolvimento de sistemas RAH, um processo de treino, validação e teste foi apresentado. Dados de treino e validação são utilizados para a otimização dos parâmetros da representação simbólica e os dados de teste são utilizados para avaliação final de desempenho do sistema. Por último, limitações de uso foram pontuadas, especificamente, para o projeto em soluções que tem limitações de recursos como *smartphones*.

O próximo capítulo descreverá o processo de avaliação do método proposto para o problema RAH e os resultados de desempenho obtidos.

Capítulo 6

Experimentos e Resultados

Este capítulo apresenta os experimentos executados para avaliar o MBOSS VS aplicado ao problema de reconhecer atividades humanas. Um protocolo experimental é executado avaliando o método em termos de eficácia de classificação e eficiência em termos do uso dos recursos computacionais. Além disso, bases de dados públicas utilizadas nos experimentos são descritas, bem como as métricas de desempenho e as estratégias de avaliação aplicadas na área de RAH. Por último, os resultados experimentais são apresentados junto as análises e conclusões obtidas.

6.1 Protocolo Experimental

O protocolo experimental divide os experimentos em duas partes:

- **Análise Comparativa de Eficácia de Classificação**: experimentos são executados para verificar a eficácia de classificação do MBOSS VS e comparar com resultados de classificadores de atividades *baseline*. Além disso, duas estratégias de avaliação de modelos de RAH são avaliados, modelo personalizado e modelo generalizado. Esses modelos exploram diferentes configurações experimentais que são frequentemente encontrados nos trabalhos da literatura e possibilitam uma análise abrangente dos classificadores, não limitando a uma configuração específica;
- **Análise de Eficiência de Tempo e Espaço**: experimentos são executados para verificar eficiência do MBOSS VS. Nesta análise são coletadas os tempos de computação dos MBOSS VS para três principais etapas: extração de características, treino do modelo de classificação e classificação. Esses resultados possibilitam executar uma análise do desempenho da implementação e verificar a viabilidade

de aplicação do método para dispositivos móveis. Além disso, experimentos são executados para verificar os ganhos em termos de redução de dados (espaço) com a aplicação da representação simbólica.

Detalhes dos procedimentos executados em cada uma das análises são descritos nas seções seguintes.

6.2 Análise Comparativa de Eficácia de Classificação

A análise comparativa tem proposito de verificar a eficácia do MBOSS VS para a tarefa de classificação de atividades humanas e comparar os resultados com classificadores baseados na abordagem convencional de extrair manualmente características dos dados. Além disso, alguns pontos de investigação são explorados:

- Qual o comportamento dos classificadores avaliados quando submetidos a bases de dados com diferentes configurações como: atividades observadas, técnicas e parâmetros de segmentação e diversidade de usuários participantes da coleta de dados;
- Qual o comportamento dos classificadores avaliados quando submetidos a estratégias de avaliação de modelos RAH como: o modelo personalizado e modelos generalizado.

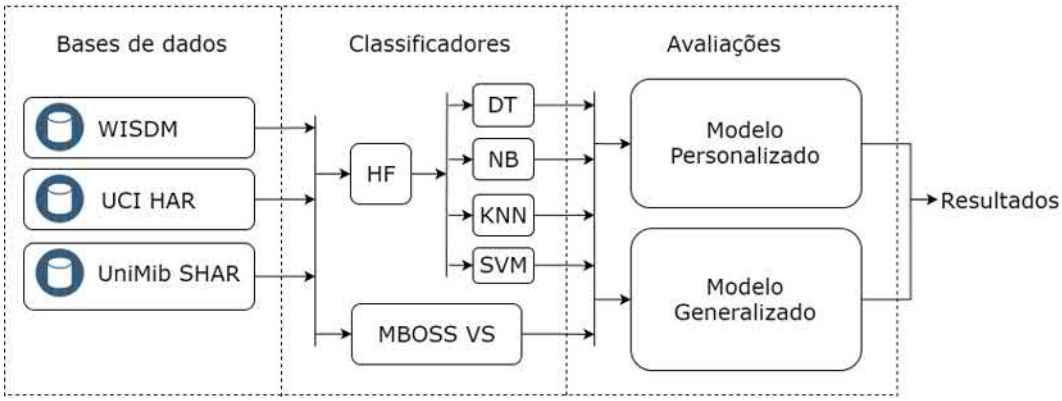


Figura 6.1. Diagrama da análise comparativa de classificação. Fonte: Elaborado pelo Autor.

Um resumo dos elementos experimentais é ilustrado na Figura 6.1. As três bases de dados são WISDM, UCI HAR e UniMiB SHAR. Detalhes das características de cada

uma das bases são detalhados na Seção 6.2.1. Os classificadores baseline são algoritmos de aprendizagem de máquina convencionais como Árvore de decisão (DT), Naive bayes (NB), K-Vizinhos próximos (KNN) e Máquina de Vetores de Suporte (SVM). Esses classificadores são treinados para classificar atividades com base na abordagem manual de extrair características no domínio do tempo e frequência. Detalhes dessa abordagem são descritas na Seção 6.2.2. Resultados são obtidos para as duas estratégias de avaliação: modelo personalizado e modelo generalizado. Detalhes da execução dessas estratégias são descritas na Seção 6.2.3.

6.2.1 Bases de Dados

A Tabela 6.1 apresenta um resumo das características das bases de dados utilizados nos experimentos. As principais características são a quantidade de usuários que participaram da coleta de dados, o número de atividades (classes) e as configurações de segmentação implementadas com base nas informações das publicações originais das bases de dados. Mais detalhes das implementações são apresentadas nas próximas seções.

Tabela 6.1. Resumo das características das bases de dados utilizadas nos experimentos.

Coleção de dados	**WISDM**	**UCI HAR**	**UniMib SHAR**
Número de usuários	19	30	30
Número de atividades	6	6	9
Tamanho da janela	200 (10 seg.)	128 (2,56 seg.)	151 (3 seg.)
Total de janelas	3345	10299	7579
Segmentação	Janela deslizante de tamanho fixo sem sobreposição.	Janela deslizante de tamanho fixo com 50% de sobreposição.	Janela centrada ao redor de um pico na magnitude do sinal.

6.2.1.1 WISDM

A primeira base de dados foi fornecida pelo laboratório de *Wireless Sensor Data Mining* (WISDM) do Departamento de Computação da Universidade de Fordham (Kwapisz et al., 2011). Ela consiste de uma coleta de dados do acelerômetro obtida a partir de diferentes tipos de *smartphones* Android (Nexus One, HTC Hero e Motorola Back-flip) e transportados em um dos bolsos frontais da calça do usuário. Para

os experimentos deste trabalho foram utilizados os dados de 19 usuários.[1] Cada um deles executou 6 atividades de movimento pré-definidas: *walking* (caminhar), *jogging* (correr), *downstairs* (descer escadas), *upstairs* (subir escadas), *sitting* (sentar) e *standing* (ficar em pé). A coleta foi supervisionada pelos responsáveis da pesquisa, mas não foi mencionada a metodologia utilizada para rotular os dados. As amostras foram coletadas a uma taxa fixa de frequência de 20Hz. A etapa de pré-processamento está de acordo com a metodologia utilizada pelos respectivos autores originais, na qual foi aplicada uma segmentação com o algoritmo de janela deslizante de tamanho fixo igual a 200 amostras por segmento, sem sobreposição de janelas. Nenhuma normalização ou filtragem foi aplicada aos segmentos.

Ao total foram obtidos 3345 segmentos com informações de 6 diferentes classes de atividades humanas. A distribuição das instâncias na base de dados em relação às classes de atividade é apresentada na Tabela 6.2.

Tabela 6.2. Distribuições das instâncias na base de dados WISDM.

Classes	Distribuição das instâncias
walking	35, 78%
jogging	30, 76%
upstairs	10, 82%
downstairs	9, 17%
sitting	7, 83%
standing	5, 62%

6.2.1.2 UCI HAR

A segunda base de dados foi fornecida pelo repositório de aprendizagem de máquina da Universidade da Califórnia em Irvine (UCI) (Anguita et al., 2013b). A base consiste de uma coleta de dados dos sensores inerciais de um smartphone Android Galaxy SII. Os dados foram coletados de 30 usuários. Cada um deles executou 6 atividades de movimento pré-definidas: *walking* (caminhar), *walking upstairs* (subir escadas), *walking downstairs* (descer escadas), *sitting* (sentar), *standing* (ficar em pé) e *laying* (deitar). As amostras dos sensores foram capturadas à frequência fixa de 50Hz, e os usuários executaram duas vezes uma sequência de atividades. Na primeira execução, o usuário transportou o smartphone no lado esquerdo da cintura, e na segunda execução o sujeito ficou livre para situar o smartphone à sua escolha. A coleta

[1] A base de dados WISDM original contém dados de 36 usuários, entretanto, nem todos apresentam instâncias para cada classe de atividade. Desse modo, somente os que apresentam foram selecionados. Os identificadores dos usuários selecionados são: 3, 5, 6, 7, 8, 12, 13, 18, 19, 20, 21, 24, 27, 29, 31, 32, 33, 34, 36.

foi realizada em laboratório e rotulada manualmente com auxílio de filmagens produzidas pelos autores. Para os experimentos foram selecionadas apenas as informações do acelerômetro. Esses dados foram pré-processados segundo a metodologia dos autores originais, cuja segmentação se aplicou o método de janela deslizante de tamanho fixo igual a 2,56 segundos, equivalente a 128 amostras por segmento. Ao contrário da coleção WISDM, neste caso se aplicou a sobreposição de janela em 50% dos dados. Nenhuma normalização ou filtragem foi aplicada aos segmentos.

Ao total foram obtidos 10299 segmentos com informações de 6 diferentes classes de atividades humanas. A distribuição das instâncias na base de dados em relação às classes de atividade é apresentada na Tabela 6.3.

Tabela 6.3. Distribuições das instâncias na base de dados UCI HAR.

Classes	Distribuição das instâncias
walking	$17,30\%$
walking upstairs	$15,48\%$
walking downstairs	$12,86\%$
sitting	$17,84\%$
standing	$18,21\%$
laying	$18,28\%$

6.2.1.3 UniMib SHAR

A base de dados UniMib SHAR foi fornecida pelo departamento de informática da Universidade de Milano-Bicocca (Micucci et al., 2017). A coleção foi projetada para atender as tarefas de reconhecer atividades diárias, e também, de atividades de quedas. Além do mais, os autores tomaram um cuidado extra na seleção dos voluntários observando alguns atributos, como altura, peso, idade e sexo. A coleção for construída coletando dados do acelerômetro de 30 voluntários com idade entre 18 a 60 anos, 24 mulheres e 6 homens, com uma média de altura $169 \pm 7cm$ e média de peso $64 \pm 10kg$. Um smartphone Android Galaxy Nexus I9250 foi utilizado para coleta das amostras a uma taxa de frequência fixa de 50Hz. A coleta foi realizada em laboratório e rotulada manualmente com auxílio de gravações de áudio do microfone do dispositivo. Aos usuários foi solicitado que transportassem o smartphone nos bolsos frontais da calça, metade do tempo no bolso esquerdo e a outra metade no bolso direito, e executassem uma sequência de 9 atividades de movimento e 8 atividades relacionadas a quedas. Para os experimentos somente os dados relacionados a atividades de movimento foram utilizados para avaliar os métodos, conforme apresentado na Tabela 6.4.

