Universidade Federal da Paraíba
Centro de Tecnologia
Programa de Pós-Graduação em Engenharia Mecânica
Doutorado

UTILIZAÇÃO DE ACELERÔMETROS EM UMA REDE DE SENSORES SEM FIOS DE ALTA QUALIDADE PARA MONITORAMENTO DE TRENS

por

EUDISLEY GOMES DOS ANJOS

Tese de Doutorado apresentada à Universidade Federal da Paraíba para obtenção do grau de Doutor

João Pessoa – Paraíba **Junho, 2018**

EUDISLEY GOMES DOS ANJOS

UTILIZAÇÃO DE ACELERÔMETROS EM UMA REDE DE SENSORES SEM FIOS DE ALTA QUALIDADE PARA MONITORAMENTO DE TRENS

Tese de doutorado submetida ao Programa de Pós-Graduação em Engenharia Mecânica do Centro de Tecnologia da Universidade Federal da Paraíba em cumprimento às exigências para obtenção do grau de Doutor em Ciências no Domínio da Engenharia Mecânica.

Orientador: Prof. Dr. Francisco Antônio Belo

João Pessoa – Paraíba, Brasil

Catalogação na publicação
Seção de Catalogação e Classificação

A599u Anjos, Eudisley Gomes Dos.
UTILIZAÇÃO DE ACELERÔMETROS EM UMA REDEDE SENSORES SEM FIOS DE ALTA QUALIDADE PARAMONITORAMENTO DE TRENS / Eudisley Gomes Dos Anjos. - João Pessoa, 2018.
95 f. : il.

Tese (Doutorado) - UFPB/CT.

1. Acelerômetro. 2. Telemetria. 3. Monitoramento ferroviário. 4. redes de sensores sem fios. 5. IEEE 802.15.4. 6. Inteligência Artificial. I. Título

UFPB/BC

UTILIZAÇÃO DE ACELERÔMETROS EM UMA REDE DE SENSORES SEM FIOS DE ALTA QUALIDADE PARA MONITORAMENTO DE TRENS

por

EUDISLEY GOMES DOS ANJOS

Tese aprovada em 20 de junho de 2018

Prof. Dr. FRANCISCO ANTONIO BELO
Orientador

Prof. Dr. ABEL CAVALCANTE LIMA FILHO
Examinador Interno

Prof. Dr. MOISÉS DANTAS DOS SANTOS
Examinador Interno

Profa. Dra. DANIELLE ROUSY DIAS DA SILVA
Examinadora Externa

Prof. Dr. ROMULO CESAR CARVALHO DE ARAUJO
Examinador Externo

João Pessoa – PB - *junho/2018*

Agradecimentos

Primeiramente agradeço ao meu orientador professor Francisco Antônio Belo pela amizade, atenção e principalmente por sempre confiar no meu trabalho. Foi com ele que pude ter os melhores ensinamentos e exemplos a seguir em minha carreira acadêmia e vida pessoal.

Ao meu amigo de toda uma vida acadêmica Abel Cavalcante Lima Filho que sempre esteve junto e ajudou-me em todas as empreitadas que resolvi entrar.

Ao professor, amigo e companheiro de muitos risos e conversas Rômulo César Carvalho de Araújo que sempre me fez ver que os problemas eram menores que o esperado e incentivou-me a enfrentá-los com garra e com bons exemplos.

A professora e grande amiga Danielle Rousy Dias da Silva que sempre me deu forças para seguir na carreira acadêmia com o seu bom humor e sua bondade incansáveis.

À minha amada família, minhã mãe, irmãs e sobrinhos pela companhia, cuidados e apoio imensurável sempre presente.

Aos meus companheiros de pesquisa do RailBee, em especial Letícia Ismael, Joanacelle Bandeira, Iury Rogério, Jéssica Maciel, Rafael Praxedes, Jansepetrus Brasileiro e Luiz Lima pelos momentos vividos, os conhecimentos trocados e pela valiosa amizade.

A companheira de doutorado e pesquisa Tatiana Simões que sempre me deu forças e esteve presente em todos os momentos que precisei. Ao meu parceiro de pesquisas, vida pessoal e acadêmica e jogos de tabuleiro Iury Rogério pelo apoio incansável para alcançar os objetivos desejados.

Aos demais amigos que contribuíram direta ou indiretamente para execução e finalização deste trabalho.