Tabela 6.4. Sumário das atividades de movimento da base de dados UniMib SHAR.

Atividades	Abreviatura
Ficar em pé a partir de sentado	StandingUpFS
Ficar em pé a partir de deitado	StandingUpFL
Caminhar	Walking
Correr	Running
Subir Escadas	GoingUpS
Pular	Jumping
Descer Escadas	GoingDownS
Deitar a partir de sentado	LyingDownFS
Sentar	SittingDown

Diferente das bases de dados descritas anteriormente, esta base foi segmentada pelos autores analisando a magnitude $mag(t) = \sqrt{a_x(t)^2 + a_y(t)^2 + a_z(t)^2}$ do sinal do acelerômetro. O método consiste na aplicação da técnica de janela deslizante para extrair segmentos de 3 segundos toda vez que um pico na magnitude é encontrado. Um pico é somente válido se atender as seguintes condições: (1) A magnitude do sinal mag_t é maior que $1.5g$, sendo g o valor da constante de gravitação universal; (2) A magnitude do sinal m_{t-1} no instante $t-1$ é menor que $1.5g$. Essas condições são justificadas pelos autores por se tratar de uma base de dados para classificar tanto atividades de movimento quanto quedas. Cada segmento de 3 segundos é centrado no pico encontrado, e nenhuma normalização ou filtragem foi aplicada.

Ao total foram obtidas 7579 instâncias com informações de 9 classes de atividades humanas. A distribuição das instâncias na base de dados em relação às classes de atividade é apresentada na Tabela 6.5.

Tabela 6.5. Distribuições das instâncias na base de dados UniMiB SHAR.

Classes	Distribuição das instâncias
StandingUpFS	$2,01\%$
StandingUpFL	$2,84\%$
Walking	$22,93\%$
Running	$26,19\%$
GoingUpS	$12,15\%$
Jumping	$9,84\%$
GoingDownS	$17,46\%$
LyingDownFS	$3,90\%$
SittingDown	$2,63\%$

6.2.2 Classificadores Baseline

Para comparar os resultados obtidos pelo método MBOSS, quatro diferentes algoritmos de classificação foram avaliados utilizando a ferramenta WEKA (em Inglês, Waikato Environment for Knowledge Analysis) (Witten et al., 2016). Essa ferramenta contém um conjunto de algoritmos de aprendizagem de máquina e uma interface intuitiva que facilita a execução dos experimentos. Os algoritmos de classificação utilizados são: Árvore de Decisão (DT) implementada com J48, Naive Bayes (NB), K-Vizinhos Próximos (KNN) com $k = 1$, e Máquina de Vetores de Suporte (SVM) implementado com SMO One-vs-One com kernel polinomial. Cada algoritmo é treinado com um conjunto de características do domínio do tempo e da frequência (*Hand-Crafted Features*). As características utilizadas foram selecionadas manualmente observando a aplicabilidade em dispositivos móveis e o desempenho obtido nos resultados experimentais obtidos por Figo et al. (2010) e Shoaib et al. (2014) . A Tabela 6.6 apresenta as caraterísticas selecionadas para os experimentos.

Tabela 6.6. Conjunto de características utilizadas nos experimentos com os baselines.

Domínio	**Sinal**	**Característica**
Tempo	x-, y- e z-eixos do Acelerômetro	Média, Mediana, Mínimo, Máximo, Variança, Desvio Padrão, *Zero crossing rate* e *Root mean square*
Frequência	x-, y- e z-eixos do Acelerômetro	Componente DC (0Hz), Soma dos cinco primeiros coef. de Fourier (0.5-3Hz), Energia espectral e Entropia espectral

No total, 36 características foram obtidas dos dados do acelerômetro tri-axial, 12 características para cada eixo (componente).

6.2.3 Estratégias de Avaliação

Para avaliação dos classificadores RAH, duas estratégias de avaliação foram adotadas. Ambas são baseadas nos modelos personalizado e generalizado. Ilustrados na Figura 6.2, esses modelos foram apresentados por Lockhart & Weiss (2014) e estão diretamente relacionados com as aplicações do mundo real. O modelo personalizado se aplica na construção de classificadores de atividades utilizando apenas dados de um único usuário, ao passo que o modelo generalizado se aplica na construção de clas-

sificadores de atividades utilizando dados de um conjunto de usuários. Detalhes da implementação são apresentados a seguir.

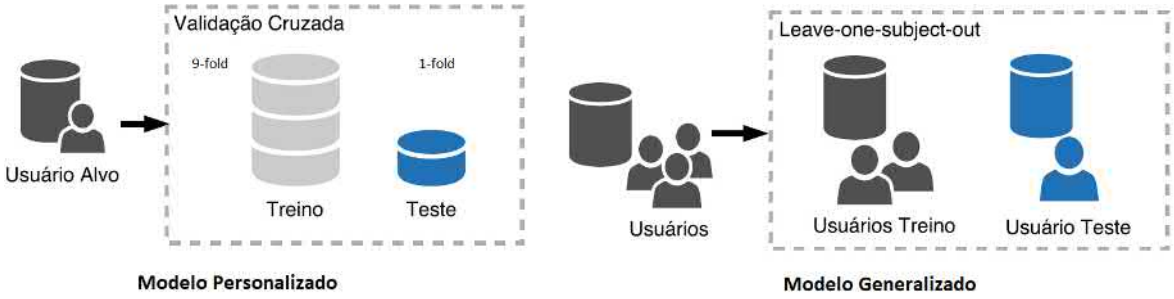


Figura 6.2. Modelos de avaliação utilizados nos experimentos: Modelo Personalizado (esquerda) e Modelo Generalizado (direita). Fonte: Elaborado pelo Autor.

Avaliação do modelo personalizado: a avaliação consiste em selecionar um usuário alvo dentre um conjunto de usuários da base de dados. Os dados do usuário alvo são particionados em dez partes (*10-fold cross-validation*), na qual uma parte é utilizada como conjunto de teste e as nove partes restantes são utilizadas como conjunto de treino. Trabalhos da literatura identificam esta estratégia como modelo pessoal, ou modelo dependente do usuário (do Inglês, *user-dependent*).

Avaliação do modelo generalizado: a avaliação consiste em selecionar um usuário alvo entre um conjunto n de usuários, separar os dados do usuário alvo para formar o conjunto de teste e reunir os dados dos $(n-1)$ usuários restantes para formar o conjunto de treino. Na literatura, esse tipo de avaliação também é chamada de *cross validation leave-one-subject-out* ou de modelo independente do usuário (do Inglês, *user-independent*).

As duas avaliações são executadas para todos os usuários da base de dados e os resultados da classificação são apresentados com o cálculo das métricas de desempenho.

6.2.4 Métricas de Desempenho

Nos experimentos foram utilizados os resultados da matriz de confusão para calcular as seguintes métricas de desempenho: Acurácia e a Macro Medida-F. Esse conjunto de métricas demostram, em termos matemáticos, o quão eficaz é o classificador avaliado (Sokolova & Lapalme, 2009). A matriz de confusão é gerada a partir dos valores reais e preditos (classificados) das instâncias de teste submetidas a classificação. Os elementos necessários para construção da matriz são exemplificados na Tabela 6.7 e descritos a seguir, juntamente, com os cálculos das métricas de desempenho.

Tabela 6.7. Matriz de confusão para um problema de classificação com 3 classes.

		Rótulo predito			
		1	2	3	*soma*
Rótulo verdadeiro	1	VP_{11}	FP_{12}	FP_{13}	NI_1
	2	FP_{21}	VP_{22}	FP_{23}	NI_2
	3	FP_{31}	FP_{32}	VP_{33}	NI_3
	soma	NT_1	NT_2	NT_3	*total*

Para auxiliar na descrição dos cálculos, algumas variáveis são apresentadas primeiramente: C é a quantidade de classes do problema; VP_{ii} (Verdadeiro-Positivo) é a quantidade de amostras de teste da classe i que foram corretamente classificadas para o rótulo i; FP_{ij} (Falso-Positivo) o total de amostras de teste da classe i que foram incorretamente classificadas com rótulo j; NI_i é a quantidade de amostras de teste da classe i; e NT_j o total de amostras de testes que foram classificadas com o rótulo j. As duas últimas podem ser calculadas pelas seguintes equações:

$$NI_i = VP_{ii} + \sum_{j=1, j\neq i}^{C} FP_{ij} \ ; \tag{6.1}$$

$$NT_j = VP_{jj} + \sum_{i=1, i\neq j}^{C} FP_{ij} \ ; \tag{6.2}$$

Acurácia representa a taxa de acerto do classificador e pode ser interpretada como a probabilidade de se classificar corretamente uma instância de teste. Ela pode ser determinada pela quantidade de instâncias que foram corretamente classificadas em razão da quantidade total de instâncias classificadas ou a partir da seguinte equação:

$$Acurácia = \frac{\sum_{i=1}^{C} VP_{ii}}{total} = \frac{\sum_{i=1}^{C} VP_{ii}}{\sum_{i=1}^{C} NI_i} = \frac{\sum_{j=1}^{C} VP_{jj}}{\sum_{j=1}^{C} NT_j}. \tag{6.3}$$

Precisão representa a média ponderada da fração dos rótulos preditos que estão corretamente classificados para cada classe de atividade. A precisão para a classe i pode ser calculada pela seguinte equação:

$$Precisão_i = \frac{VP_{ii}}{NT_i} \ . \tag{6.4}$$

Revocação representa a média ponderada da fração dos rótulos verdadeiros que foram corretamente classificados para cada classe de atividade. A revocação para a

classe i pode ser calculada pela seguinte equação:

$$Revocação_i = \frac{VP_{ii}}{NI_i} \ , \tag{6.5}$$

Medida-F (do Inglês *F-score*) proporciona uma maneira de combinar precisão e revocação em uma única métrica. A Medida-F é calculada da seguinte forma:

$$Medida\text{-}F_i = \frac{2 \cdot Precisão_i \cdot Revocação_i}{Precisão_i + Revocação_i} \ . \tag{6.6}$$

Para lidar com possíveis problemas de desbalanceamento das classes nas bases de dados, a Macro Medida-F é utilizada em substituição à média da Medida-F. A Macro Medida-F é obtida da seguinte forma:

$$Macro\ Medida\text{-}F = \frac{\sum_{i=1}^{c} w_i \cdot \text{Medida-F}_i}{\sum_{i=1}^{c} w_i}, \tag{6.7}$$

onde Medida-F$_i$ representa a Medida-F para a classe i e w_i corresponde ao número de instâncias da classe i.