Resumo

Atributos de qualidade como velocidade, confiabilidade, segurança e conveniência são os fatores essenciais e decisivos para a atratividade dos serviços prestados nos trens urbanos. Atender estes critérios requer ênfase em automação ferroviária e utilização ótima dos recursos disponíveis. Para que a automação seja realizada com sucesso, é necessário informação em tempo real de todos os trens de forma a controlá-los com eficiência e segurança. Este trabalho tem como objetivo a definição de métodos e técnicas de medição de aceleração de alta precisão e baixo custo através de acelerômetros em uma rede ZigBee para trens urbanos e a avaliação das informações destes dispositivos como forma de validar a possibilidade de criação de modelos de classificação. Essas técnicas devem proporcionar uma melhoria e finalização do projeto RailBee. O RailBee consiste de uma RSSF baseada no padrão IEEE 802.15.4, composta de módulos remotos embutidos nos veículos, roteadores fixos dispostos nas vias e módulos base para recepção dos dados. A utilização de um acelerômetro permite o acompanhamento e avaliação de diversas medidas, entre elas: velocidade, aceleração e posição, além de outras estimativas como: eficiência energética, características de condução e outras técnicas que auxiliam a manutenção preditiva. O monitoramento é realizado em tempo real através da Internet e abrange um grande conjunto de veículos que circulam em vias permanentes, aumentando a eficácia e segurança do tráfego e diminuindo custos de operação. A tecnologia utilizada garante que diversos atributos de qualidade sejam assegurados, tais como: alta disponibilidade, segurança na transmissão de dados através de protocolos padronizados, modularidade da arquitetura e facilidade de manutenção. Além disso, uma vez que este projeto apresenta uma maior autonomia, maior precisão e menor custo tanto de implantação como de manutenção, ele surge como uma alternativa inovadora frente aos atuais métodos empregados no mundo para monitoramento de trens urbanos, geralmente baseados em GPS, transmissão por satélite ou por rede celular. Como meio de comprovação das vantagens de tal tecnologia foram aplicadas técnicas e agoritmos de IA para demonstrar a viabilidade de criação de um modelo para classificação de instâncias de valores obtidos a partir do acelerômetro e giroscópio.

Palavras-chave: Acelerômetro, Telemetria, Monitoramento ferroviário, redes de sensores sem fios, IEEE 802.15.4, Inteligência Artificial.

Abstract

Quality attributes such as speed, reliability, safety and convenience are the essential and decisive factors for the attractiveness of the services provided in urban trains. Meeting these criteria requires an emphasis on rail automation and optimal utilization of available resources. In way to obtain success in automation, real-time information is required from all trains in order to control them efficiently and safely. This work aims at the definition of methods and techniques of highly precise and low cost measurement of acceleration using accelerometers in a ZigBee network for urban trains, and, the evaluation of the information of these devices as a way to validate the possibility of creation of classification models. These techniques should provide an improvement and finalization of the RailBee project. The RailBee consists of a WSN based on the IEEE 802.15.4 standard, composed of embedded remote modules in the vehicles, fixed routers arranged in the roads and base modules for reception of the data. The use of an accelerometer allows the monitoring and evaluation of several measures, among them: speed, acceleration and position, besides other estimates such as: energy efficiency, driver profiles and other techniques that help predictive maintenance. Monitoring takes place in real-time over the Internet and covers a large number of vehicles traveling on permanent roads, increasing traffic efficiency and safety and reducing operating costs. The technology used provides that several quality attributes are ensured, such as: high availability, security in data transmission through standardized protocols, modularity of the architecture and easiness of maintenance. In addition, since this project presents a greater autonomy, greater precision and lower cost of both implantation and maintenance, it emerges as an innovative alternative to the current methods used in the world for urban trains monitoring, generally based on GPS, transmission by satellite or by cellular network. As a means of proving the advantages of such technology, IA techniques and algorithms were applied to demonstrate the feasibility of creating a model to classify instances of values obtained from the accelerometers and gyroscopes.

Keywords: Accelerometer, Telemetry, rail monitoring, wireless sensor networks, IEEE 802.15.4, Maintenance.

Sumário

Lista de Figuras

Lista de Tabelas

CAPÍTULO 1

INTRODUÇÃO

1.1 DEFINIÇÃO DO PROBLEMA

No segmento do transporte ferroviário, a falta de informações sobre os trens em movimento torna o trabalho dos engenheiros e técnicos mais difícil e impede que decisões importantes envolvendo o tráfego e a manutenção sejam tomadas com eficiência. Atualmente, grande parte dos métodos convencionais de manutenção e monitoramento de trens são baseados em inspeções periódicas das unidades e vias (Lidén, 2015). Na maioria das vezes essas inspeções não são suficientes para prevenir falhas de operação, levando a altos custos de manutenção e reduzindo a qualidade geral do sistema e dos serviços prestados (Aboelela et al., 2006).

A maioria dos sistemas de monitoramento de trens propostos na literatura empregam tecnologias de alto custo e/ou muito limitadas, restringindo-se à medição de poucas variáveis, em geral, velocidade e posição (NK Das and Bhowmik, 2009). Dessa forma, ainda existe uma demanda muito grande de sistemas que permitam o monitoramento de trens em movimento de forma contínua e em tempo real. É preciso que esses sistemas forneçam informações detalhadas sobre o funcionamento dos trens e que sejam de baixo custo e flexíveis o bastante para adequar-se aos diversos tipos de veículos que requerem essa tecnologia, incluindo tanto os modelos mais antigos como novos.

O uso de acelerômetros tem se configurado com uma alternativa inovadora para monitoramento, inferência e análise de diversas informações de veículos. O acelerômetro é um transdutor eletrônico que devido a sua alta aplicabilidade tem sido utilizado em diversos projetos de inovação tecnológica (Ataei et al., 2016; Kouroussis et al., 2016). O sensor, ao ser colado em um corpo, mede sua aceleração, permitindo a obtenção de diversos parâmetros tais como velocidade e distância. Particularmente na área de transportes metroferroviários os acelerômetros permitem o aumento da precisão de valores monitorados, formas de locomoção dos trens, vibração de materiais, análises de balanço e movimento, entre outros.

É comum encontrar acelerômetros sendo utilizados para a previsão de quebras do veículo mas as potencialidades desta tecnologia vão muito além do que se imagina. Além isso,

tais vantagens aliam-se ao fato de que a facilidade de instalação dos acelerômetros e giroscópios permitem uma integração aos diversos modelos de transporte existentes, desde trens mais antigos, movidos a diesel, até os mais modernos. Neste contexto, este trabalho tem como foco o uso de acelerômetros e giroscópios para verificar a viabilidade do uso de inteligência artificial, mais especificamente da mineração de dados, como base para criação de modelos de medição de aceleração em trens urbanos através da integração destes dispositivos à uma rede de sensores sem fios (RSSF). O dispositivo desenvolvido como protótipo para validação de da hipótese foi acoplado a um Trem Unidade Elétrica (TUE) e conectado a um sistema sem fios de monitoramento de baixo custo, simples instalação e que permite o monitoramento de múltiplos dados. Mais especificamente, os resultados deste trabalho são relevantes para permitir que o uso de acelerômetros em uma RSSF seja aplicado no acompanhamento e suporte ao controle em tempo real de um conjunto de veículos que circulam em vias permanentes.

O sistema ao qual a tecnologia desenvolvida foi integrada, denominado RailBee, é baseado em um método original proposto por nossa equipe e que utilizara o padrão de comunicação IEEE 802.15.4. Esse padrão é designado para aplicações de redes de sensores sem fio e oferece comunicação com baixo consumo de energia e baixo custo, para aplicações de monitoramento e controle que não exijam grande largura de banda. Em comparação com outros padrões de comunicação sem fio, como o IEEE 802.11 (WiFi) e o IEEE 802.15.1 (Bluetooth), ele apresenta vantagens no que diz respeito ao consumo de energia, escalabilidade e menor tempo para adição de novos nós na rede (dos Santos et al., 2011).

1.2 MOTIVAÇÃO

Assim como em muitos sistemas metroferroviários, no Metrô da Cidade do Recife (METROREC), o monitoramento e manutenção são baseados em métodos de inspeção periódica. Tais métodos possuem várias limitações e podem não ser suficientes para evitar alguns defeitos operacionais (Aboelela et al., 2006). Além disso, se o intervalo de tempo entre as inspeções for relativamente longo, muitos defeitos podem não ser detectados até que os danos causados por eles sejam reconhecíveis ou tenham sérias consequências para o equipamento (Nejikovsky and Keller, 2000).

Portanto, a hipótese central deste trabalho é que podemos melhorar a manutenção preditiva de trens urbanos através do uso de um sistema de medição de sinais que inclui o uso de acelerômetros e giroscópios. O uso dos valores e estimativas feitas através da informação colhida pode dar ao centro de engenharia da manutenção um poder maior de decisão sobre os fatores que afetam a manutenção dos trens e principalmente, chamar atenção para anomalias e mudanças em tempo real que podem vir a ocorrer nos veículos e vias.