6.3 Análise de Eficiência de Tempo e Espaço

Esta análise tem o propósito de verificar o desempenho MBOSS VS para a tarefa de classificação de atividades humanas. Os seguintes pontos são analisados:

- Qual a eficiência do MBOSS VS para classificar atividades em comparação a abordagem convencional;
- Qual a eficiência do MBOSS VS na redução do espaço de dados necessário para armazenamento e processamento dos dados.

Os experimentos são divididos em duas partes: tempo de computação e espaço de dados.

6.3.1 Tempo de Computação

São coletados dos experimentos os tempos de computação nas três principais etapas dos sistemas RAH. Estes são:

- **Extração de características**: tempo de computação requerido para gerar o conjunto de características (representação) sobre o conjunto de teste.

- **Treino do modelo de classificação**: tempo de computação requerido para gerar o modelo de classificação e avaliar sua classificação sobre o conjunto de treino.

- **Classificação**: tempo de computação requerido para classificar somente o conjunto de teste.

Para os experimentos é utilizado um computador Desktop Dell OptiPlex 9020 (Intel Core i7-4790/32GB DDR3). A base de dados utilizada é a UCI HAR, na qual está dividida em dois conjuntos de dados: treino e teste. O conjunto de treino corresponde aos dados de 21 pessoas, que equivale a um total de 7352 instâncias de dados de sensores, e o conjunto de teste corresponde aos dados de 9 pessoas, que equivale a um total de 2947 instâncias. Cada instância corresponde a uma janela de tamanho 128. A técnica de segmentação é baseada na janela deslizante com 50% de sobreposição. Por fim, os resultados dos experimentos são a média dos tempos de dez execuções para todos os classificadores da análise comparativa de classificação.

6.3.2 Espaço de Dados

Para avaliação do espaço de dados, um experimento de transformação dos dados brutos para a representação simbólica é executado para um conjunto com 10000 instâncias obtidas da base de dados UCI HAR. Cada instância corresponde a uma série temporal de tamanho igual a 128 amostras. Cada amostra é representada em memória por um valor com 8 bytes (double). Logo, um instancia corresponde a um total de 1024 bytes.

Os experimentos são realizados para transformação simbólica com o MBOSS VS com e sem redução de numerosidade. Os resultados dos experimentos são os tamanhos de espaço em bytes dos dados brutos logo após a transformação.

6.4 Resultados da Análise de Eficácia de Classificação

Esta seção apresenta os resultados experimentais obtidos da análise comparativa do classificadores de RAH. Os resultados são divididos em duas partes, a primeira para a avaliação do modelo personalizado, e a segunda para o modelo generalizado. A seção encerra com uma discursão geral dos resultados obtidos.

6.4.1 Modelo Personalizado

Esta avaliação foi conduzida para verificar o desempenho do método proposto conforme o procedimento de geração de modelos de classificação de atividades pessoais, isto é, próprio para um usuário específico. Para isto, cada experimentação (treino e teste) foi realizado com somente os dados de um usuário da base. As bases WISDM, UCI HAR e UniMiB SHAR têm, respectivamente, em média 176, 343 e 253 instâncias por usuário. A experimentação foi realizada para todos os usuários da base. As Figuras 6.3 e 6.4 mostram os resultados consolidados da experimentação, respectivamente, para Acurácia e a Macro Medida-F.

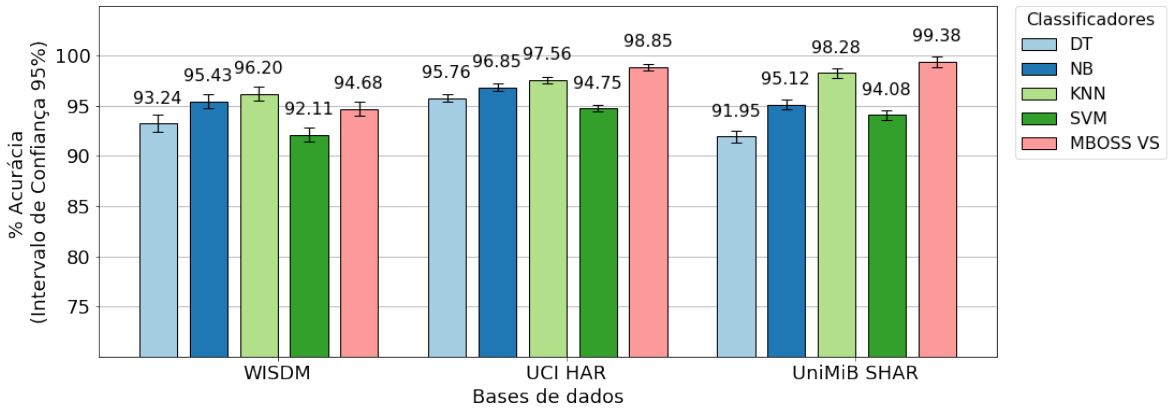


Figura 6.3. Resultados consolidados da Acurácia para os classificadores de atividades no modelo personalizado.

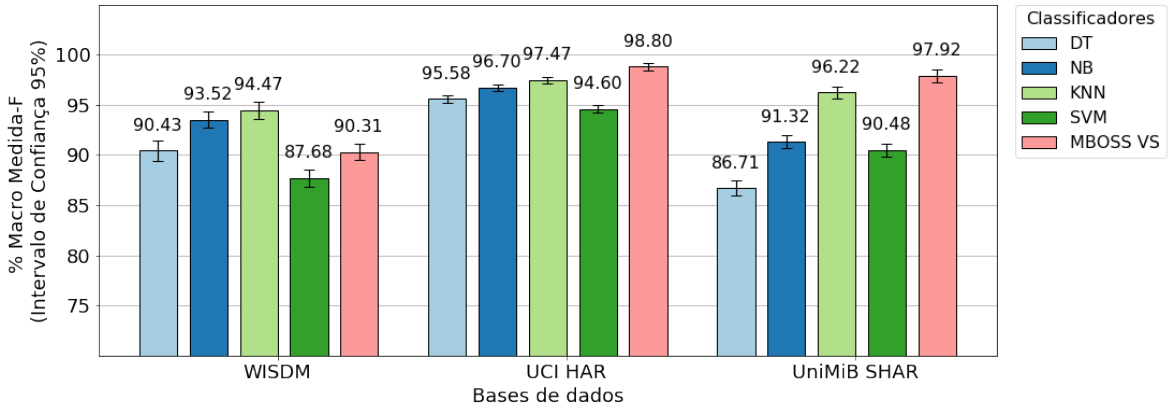


Figura 6.4. Resultados consolidados da Macro Medida-F para os classificadores de atividades no modelo personalizado.

A questão principal de interesse é verificar se o método proposto é equivalente ou superior aos classificadores baseados na extração manual de características. Os resul-

tados revelam dois interessantes achados. O primeiro achado mostra que o classificador MBOSS VS, em geral, apresenta resultados satisfatórios (mais de 90% de Acurácia) para todas as três bases de dados. O segundo revela que o MBOSS VS demostra uma vantagem sobre os classificadores adversários nas bases UCI HAR e UniMiB SHAR, com 98,85% e 99,37% para a Acurácia e com 98,79% e 97,91% para a Macro Medida-F, respectivamente. Para a base WISDM, o melhor desempenho foi obtido pelo classificador KNN, com 96,2% (1,53% superior ao MBOSS VS) para a Acurácia e 94,5% (4,17% superior ao MBOSS VS) para a Macro Medida-F.

As Figuras 6.5, 6.6, 6.7 e 6.8 apresentam as matrizes de confusão[2] obtidas para os classificadores MBOSS VS e KNN. A análise por classe de atividade na base WISDM mostra que o KNN é melhor que o MBOSS VS para classificar as atividades estáticas com sentar (sitting) e ficar em pé (standing). Contudo, essa inferioridade não ocorre nas demais bases. Uma razão para esta fato está na diferença do tamanho de segmento entre as bases de dados. No caso das bases UCI HAR e UniMiB SHAR, cada instância contém no máximo 3 segundos de amostras. Por outro lado, na WISDM, cada instância contém 10 segundos de amostras. Logo, um tamanho de segmento maior penaliza o MBOSS VS para atividades consideradas estáticas (sentar e ficar em pé).

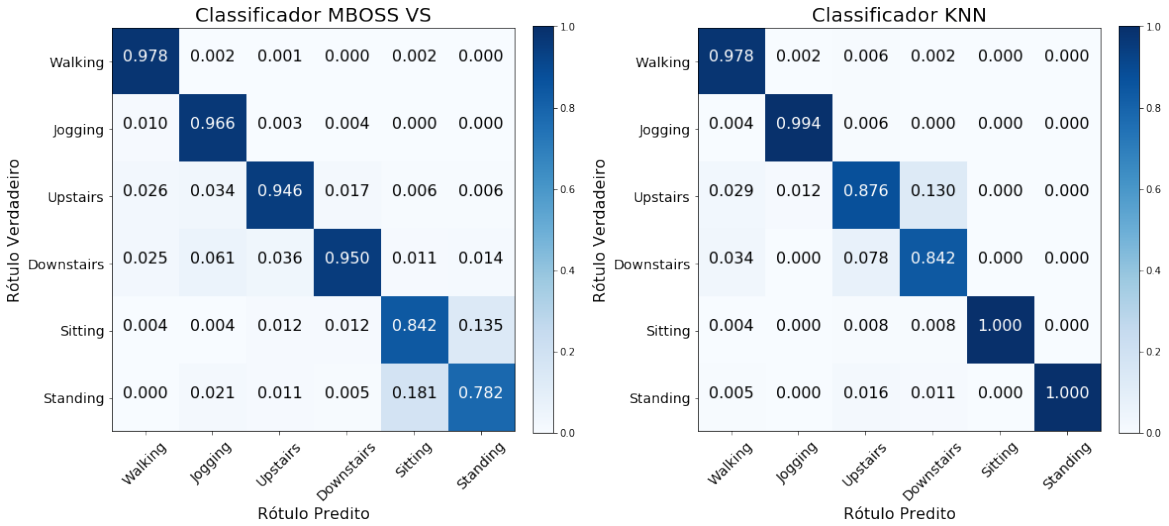


Figura 6.5. Matrizes de confusão do MBOSS VS e KNN para a base de dados WISDM. Fonte: Elaborado pelo Autor.

[2]As matrizes de confusão foram normalizadas com base na revocação (métrica de desempenho), isto é, a diagonal principal apresenta os valores da revocação para cada classe de atividade.

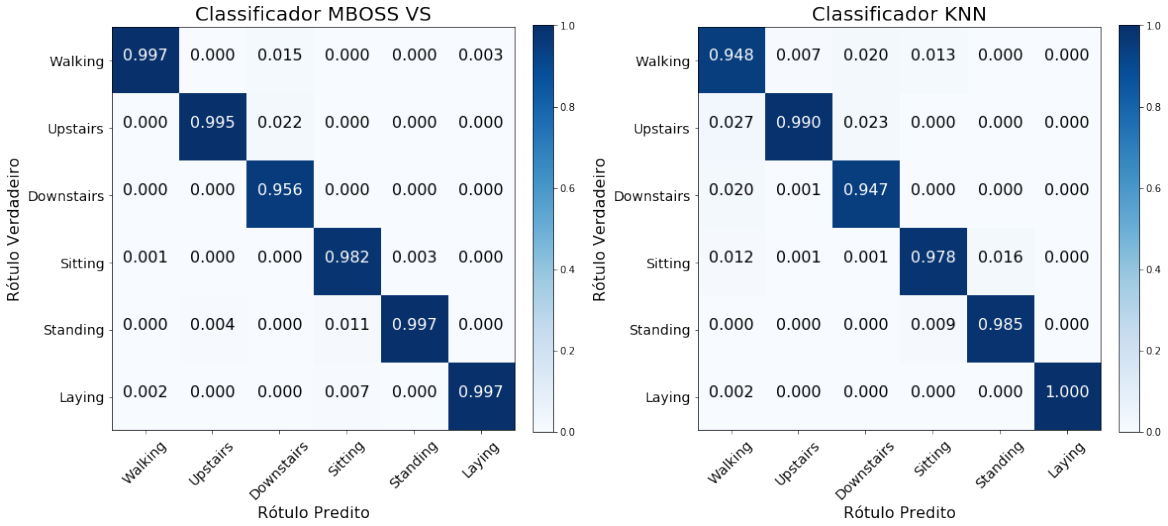


Figura 6.6. Matrizes de confusão do MBOSS VS e KNN para a base de dados UCI HAR. Fonte: Elaborado pelo Autor.

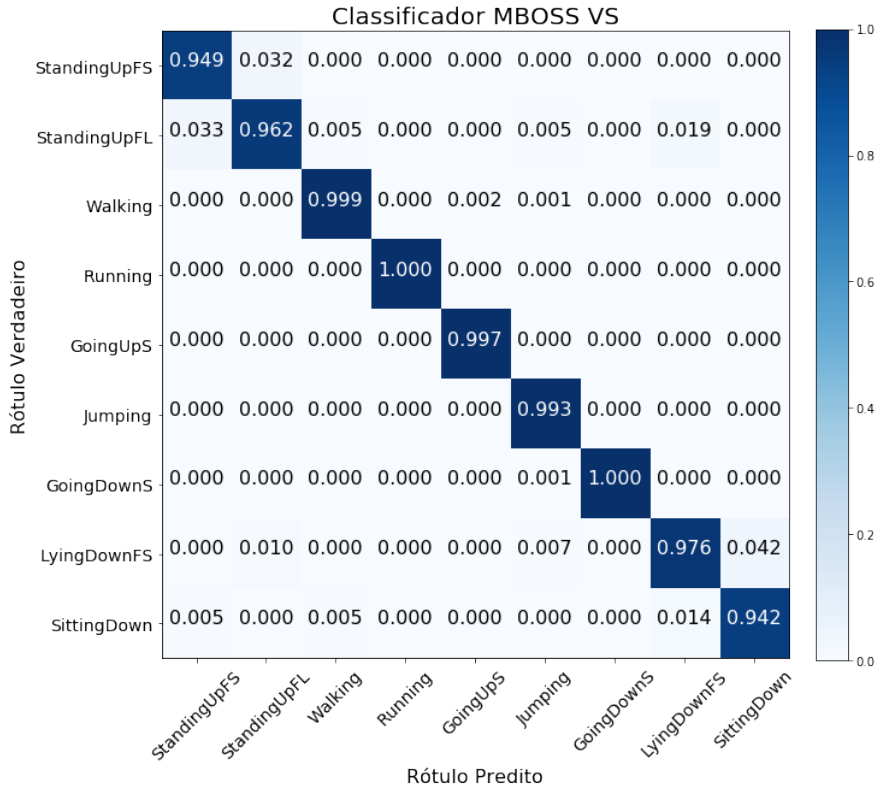


Figura 6.7. Matriz de confusão do classificador MBOSS para a base de dados UniMiB SHAR.

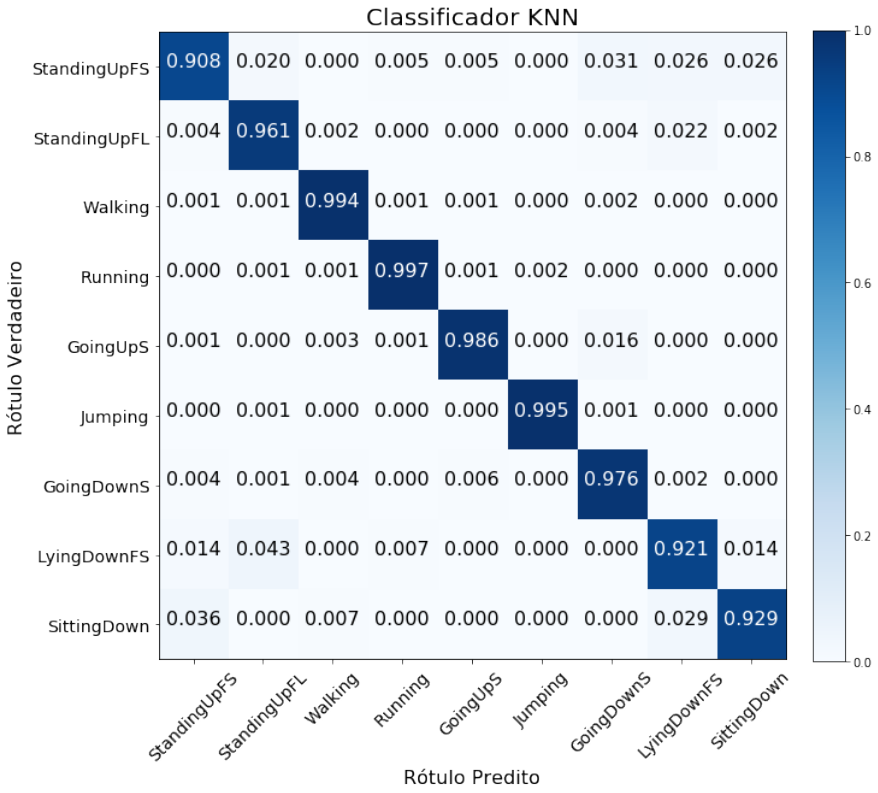


Figura 6.8. Matriz de confusão do classificador KNN para a base de dados UniMiB SHAR. Fonte: Elaborado pelo Autor.

6.4.2 Modelo Generalizado

Esta avaliação foi conduzida para verificar o desempenho do método proposto conforme o procedimento de geração de modelos de classificação de atividades impessoais, isto é, para generalizar para diferentes indivíduos. Esta estratégia insere mais dificuldade à classificação e permite verificar a capacidade de generalização dos classificadores avaliados. Cada experimentação, neste caso, compreende uma partição de teste composta por instâncias de dados de um usuário e uma partição de treino com instâncias dos usuários restantes da base de dados. Esse processo foi realizado alternando o usuário de teste até que todos sejam avaliados. As Figuras 6.9 e 6.10 apresentam os resultados consolidados da experimentação, respectivamente, para Acurácia e a Macro Medida-F.

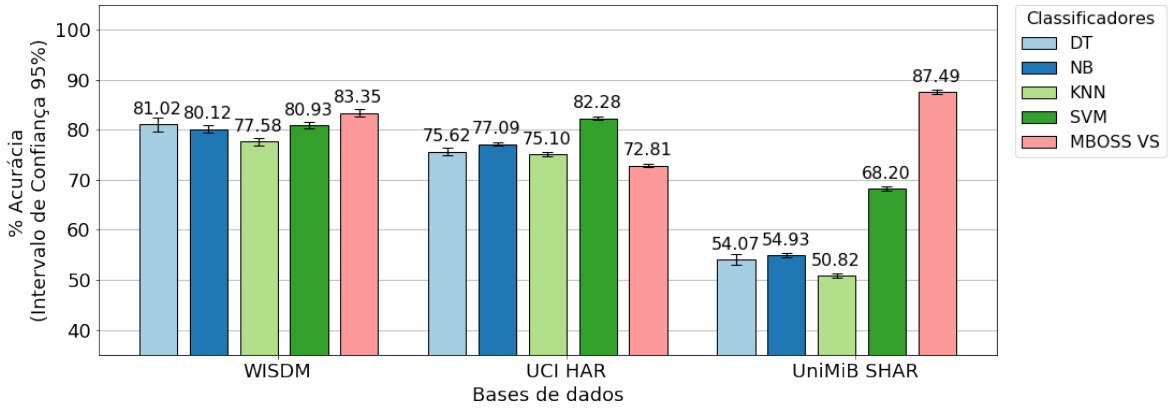


Figura 6.9. Resultados consolidados da Acurácia para os classificadores de atividades no modelo generalizado.

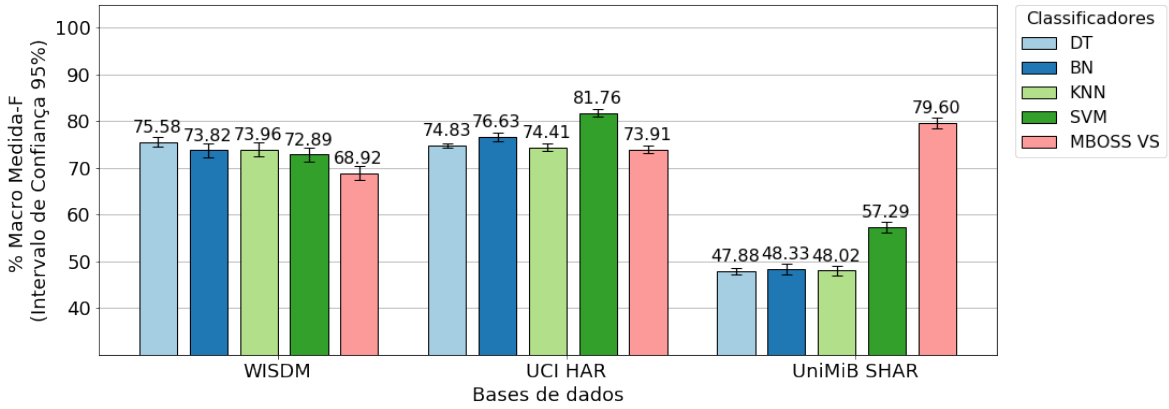


Figura 6.10. Resultados consolidados da Macro Medida-F para os classificadores de atividades no modelo generalizado.