Além disso, ao ser comparado com dados reais de velocidade, posição, sentido e movimento do trem será possível utilizar os dados do acelerômetro para inferir um número ainda maior de informações em tempo real. Essas informações são de extrema importância para a

equipe de monitoramento no Centro de Controle. Atualmente deve-se utilizar de rádios, celulares ou outros meios menos precisos para obter essas informações. Não obstante, a possibilidade de cálculos e estimativas que podem ser feitos com os valores obtidos é imensa. Essas possibilidades serão melhor descritas ao longo deste trabalho.

1.3 OBJETIVOS E CONTRIBUIÇÕES

1.3.1 Objetivos Gerais

O objetivo geral desta tese consiste na análise e avaliação de dados de acelerômetros e giroscópios instalados em TUE para que se possa inferir através de técnicas de inteligência artificial possíveis direcionamentos e melhorias no monitoramento de trens. Para isso, o projeto contou com o desenvolvimento e implementação de um sistema de hardware e software que tem como base uma rede de sensores sem fios de alta qualidade. Este sistema deve integrar-se ao projeto RailBee que tem como intuito a utilização de uma RSSF para automação de trens urbanos e que encontra-se em expansão.

A qualidade gerada através do RailBee abrange tanto a qualidade do sistema implementado nos trens como a qualidade na gestão dos serviços oferecidos aos usuários finais. Além do avanço tecnológico, este trabalho visa contribuir para área de pesquisa altamente qualificada por meio de estudos, publicação de artigos, manuais, aplicativos e especialização de pessoal. Espera-se que com essa nova tecnologia, aqui desenvolvida, o projeto RailBee mais completo e mais próximo de ser direcionado à etapa de elaboração e criação de um produto final.

1.3.2 Objetivos Específicos

Como objetivos específicos do projeto podemos citar:

- Mapeamento sistemático sobre a área abordada para verificar o estado atual de pesquisas semelhantes ao redor do mundo, contribuindo como base teórica para a pesquisa e levantando pontos-chave relevantes no estado da arte, bem como definindo diversos nichos de pesquisa da área;
- Levantamento de todas as medidas que podem ser obtidas através do acelerômetro e giroscópio;
- Estudo e utilização de um giroscópio para aumentar a precisão dos dados;
- Desenvolvimento de um protótipo para testes locais e remotos;
- Criação de uma base de dados geográfica que contemplará uma grande quantidade de dados, permitindo a integração em tempo real para monitoramento, análise e estudos estatísticos.
- Desenvolvimento de um sistema de monitoramento com interfaces mais próximas do real para os controladores, gerentes e usuários finais;

- Incorporação de novas medidas no projeto inicial do RailBee ampliando as possibilidades de atuação do sistema. Além das medidas previstas no projeto inicial (posição, peso e a estimativa de passageiros em qualquer instante de qualquer veículo na via) o novo circuito pode incluir medidas tais como: medidores de alta precisão das condições ambientais, sinais elétricos, aceleração, velocidade, potência elétrica, torques e vibrações nos eixos e etc, aumentando a quantidade de dados obtidos e contribuindo para o aprimoramento na gestão e manutenção dos trens;
- Desenvolvimento de técnicas e protocolos para melhorar o desempenho de comunicação e aumentar a segurança dos dados da rede RailBee nos testes finais.
- Aplicar os algoritmos das classes escolhidas para mineração de dados e extração de padrões;
- Analisar os padrões e informações extraídos no processo de mineração;
- Direcionar novos caminhos de pesquisas e melhorias futuras para o sistema de monitoramento do MetroRec

1.4 APRESENTAÇÃO

Os capítulos restantes deste trabalho de estão estruturados da seguinte forma: no Capítulo 2 é possível entender melhor sobre os temas e áreas que são utilizados como base para este trabalho. O Capítulo 3 aborda parte dos diversos trabalhos que foram abordados durante o mapeamento sistemático e serviram como base para a elaboração do estado da arte. Já no Capítulo 4 a metodologia seguida para elaboração, desenvolvimento e testes deste trabalho é detalhada, bem como as ferramentas e materiais necessários para que se fosse possível atingir os objetivos iniciais. O protótipo desenvolvido baseando-se nos estudos realizados e nos testes em laboratório é apresentado no Capítulo 5. O Capítulo 6 aborda os resultados obtidos após a aplicação dos algoritmos de inteligência artifical que foram executados sobre os dados coletados a partir do protótipo e dos experientos realizados. Por fim, no capítulo 7, apresenta-se a concusão do trabalho, as limitações encontradas e novas hipóteses para trabalhos futuros.

CAPÍTULO 2

FUNDAMENTAÇÃO TEÓRICA

O refernecial teórico de um trabalho científico ajuda a identificar os limites para as generalizações em relação aos diversos aspectos do fenômeno estudado, bem como, especificar as características chave que influenciam o fenômeno de interesse e a necessidade de examinar tais características. Um referencial teórico bem definido é de extrema importância em trabalhos acadêmicos, principalmente porque proporciona as orientações necessárias para análise e interpretação dos dados coletados, uma vez que estes devem ser analisados em relação ao referencial existente.

Portanto, esta sessão consiste do embasamento teórico necessário para um melhor entendimento deste trabalho. Para isso, diversos trabalhos acadêmicos dentre livros, artigos, teses e relatórios técnicos foram lidos e analisados. Esta etapa foi de extrema importância para fazer com que o domínio na área fosse o maior possível e ajudasse na definição das hipóteses que norteiam este trabalho. Os principais temas trabalhados que são abordados e fundamentados aqui são: sistemas de tempo real, redes de sensores sem fios, protocolos, acelerômetros, giroscópios e mineração de dados.

2.1 Sistemas de Monitoramento em Tempo Real

Sistemas de alta confiabilidade são orientados pelas funcionalidades que os mesmos devem executar, bem como as falhas que os mesmos devem superar. Para isso, muitos sistemas adotam restrições dos prazos operacionais denominados de tempo real (Goodloe and Pike, 2010). Portanto, os sistemas computacionais de tempo real podem ser definidos como sistemas onde seus comportamentos não dependem somente dos resultados lógicos que são obtidos de técnicas computacionais, mas também do tempo físico de onde estes resultados são produzidos (Kopetz, 1997).

Dependendo de como as falhas afetam o funcionamento, os sistemas de tempo real (Real Time Systems ou RTS) podem ser classificados de diversas formas (Kopetz, 1997). A maneira mais comum encontrada na literatura é a classificação de sistemas de tempo real do tipo *soft* ou

hard. Nos sistemas soft um não atendimento de prazo pode ser tolerável dependendo da aplicação. Já os sistemas hard não devem violar os prazos estipulados ou, neste caso, inviabilizariam o funcionamento do sistema.

Para utilização de sistemas de tempo real faz-se necessário o uso de Real-Time Operating Systems (RTOS) que são sistemas operacionais específicos para estes tipos de sistemas. Os RTOS são utilizados para prover serviços previsíveis às aplicações e primitivas que permitam a implementação de políticas de escalonamento em tempo real, comunicação entre processos e monitoramento run-time (Penumuchu, 2007).

Neste trabalho o sistema desenvolvido para aquisição dos dados é do tipo *soft* e utiliza-se de sistemas operacionais e pilhas de protocolo de comunicação de tempo real. O uso de sistemas do tipo *soft* deve-se ao fato de que os dados devem ser recebidos apensas para análise e nenhuma atuação no sistema metroferroviário é necessária neste momento. Com isso, foi possível verificar que o monitoramento realizado pela central pode ser realizado em tempo real e que o sistema atende os requisitos necessários. Para uma melhor avaliação é preciso testes de estresse, disponibilidade e escalabilidade do sistema.

2.2 Redes de Sensores Sem Fios

Os avanços da última década nas áreas de sensores, microprocessadores e comunicações sem fios, estimulou a produção e o uso de sensores inteligentes em grande escala. Isso favoreceu a diminuição dos custos e o aumento de suas capacidades permitindo uma gama ainda maior de aplicações. Com isso, os ambientes inteligentes têm se configurado como o próximo passo para o desenvolvimento evolutivo em áreas como: construção civil, automação industrial, residencial, sistemas de automação de transportes e muitos outros. A dependência de informações do meio ambiente no qual o sistema atua leva a necessidade do uso dos sensores inteligentes capazes de captar dados do meio e comunicarem-se através de uma rede de dados, mais especificamente, uma rede de sensores sem fios (Lewis, 2005).

As redes de sensores sem fios (RSSF) diferem das redes sem fio convencionais em diversas aspectos. Essas redes normalmente possuem diversos elementos, chamados nós sensores, autônomos e distribuídos em uma única rede, com restrição de energia, capazes de adquirir dados de forma rápida, confiável e precisa. Além disso, as RSSF devem ser autoconfiguráveis e adaptar-se à possíveis situações inesperadas na rede (Nakamura et al., 2007).

As RSSF podem ser vistas como uma área formada de duas características independentes: miniaturização e interconexão (Crnjin, 2011). Tais características são devido as dimensões dos componentes utilizados na composição dos nós. Os nós utilizados funcionam com baixa frequência e com memória limitada e se conectam através de uma rede parcialmente ou completamente sem fios, permitindo maior liberdade na distribuição dos sensores.