Os resultados da comparação destacam três pontos. Primeiro, os resultados do MBOSS VS e dos demais classificadores são, em geral, menores que os obtidos na avaliação do modelo personalizado (Acurácia abaixo de 90%). Esses resultados evidenciam a dificuldade adicional de classificação, uma vez que os classificadores não têm conhecimento prévio sobre os padrões de atividades do usuário teste. Segundo ponto, o classificador MBOSS VS mostra uma vantagem significante em relação aos adversários na base UniMiB SHAR, com 87,49% e 79,60% para Acurácia e a Macro Medida-F respectivamente. A diferença entre o segundo colocado (SVM) é de 19,29% e 22,32%, respectivamente. Dois motivos justificam esta diferença. O primeiro se deve a maior dificuldade de classificação em relação ao número de classes de atividades, que neste caso a UniMiB SHAR contém instâncias de nove classes ao contrário das demais (WISDM e UCI HAR) que contêm apenas seis classes. Desse modo, os classificadores

adversários foram penalizados com o maior numero de atividades presentes na base de dados UniMiB SHAR. O segundo motivo se deve a técnica de segmentação utilizada, que neste caso, a janela centrada no pico da magnitude demostrou ser a melhor configuração para o classificador MBOSS VS. Por fim, o último ponto a destacar é a diferença significativa observada entre a Acurácia e a Macro Medida-F (14,43%) para o MBOSS VS na base de dados WISDM. O pressuposto desta diferença se deve ao desbalanceamento da base, que apresenta 66% das instâncias concentradas em duas classes de atividades (walking e jogging).

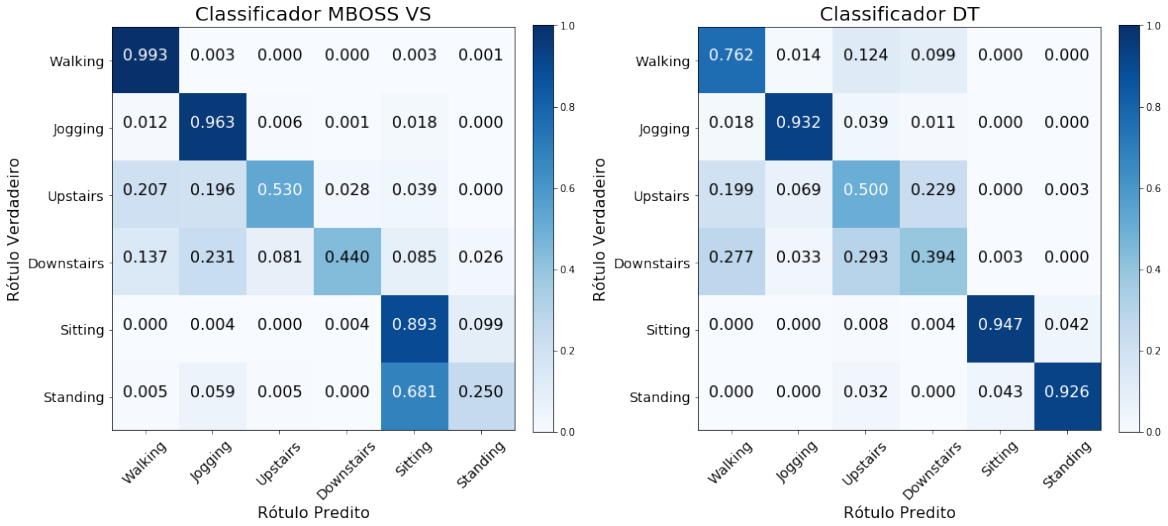


Figura 6.11. Matrizes de confusão do MBOSS VS e DT para a base de dados WISDM. Fonte: Elaborado pelo Autor.

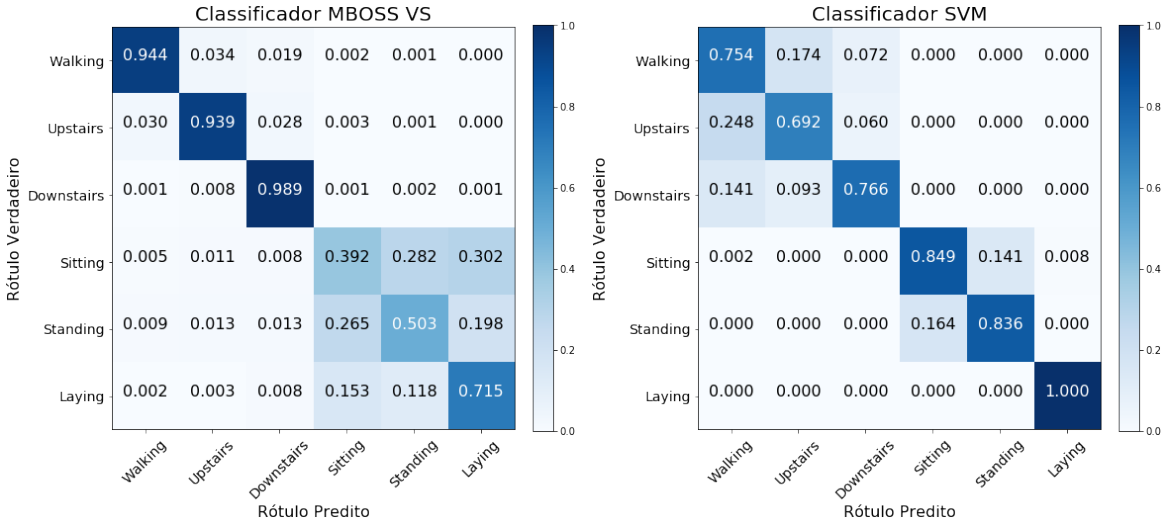


Figura 6.12. Matrizes de confusão do MBOSS VS e SVM para a base de dados UCI HAR. Fonte: Elaborado pelo Autor.

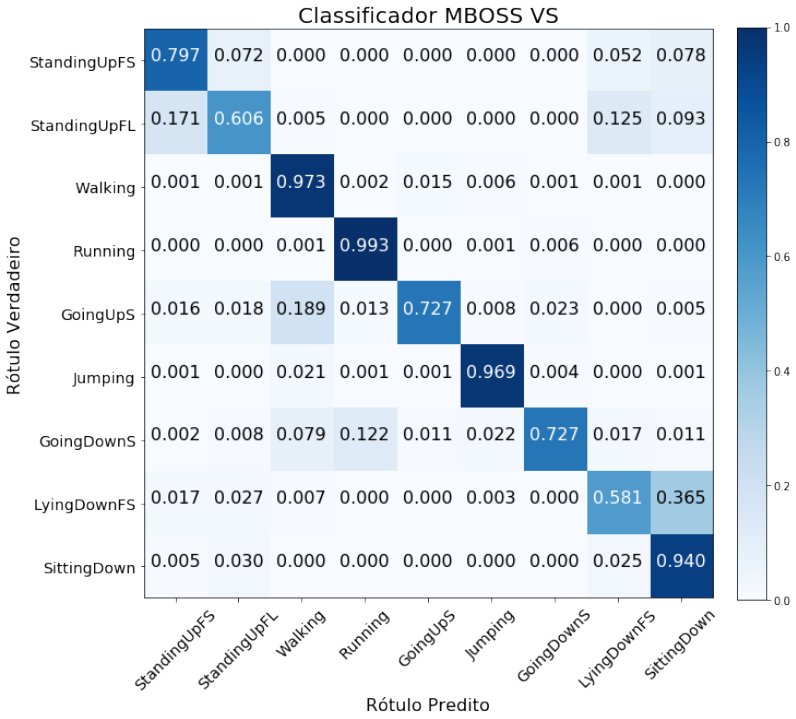


Figura 6.13. Matriz de confusão do classificador MBOSS para a base de dados UniMiB SHAR. Fonte: Elaborado pelo Autor.

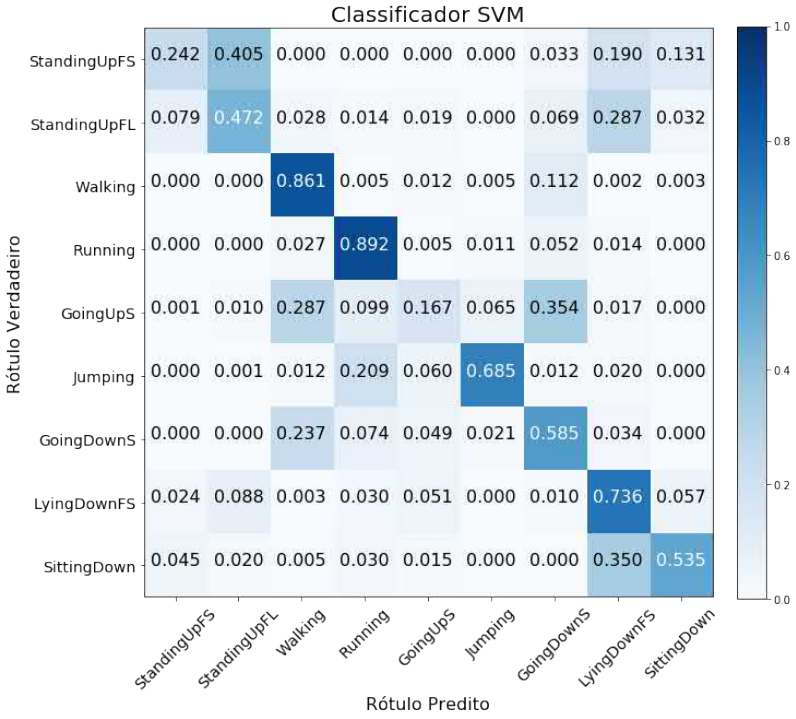


Figura 6.14. Matriz de confusão do classificador SVM para a base de dados UniMiB SHAR. Fonte: Elaborado pelo Autor.

A Figura 6.11 apresenta as matrizes de confusão obtidas para os classificadores MBOSS VS e DT. A análise por classe de atividades mostra que o MBOSS VS é melhor que o DT para classificar as classes *walking* e *jogging*, que reforça a compreensão de que a Acurácia é favorecida pelo desbalanceamento da base WISDM. Por outro lado, o classificador DT é melhor para classificar as atividades estáticas (*sitting* e *standing*). Para as demais atividades, ambos os classificadores não apresentam boas taxas de classificação.