Esses nós podem ser utilizados de forma móvel ou estática e servem para colher informações e enviá-las a um nó base (também conhecido como sink ou coordenador). Essa informação

pode ser transmitida tanto diretamente do nó sensor ao base como através de roteadores, conhecidos como gateways. A Figura 2.1 mostra um modelo básico de uma rede de sensores sem fios com nós sensores, roteadores e nó base. Em geral os nós são capazes apenas de capturar e gravar dados obtidos através do monitoramento do ambiente ao seu redor, como temperatura, pressão, umidade, aceleração, etc. Além disso possuem capacidade limitada de processamento, necessitando de unidades externas para manipulação e exibição dos dados.

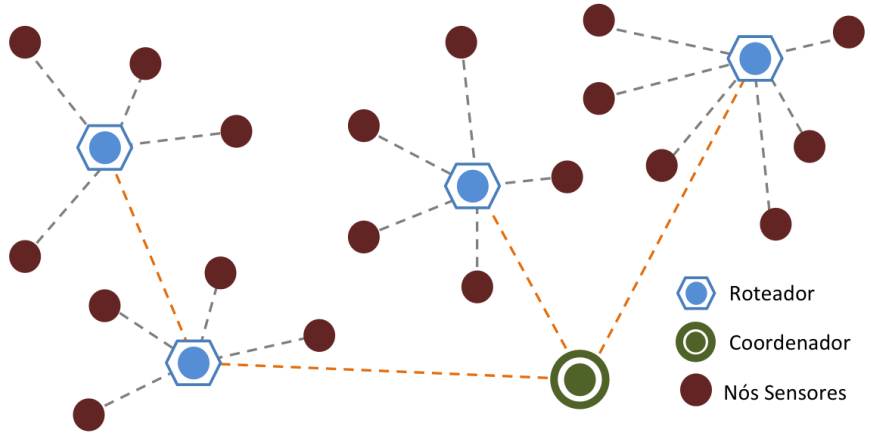


Figura 2.1 - Modelo de uma rede de sensores sem fio. Fonte: Dados produzidos pelo autor.

Assim como as redes cabeadas, as RSSF podem utilizar diversas topologias de organização, física ou lógica, gerenciando as interligações entre os nós da rede. Quando se trata de redes de sensores sem fios, a topologia pode ser altamente variável devido a limitação da energia, processamento, obstáculos e alcance.

As principais topologias utilizadas para este tipo de redes são: estrela, árvore, malha e híbrida. Uma rede estrela (Star) é uma topologia em que uma única estação base pode enviar e/ou receber mensagens para um número de nós remotos ao seu redor. Os nós sensores só podem comunicar-se com a estações base, não permitindo comunicação entre eles. A principal vantagem desse modelo é a simplicidade e o consumo mínimo de energia do nó sensor. Porém, como desvantagem, o nó base fica ao alcance de todos os nós, podendo aumentar o tráfego de dados e centralizando o gerenciamento e responsabilidade da rede em um único nó (Yang, 2013).

O modelo em árvore de agrupamento, também conhecido como clustering, é um tipo especial de rede peer-to-peer onde os nós sensores se conectam diretamente a nós coordenadores. Entretanto esses coordenadores podem ser locais e referentes a um cluster ou o coordenador geral da rede completa (sink). Ambos podem prover serviços de sincronização na rede. A principal vantagem inerente a esse tipo de rede é o alcance da área de cobertura, entretanto, a latência das mensagens pode ser bastante elevada (Cuomo et al., 2008).

A topologia em malha (também conhecida como Mesh) permite que qualquer nó da rede transmita a qualquer outro nó ao seu alcance. Isso possibilita a criação de redundância e maior escalabilidade na rede visto que na falha de um nó a rota de comunicação dos dados pode ser

modificada. Porém, esses tipos de redes possuem um alto custo energético quando os dados realizam múltiplos saltos para entrega da mensagem ao destinatário (Waharte et al., 2006)

Por fim, a topologia híbrida (hybrid star) proporciona comunicações robustas e versáteis. É considerada híbrida devido a possibilidade de junção de mais de um modelo mencionado anteriormente (Yang, 2013). Com isso, pode-se adaptar a rede às necessidades do projeto diminuindo o consumo energético ou selecionando nós específicos para realização de múltiplos saltos (multi-hop), embora essa característica tenha impacto no consumo (Geethapriya and Jawahar, 2013). A Figura 2.2 mostra um modelo simples de cada topologia mencionada anteriormente.