As Figuras 6.12, 6.13 e 6.14 apresentam as matrizes de confusão para os classificadores MBOSS VS e SVM. Para a base UCI HAR, o MBOSS VS mostra um mau desempenho em relação ao SVM para classificar as atividades estáticas como sentar, ficar em pé e deitar. Analisando as instâncias, verificou-se que as atividades estáticas geram representações com MBOSS similires, o que revela que a representação simbólica para esta configuração não foi capaz de evidenciar pardões que favoreçam a discriminação dessas atividades. Por outro lado, o MBOSS VS apresentam a melhor acurácia para classificar atividades dinâmicas como caminhar, subir e descer escadas. Esses ganhos mostram que o MBOSS é uma representação eficaz para classificar atividades dinâmicas. Para a base UniMiB SHAR, o MBOSS VS mostra um desempenho superior de classificação em relação ao SVM, principalmente, nas atividades de transição, como ficar em pé a partir de sentado (*StandingUpFS*) e ficar em pé a partir de deitado (*StandingUpFL*), e nas atividades dinâmicas, como subir e descer escadas (*GoingUpS* e *GoingDownS*) e pular (J*umping*), e na atividade estática sentar (*SittingDown*). Motivos para estes ganhos de classificação do MBOSS VS estão fortemente relacionados a configuração de segmentação implementada na base de dados UniMib SHAR. Verificou-se que uma segmentação centrada em um pico na magnitude dos sinais do acêlerometro beneficia a representação simbólica, produzindo padrões que discriminam melhor as classes de atividades.

6.4.3 Discussão

Para analisar o desempenho dos classificadores avaliados de modo geral, diagramas de caixa foram construídos resumindo os resultados obtidos para as três bases de dados. Esses diagramas são apresentados nas Figuras 6.15 e 6.16, respectivamente, para o modelo personalizado e generalizado.

Os resultados revelam que o MBOSS VS é equivalente em desempenho em relação aos classificadores baseados na abordagem HF para aplicações que necessitam da construção de modelos pessoais de classificação de atividades, com medianas[3] 98,85% e

[3]A mediana é utilizada para melhor descrever o ponto médio dos resultados, uma vez que a média

97,91% para Acurácia e Macro Medida-F respectivamente. Para aplicações que necessitam de modelos impessoais, o MBOSS VS demostra uma capacidade de generalização mais estável que os classificadores adversários, com medianas 83,34% e 73,91% para Acurácia e Macro Medida-F respectivamente. As caixas do MBOSS VS são menores em relação aos demais, o que sinaliza uma variação menor no desempenho do classificador.

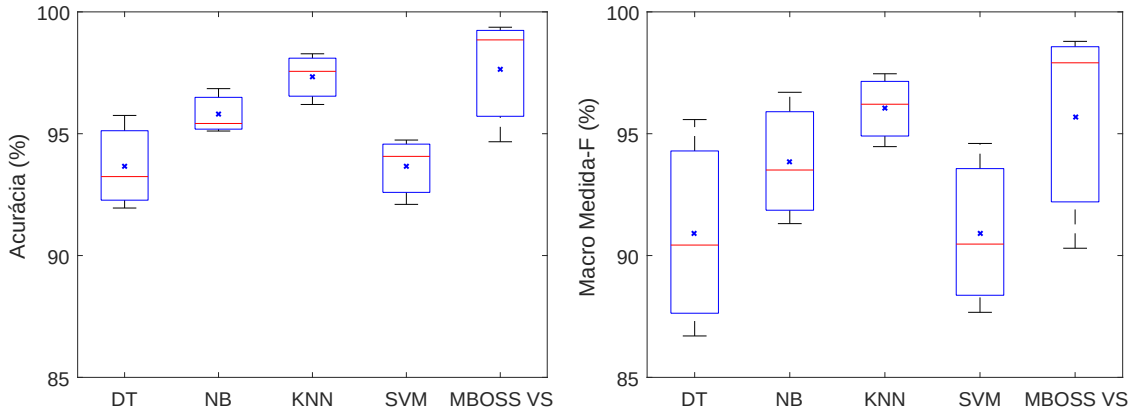


Figura 6.15. Diagramas de caixa para os resultados experimentais do modelo personalizado.

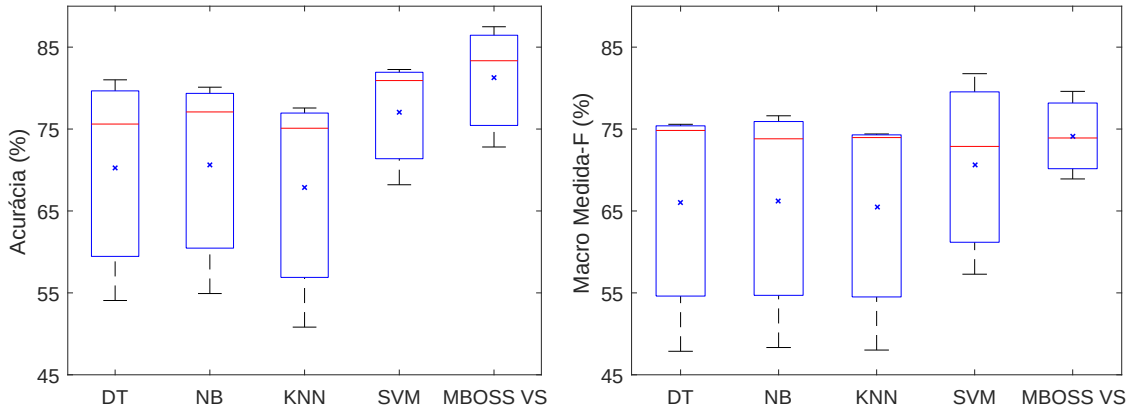


Figura 6.16. Diagramas de caixa para os resultados experimentais do modelo generalizado.

Em síntese, conclui-se que o método proposto é capaz de reconhecer atividades humanas e apresentam boa eficácia de classificação. Nos experimentos foi explorado diferentes configurações experimentais e verificado que o MBOSS VS é mais estável as mudanças, o que sinaliza que o método proposto, em geral, é estável e se adapatar a diferentes domínios de dados. Entre as diferentes configurações avaliadas, a melhor

pode ser distorcida por alguns poucos valores discrepantes.

configuração para representação simbólica é verificada na base de dados UniMiB SHAR. Observa-se que é possível classificar uma quantidade superior de classes de atividades quando é utilizada uma técnica de segmentação centrada no pico da magnitude. Para as demais configurações, o MBOSS VS é capaz de classificar as atividades, entretanto, dificuldades surgem na classificação de atividades estáticas (e.g., sentar, ficar em pé, deitar). Trabalhos futuros podem ser direcionados a melhorar a discriminação dessas atividades.

6.5 Resultados da Análise de Tempo e Espaço

Esta seção apresenta os resultados experimentais obtidos da análise de eficiência dos classificadores para RAH. Os resultados são divididos em duas partes, a primeira para a avaliação do tempo de computação, e a segunda para a avaliação de espaço de dados.

6.5.1 Tempo de Computação

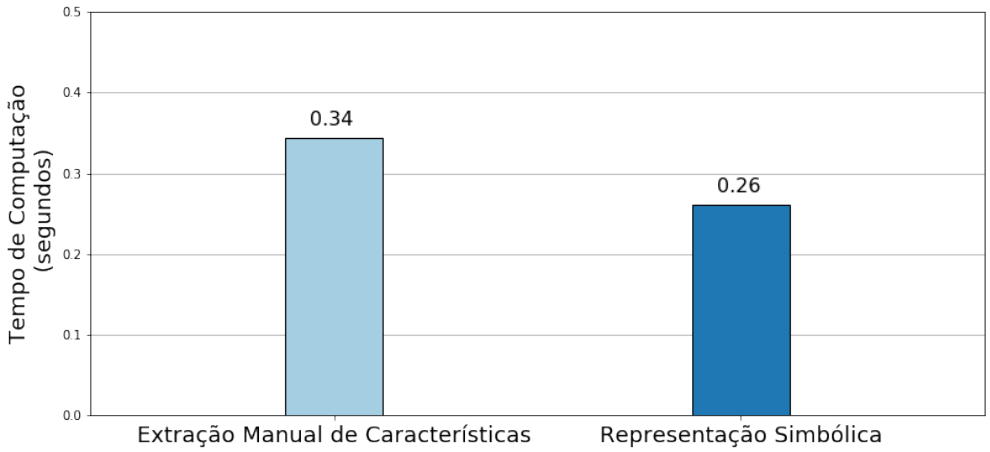


Figura 6.17. Comparativo de tempo de computação para extração de características.

A Figura 6.17 apresenta os tempos de computação, respectivamente, para extração manual de características no domínio do tempo e frequência descrita na Seção 6.2.2 e para a representação simbólica do MBOSS VS. Na figura, os dados mostram que a representação simbólica de fato tem tempo de computação inferior a abordagem manual de extração de características, com uma diferença de 0,08 segundos (23,5%), para a execução sobre o conjunto de teste (2947 instâncias) por completo. Neste caso,

as características extraídas no domínio do discreto do MBOSS VS apresentam uma eficiência superior a extração manual, até mesmo para um conjunto de características pequeno como o proposto neste trabalho.

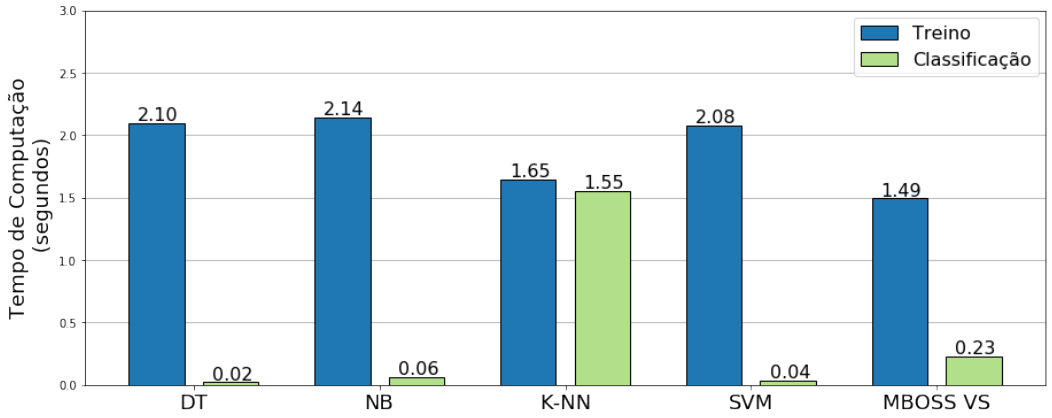


Figura 6.18. Comparativo de tempo de computação para o treino e classificação de atividades humanas.

A Figura 6.18 apresenta os tempos de computação para etapas de treino e classificação dos classificadores de RAH. Na figura, os dados revelam que o MBOSS é o classificador mais equilibrado em relação ao tempo requerido para treino. Neste caso, a redução de dimensionalidade aplicada pelo MBOSS acelerou o processo de treinamento do modelo de classificação. Para o teste, o classificador MBOSS obteve uma taxa intermediária, entre o tempo do classificador K-NN e os demais classificadores.

6.5.2 Espaço de Dados

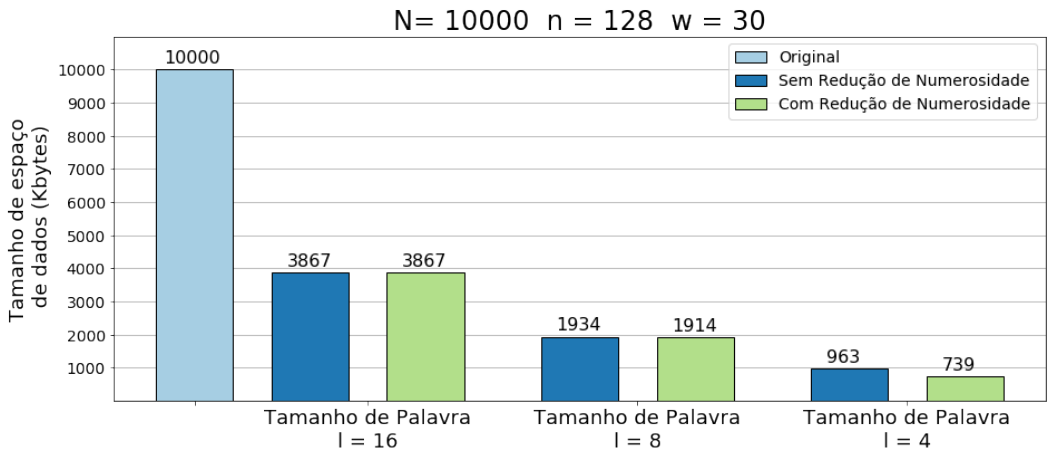


Figura 6.19. Comparativo do custo de espaço em mémoria necessario para o processamento de séries temporais.

A Figura 6.19 apresenta os resultados de espaço obtidos em *Kbytes*. A transformação simbólica foi executada com os seguintes parâmetros: tamanho do alfabeto $c = 4$, tamanho de janela $w = 30$ e tamanhos de palavra $l = \{4, 8, 16\}$.

Na figura, os dados mostram que a representação simbólica reduz a quantidade de espaço requerido na ordem de 60% a 90% sobre a quantidade original. Além disso, verifica-se que o tamanho de palavra tem influencia direta na redução de dimensionalidade, na qual um tamanho de palavra pequeno compacta mais os dados. Além disso, aplicação da técnica de redução de numerosidade se mostrou capaz de reduzir a quantidade de espaço, entretanto, valores significantes de redução foram observadas em tamanhos de palavras pequenos, como 4 e 6.

6.6 Considerações Finais

Este capítulo apresentou os experimentos e os resultados obtidos da avaliação do método proposto no Capítulo 5 para reconhecer atividades humanas. Foram apresentados resultados de desempenho de classificação e eficiência no processamento dos dados. Os resultados de classificação mostram que o MBOSS VS foi capaz de classificar atividades para diferentes configurações experimentais e bases de dados de atividades. Os resultados de eficiência mostram que o MBOSS VS tem ganhos no tempo de computação em relação a abordagem de extração manual de características e na redução da quantidade de dados necessarios para processamento, o que indica que a solução proposta é viavel de implementação em dispositivos móveis.

O próximo capítulo apresentará as conclusões finais do trabalho, limitações e trabalhos futuros desta pesquisa.

Capítulo 7

Conclusões

Este trabalho apresentou o desenvolvimento de um método para reconhecer atividades humanas. Ao longo do processo de desenvolvimento, uma extensão do método de representação simbólica BOSS foi proposta para lidar com múltiplos sinais de sensores. Essa extensão, nomeada MBOSS, foi desenvolvida aplicando a técnica de fusão a nível de características, a qual possibilitou que uma representação única fosse gerada a partir dos histogramas gerados pela representação simbólica para cada série temporal de entrada do sistema.

Com a aplicação do MBOSS, este trabalho contribui com a área de pesquisa apresentando uma nova abordagem para extrair automaticamente características dos dados de sensores e gerar modelos de classificação de atividades compactos e eficientes para aplicação em dispositivos móveis. A representação simbólica introduzida pelo SFA é uma novidade na área de RAH. Dentres os trabalhos da literatura avaliados neste trabalho, esta é o primeiro trabalho a avaliar o SFA para reconhcer atividades humanas.

Experimentos foram realizados de forma a avaliar o desempenho do método para RAH para diferentes configurações experimentais e de aplicação. Os resultados foram comparados com classificadores projetados baseados na abordagem tradicional de extrair características de forma manual. Além disso, os classificadores foram submetidos a duas estratégias de avaliação, modelo personalizado e generalizado. Os experimentos demostraram a viabilidade de aplicação do método MBOSS VS para classificar atividades humanas sob diferentes configurações de aplicação. O desempenho obtido foi equivalente nas três bases de dados utilizadas para avaliação. Dentre os resultados, destaca-se o apresentado na base de dados UniMiB SHAR, na qual o MBOSS VS obteve uma acurácia de 99% e 87% para o modelo personalizado e generalizado, respectivamente. Na avaliação geral, conclui-se que a abordagem proposta é eficaz para o

problema RAH, flexível a mudanças de configurações e capaz de automatizar o projeto de sistemas RAH eficientes para implementação em *smartphones*.

7.1 Trabalhos futuros

Nesta pesquisa os experimentos foram conduzidos em condições totalmente offline. Dessa maneira, trabalhos futuros serão conduzidos a avaliar a implementação do MBOSS VS em *smartphones* e avaliar o desempenho de classificação de atividades em tempo real. O MBOSS VS foi projetado especificamente para o problema RAH. Entretanto, essa solução pode ser estendida a outros domínios ou generalizado para classificação de séries temporais multivariadas. Logo, uma avaliação experimental para outros domínios como classificação de sinais fisiológicos para diagnóstico de anomailas são possíveis contribuições futuras. Os resultados experimentais demostraram dificuldades de classificação do MBOSS VS em um grupo de atividades consideradas estáticas (e.g., sentar, ficar em pé, deitar). Logo, trabalhos futuros podem ser direcionados a explorar uma melhoria da representação simbólica adicionando novas informações. Essas novas informações podem ser características globais do sinal como média e variância, visto que essas características são ótimas para descriminar esse grupo de atividades.

Com relação às opções de técnicas de fusão de dados, este trabalho optou por aplicar a solução mais simples. Entretanto, novas opções podem ser exploradas, como a aplicação de métodos para apresentar padrões de correlação entre palavras ou a aplicação de técnicas de fusão a nível de decisão como, por exemplo, *ensemble classifiers*.

Com relação a manipulação de palavras, outras técnicas da área de recuperação de informação podem ser introduzidas ao método. Por exemplo, a implementação de bigramas para extrair informações das co-ocorrências de palavras na representação simbólica, que pode incrementar o desempenho de classificação da mesma maneira que ocorre na classificação de texto.

Por último, há muitas direções nas quais a pesquisa descrita nesta dissertação pode continuar. Há ainda mais quando se considera também a utilização de outros tipos de sensores ou grupo de atividades observadas. Pode-se dizer que este trabalho não chegou ao fim, mas ao início de uma pesquisa em andamento e futura sobre reconhecimento de atividades.

Referências Bibliográficas

Akhavian, R. & Behzadan, A. H. (2016). Smartphone-based construction workers' activity recognition and classification. *Automation in Construction*, 71:198--209.

Alsheikh, M. A.; Selim, A.; Niyato, D.; Doyle, L.; Lin, S. & Tan, H.-P. (2016). Deep Activity Recognition Models with Triaxial Accelerometers. Em *AAAI Workshop: Artificial Intelligence Applied to Assistive Technologies and Smart Environments*.

Anguita, D.; Ghio, A.; Oneto, L.; Llanas Parra, F. X. & Reyes Ortiz, J. L. (2013a). Energy efficient smartphone-based activity recognition using fixed-point arithmetic. *Journal of universal computer science*, 19(9):1295--1314.

Anguita, D.; Ghio, A.; Oneto, L.; Parra, X. & Reyes-Ortiz, J. L. (2013b). A Public Domain Dataset for Human Activity Recognition using Smartphones. Em *Proceedings of the 21th International European Symposium on Artificial Neural Networks, Computatuional Intelligence and Machine Learning*, pp. 437--442.

Bagnall, A.; Bostrom, A.; Large, J. & Lines, J. (2016). The great time series classification bake off: an experimental evaluation of recently proposed algorithms. *arXiv preprint arXiv:1602.01711*.

Banos, O.; Galvez, J.-M.; Damas, M.; Pomares, H. & Rojas, I. (2014). Window size impact in human activity recognition. *Sensors*, 14(4):6474--6499.

Benocci, M.; Bächlin, M.; Farella, E.; Roggen, D.; Benini, L. & Tröster, G. (2010). Wearable assistant for load monitoring: Recognition of on Body load placement from gait alterations. Em *4th International Conference on Pervasive Computing Technologies for Healthcare*, pp. 1--8. IEEE.

Bersch, S. D.; Azzi, D.; Khusainov, R.; Achumba, I. E. & Ries, J. (2014). Sensor data acquisition and processing parameters for human activity classification. *Sensors*, 14(3):4239--4270.

Bulling, A.; Blanke, U. & Schiele, B. (2014). A tutorial on human activity recognition using body-worn inertial sensors. *ACM Computing Surveys (CSUR)*, 46(3):33.

Catal, C.; Tufekci, S.; Pirmit, E. & Kocabag, G. (2015). On the use of ensemble of classifiers for accelerometer-based activity recognition. *Applied Soft Computing*, 37:1018--1022.

Chen, Y. & Xue, Y. (2015). A deep learning approach to human activity recognition based on single accelerometer. Em *IEEE International Conference on Systems, man, and cybernetics (SMC)*, pp. 1488--1492. IEEE.

Cochran, W. T.; Cooley, J. W.; Favin, D. L.; Helms, H. D.; Kaenel, R. A.; Lang, W. W.; Maling, G. C.; Nelson, D. E.; Rader, C. M. & Welch, P. D. (1967). What is the fast Fourier transform? *Proceedings of the IEEE*, 55(10):1664--1674.

Cook, D. J. & Krishnan, N. C. (2015). *Activity learning: discovering, recognizing, and predicting human behavior from sensor data*. John Wiley & Sons.

de Carvalho, S. R.; Cordeiro Filho, I.; de Resende, D. C. O.; Siravenha, A. C. Q.; Meiguins, B. S.; Debarba, H. G. & Gomes, B. D. (2017). A Novel Procedure for Classification of Early Human Actions from EEG Signals. Em *2017 Brazilian Conference on Intelligent Systems*, pp. 240--245. IEEE.

Ehatisham-ul Haq, M.; Azam, M. A.; Loo, J.; Shuang, K.; Islam, S.; Naeem, U. & Amin, Y. (2017). Authentication of smartphone users based on activity recognition and mobile sensing. *Sensors*, 17(9):2043.

Figo, D.; Diniz, P. C.; Ferreira, D. R. & Cardoso, J. M. P. (2010). Preprocessing techniques for context recognition from accelerometer data. *Personal and Ubiquitous Computing*, 14(7):645--662.