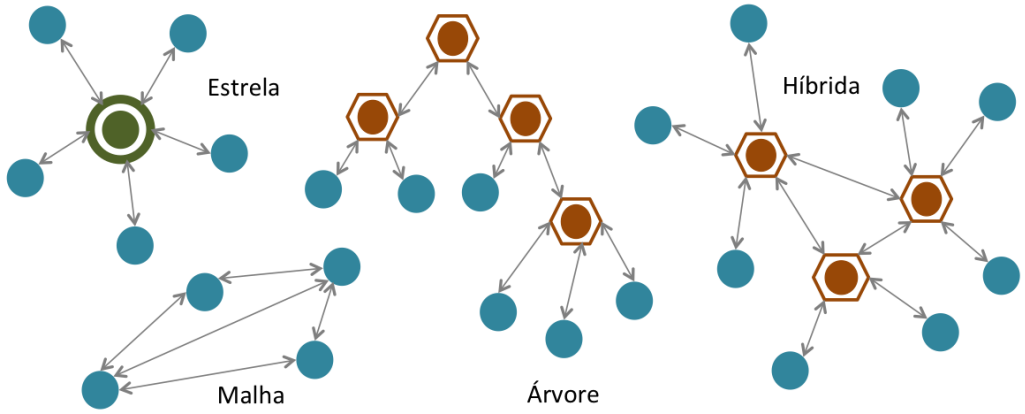


Figura 2.2 - Exemplos de topologias em RSSF. Fonte: Dados produzidos pelo autor.

Uma grande vantagem das características inerentes as RSSF é a possibilidade de uso dos nós acoplados à veículos e outros tipos de máquinas, uma vez que os sensores não necessitam estar conectados através de um componente físico (Culler and Hong, 2004). Portanto, um subconjunto das RSSF, as redes de sensores sem fio veiculares (Wireless Vehicular Sensor Networks - WVSN) desempenham um papel fundamental para garantir sensoriamento e processamento de dados em tempo real dentro de um veículo. Vários tipos de sensores de bordo podem produzir uma quantidade significativa de dados de detecção em tempo real, e estes dados necessitam ser transmitidos através de uma rede de sensores veicular eficiente e robusta. Os sensores podem estar posicionados tanto nos veículos como ao longo das estradas ou vias como postes, placas ou até mesmo no solo, criando uma rede de disseminação dos dados veiculares (Atanasovski and Gavrilovska, 2011).

De acordo com (Gil-Castineira et al., 2008) as aplicações mais comuns de WVSN utilizam os sensores para monitoramento de tráfego urbano e controle de segurança, como: previsão de acidentes, auxiliadores de mudança de vias, entre outros. Contudo, a ideia de utilizar-se veículos para aquisição de informações permite uma grande gama de coleta de medidas para auxiliar no monitoramento das mais diversas atividades, possibilitando sua aplicação em vários tipos de projetos.

As principais arquiteturas utilizadas nas redes de sensores sem fio veiculares são a veículo-infraestrutura (V2I), a infraestrutura-veículo (I2V) e a veículo-veículo (V2V). Essas arquiteturas podem ser mescladas para utilização em conjunto de forma a otimizar a aplicação. A Figura 2.3 mostra o funcionamento dos três tipos de arquiteturas mencionados.

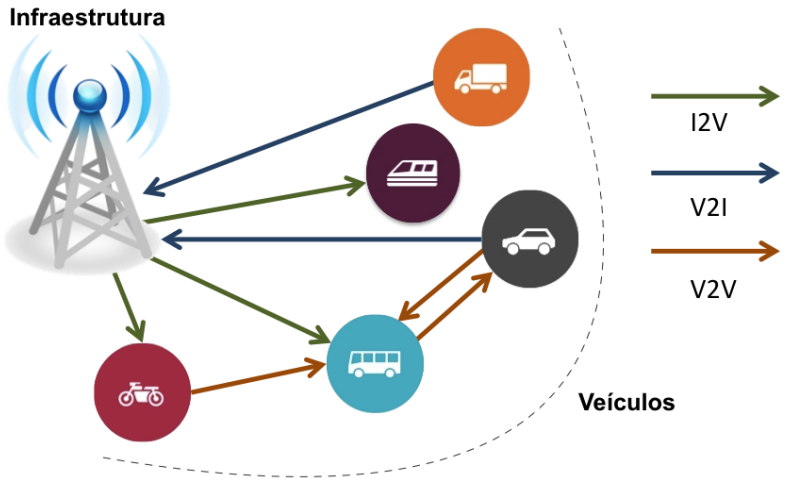


Figura 2.3 Possíveis arquiteturas utilizadas nas redes de sensores sem fios veiculares. Fonte: Dados produzidos pelo autor.