Gravina, R.; Alinia, P.; Ghasemzadeh, H. & Fortino, G. (2017). Multi-sensor fusion in body sensor networks: State-of-the-art and research challenges. *Information Fusion*, 35:68--80.

Hammerla, N. Y.; Halloran, S. & Ploetz, T. (2016). Deep, convolutional, and recurrent models for human activity recognition using wearables. *arXiv preprint arXiv:1604.08880*.

Ignatov, A. (2017). Real-time human activity recognition from accelerometer data using Convolutional Neural Networks. *Applied Soft Computing*, 62:915--922.

Kolosnjaji, B. & Eckert, C. (2015). Neural network-based user-independent physical activity recognition for mobile devices. Em *International Conference on Intelligent Data Engineering and Automated Learning*, pp. 378--386. Springer.

Kwapisz, J. R.; Weiss, G. M. & Moore, S. A. (2011). Activity recognition using cell phone accelerometers. *ACM SigKDD Explorations Newsletter*, 12(2):74--82.

Lane, N. D.; Georgiev, P. & Qendro, L. (2015). DeepEar: robust smartphone audio sensing in unconstrained acoustic environments using deep learning. Em *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 283--294. ACM.

Lane, N. D.; Miluzzo, E.; Lu, H.; Peebles, D.; Choudhury, T. & Campbell, A. T. (2010). A survey of mobile phone sensing. *IEEE Communications magazine*, 48(9).

Lara, O. D. & Labrador, M. a. (2013). A Survey on Human Activity Recognition using Wearable Sensors. *IEEE Communications Surveys & Tutorials*, 15(3):1192--1209.

Li, Y.; Shi, D.; Ding, B. & Liu, D. (2014). Unsupervised feature learning for human activity recognition using smartphone sensors. Em *Mining Intelligence and Knowledge Exploration*, pp. 99--107. Springer.

Lin, F.; Song, C.; Xu, X.; Cavuoto, L. & Xu, W. (2016). Sensing from the Bottom: Smart Insole Enabled Patient Handling Activity Recognition Through Manifold Learning. Em *Proceedings of the IEEE First International Conference on Connected Health: Applications, Systems and Engineering Technologies*, pp. 254--263.

Lin, J.; Keogh, E.; Wei, L. & Lonardi, S. (2007). Experiencing SAX: a novel symbolic representation of time series. *Data Mining and knowledge discovery*, 15(2):107--144.

Lin, J. & Li, Y. (2009). Finding structural similarity in time series data using bag-of-patterns representation. Em *Proceedings of the International Conference on Scientific and Statistical Database Management*, pp. 461--477.

Lockhart, J. W. & Weiss, G. M. (2014). Limitations with activity recognition methodology & data sets. Em *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, pp. 747--756. ACM.

Lu, H.; Pan, W.; Lane, N. D.; Choudhury, T. & Campbell, A. T. (2009). SoundSense: scalable sound sensing for people-centric applications on mobile phones. Em *Proceedings of the 7th International Conference on Mobile systems, Applications, and Services*, pp. 165--178.

Maekawa, T.; Nakai, D.; Ohara, K. & Namioka, Y. (2016). Toward Practical Factory Activity Recognition: Unsupervised Understanding of Repetitive Assembly Work in a Factory. Em *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '16, pp. 1088--1099, New York, NY, USA. ACM.

Mangai, U. G.; Samanta, S.; Das, S. & Chowdhury, P. R. (2010). A survey of decision fusion and feature fusion strategies for pattern classification. *IETE Technical review*, 27(4):293--307.

Micucci, D.; Mobilio, M. & Napoletano, P. (2017). UniMiB SHAR: A Dataset for Human Activity Recognition Using Acceleration Data from Smartphones. *Applied Sciences*, 7(10):1101.

Mousavi, A.; Sheikh Mohammad Zadeh, A.; Akbari, M. & Hunter, A. (2017). A New Ontology-Based Approach for Human Activity Recognition from GPS Data. *Journal of AI and Data Mining*, 5(2):197--210.

Mubashir, M.; Shao, L. & Seed, L. (2013). A survey on fall detection: Principles and approaches. *Neurocomputing*, 100:144--152.

Negi, T. & Bansal, V. (2005). Time series: Similarity search and its applications. Em *Proceedings of the International Conference on Systemics, Cybernetics and Informatics: ICSCI-04, Hyderabad, India*, pp. 528--533.

Nweke, H. F.; Teh, Y. W.; Al-garadi, M. A. & Alo, U. R. (2018). Deep Learning Algorithms for Human Activity Recognition using Mobile and Wearable Sensor Networks: State of the Art and Research Challenges. *Expert Systems with Applications*. ISSN 0957-4174.

Nyan, M.; Tay, F.; Seah, K. & Sitoh, Y. (2006). Classification of gait patterns in the time–frequency domain. *Journal of biomechanics*, 39(14):2647--2656.

Ortiz, J. L. R. (2015). *Smartphone-Based Human Activity Recognition*. Springer International Publishing.

Phillips, C. L.; Parr, J. M. & Riskin, E. A. (2013). *Signals, systems, and transforms*. Prentice Hall.

Plötz, T. & Guan, Y. (2018). Deep Learning for Human Activity Recognition in Mobile Computing. *Computer*, 51(5):50--59.

Ravi, D.; Wong, C.; Lo, B. & Yang, G.-Z. (2016). Deep learning for human activity recognition: A resource efficient implementation on low-power devices. Em *13th IEEE International Conference on Wearable and Implantable Body Sensor Networks (BSN)*, pp. 71--76. IEEE.

Ronao, C. A. & Cho, S.-B. (2016). Human activity recognition with smartphone sensors using deep learning neural networks. *Expert Systems with Applications*, 59:235--244.

Ronao, C. A. & Cho, S.-B. (2017). Recognizing human activities from smartphone sensors using hierarchical continuous hidden Markov models. *International Journal of Distributed Sensor Networks*, 13(1):1550147716683687.

Ronaoo, C. A. & Cho, S.-B. (2015). Evaluation of deep convolutional neural network architectures for human activity recognition with smartphone sensors. Em *In Proceedings of the KIISE Korea Computer Congress*, pp. 858--860.

Salton, G.; Wong, A. & Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613--620.

Sazonov, E.; Metcalfe, K.; Lopez-Meyer, P. & Tiffany, S. (2011). RF hand gesture sensor for monitoring of cigarette smoking. Em *Fifth International Conference on Sensing Technology (ICST)*, pp. 426--430. IEEE.

Schäfer, P. (2016). Scalable time series classification. *Data Mining and Knowledge Discovery*, 30(5):1273--1298.

Schäfer, P. & Högqvist, M. (2012). SFA: a symbolic fourier approximation and index for similarity search in high dimensional datasets. Em *Proceedings of the 15th International Conference on Extending Database Technology*, pp. 516--527.

Sekine, M.; Tamura, T.; Togawa, T. & Fukui, Y. (2000). Classification of waist-acceleration signals in a continuous walking record. *Medical Engineering and Physics*, 22(4):285--291.

Senin, P. & Malinchik, S. (2013). SAX-VSM: Interpretable time series classification using sax and vector space model. Em *Proceedings of the IEEE 13th International Conference on Data Mining*, pp. 1175--1180.

Seto, S.; Zhang, W. & Zhou, Y. (2015). Multivariate time series classification using dynamic time warping template selection for human activity recognition. Em *Computational Intelligence, 2015 IEEE Symposium Series on*, pp. 1399--1406. IEEE.

Shoaib, M.; Bosch, S.; Incel, O.; Scholten, H. & Havinga, P. (2014). Fusion of Smartphone Motion Sensors for Physical Activity Recognition. *Sensors*, 14(6):10146--10176.

Shoaib, M.; Bosch, S.; Incel, O. D.; Scholten, H. & Havinga, P. J. (2015). A survey of online activity recognition using mobile phones. *Sensors*, 15(1):2059--2085.

Shoaib, M.; Bosch, S.; Incel, O. D.; Scholten, H. & Havinga, P. J. (2016). Complex human activity recognition using smartphone and wrist-worn motion sensors. *Sensors*, 16(4):426.

Siirtola, P.; Koskimäki, H.; Huikari, V.; Laurinen, P. & Röning, J. (2011). Improving the classification accuracy of streaming data using SAX similarity features. *Pattern Recognition Letters*, 32(13):1659--1668.

Silva, F. G. d. (2013). *Reconhecimento de movimentos humanos utilizando um acelerômetro e inteligência computacional.* Tese de doutorado, Universidade de São Paulo.

Sokolova, M. & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4):427--437.

Sousa, W.; Souto, E.; Rodrigres, J.; Sadarc, P.; Jalali, R. & El-Khatib, K. (2017). A Comparative Analysis of the Impact of Features on Human Activity Recognition with Smartphone Sensors. Em *Proceedings of the 23rd Brazillian Symposium on Multimedia and the Web*, pp. 397--404. ACM.

Sztyler, T. & Stuckenschmidt, H. (2016). On-body localization of wearable devices: an investigation of position-aware activity recognition. Em *Proceedings of the IEEE International Conference on Pervasive Computing and Communications*, pp. 1--9.

Tabachnick, B. & Fidell, L. (2012). *Using Multivariate Statistics.* Pearson Education.

Terzi, M.; Cenedese, A. & Susto, G. (2017). A multivariate symbolic approach to activity recognition for wearable applications. *IFAC-PapersOnLine*, 50(1):15865--15870.

Tian, Y. & Chen, W. (2016). MEMS-based human activity recognition using smartphone. Em *Control Conference (CCC), 2016 35th Chinese*, pp. 3984--3989. IEEE.

Walse, K. H.; Dharaskar, R. V. & Thakare, V. M. (2017). *A Study on the Effect of Adaptive Boosting on Performance of Classifiers for Human Activity Recognition*, pp. 419--429. Springer Singapore, Singapore.

Weiss, G. M.; Lockhart, J. W.; Pulickal, T. T.; McHugh, P. T.; Ronan, I. H. & Timko, J. L. (2016a). Actitracker: a smartphone-based activity recognition system for improving health and well-being. Em *Proceedings of the IEEE International Conference on Data Science and Advanced Analytics*, pp. 682--688.

Weiss, G. M.; Timko, J. L.; Gallagher, C. M.; Yoneda, K. & Schreiber, A. J. (2016b). Smartwatch-based activity recognition: A machine learning approach. Em *Proceedings of the IEEE International Conference on Biomedical and Health Informatics*, pp. 426--429.

Wilson, S. J. (2017). Data representation for time series data mining: time domain approaches. *Wiley Interdisciplinary Reviews: Computational Statistics*, 9(1).

Witten, I. H.; Frank, E.; Hall, M. A. & Pal, C. J. (2016). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Yu, J.; Qin, Z.; Wan, T. & Zhang, X. (2013). Feature integration analysis of bag-of-features model for image retrieval. *Neurocomputing*, 120:355--364.