Um exemplo do uso da arquitetura V2I é a captura de informações nos veículos e envio para um servidor central. No modelo arquitetural I2V podemos enviar comandos para os sensores, visto que a comunicação se dá no sentido infraestrutura - veículo. Já a arquitetura V2V é normalmente utilizada como rota para captura de dados até a chegada no nó final (também chamado de sink ou nó base) que concentrará tais informações. Dependendo do modelo arquitetural, diferentes mecanismos de disseminação de dados podem ser utilizados (Atanasovski and Gavrilovska, 2011). Dentre eles podemos citar as técnicas de push, pull, flooding, relaying, entre outras, como pode ser visto em maiores detalhes no trabalho de (Atanasovski and Gavrilovska, 2011).

2.3 Protocolo ZigBee

Existe no mercado uma grande variedade de protocolos de radiofrequência (RF) para se trabalhar com redes de sensores sem fios. Alguns desses protocolos são exclusivos para revendedores específicos e outros são adotados como padrões industriais a ser seguidos. O protocolo ZigBee é considerado um padrão de alta qualidade utilizado na indústria para transmissão de dados o que motivou a pesquisa e uso neste trabalho. O ZigBee tem sua concepção a partir do protocolo IEEE 802.15.4 que tem sido largamente utilizado e continua em ascensão. Algumas informações sobre frequências utilizadas e também sobre largura de banda que ocupa serão expostas a seguir com o intuito de ampliar o entendimento sobre o funcionamento do protocolo em questão.

2.3.1 Características do protocolo ZigBee

O protocolo IEEE 802.15.4 foi criado com o intuito de se ter um padrão com uma baixa taxa de dados, conectividade simples e, principalmente, visando baixo consumo energético. A faixa de frequência regulamentada no Brasil para este padrão vai de 2,4000 GHz a 2,4835 GHz. Na teoria, a taxa de transmissão de dados deveria ser de 250kbps, porém, na prática, devido à sobrecarga do protocolo essa taxa é reduzida pela metade. O padrão IEEE 802.15.4 permite uma comunicação ponto-a-ponto ou ponto-a-multipontos. As aplicações mais comuns e menos escaláveis em termos de abrangência da área de cobertura são baseadas no modo estrela, mencionado anteriormente, onde existe um nó coordenador ao centro e nós finais distribuídos pelo ambiente.

O protocolo ZigBee foi projetado pela ZigBee Alliance, que é um grupo de mais de 300 empresas associadas de vários países que trabalham cooperativamente. O objetivo é construir um protocolo com o propósito de permitir confiança, baixo consumo de energia, segurança e custo efetivo em redes de sem fios baseadas em um padrão global e que possa ser utilizado por várias aplicações comerciais e industriais. Algumas aplicações do ZigBee podem ser vistas na Figura 2.4 a seguir.

Figura 2.4 - Áreas de aplicação do protocolo ZigBee. Fonte: adaptado de www.zigbee.org.

Apesar do protocolo ZigBee ser baseado no padrão IEEE 802.15.4, algumas soluções de roteamento e *networking* foram implementadas. O ZigBee foi projetado para permitir a adição de um nó a redes do tipo malha permitindo a conexão com rádios subjacente que também trabalhem com 802.15.4. A topologia de rede do tipo malha é geralmente utilizada para aplicações em que a distância entre dois pontos é superior ao alcance dos dois rádios localizados nesses

pontos. Como existem rádios intermediários nessa topologia, eles poderiam encaminhar essas mensagens para outros rádios da rede, aumentando o poder de ação da malha.

Outra característica interessante é que o protocolo ZigBee foi projetado para que rádios diferentes fossem implantados à rede de maneira automática sem a necessidade de intervenção manual. O próprio protocolo se responsabiliza por tratar dos reenvios, do roteamento dos dados e dos relatórios de recebimento. Além disso, também possui uma habilidade de "curar"a rede. Em outras palavras, se por um acaso um nó Y, que estiver entre os nós X e Z, for removido, o próprio protocolo cuida de interconectar os nós X e Z sem a necessidade do nó intermediário. Apesar disso, vale ressaltar que essa possibilidade depende do alcance dos rádios dos nós e que nem sempre essa cura é feita com sucesso.

Para que uma rede ZigBee seja montada, ela deve possuir os seguintes componentes: nós sensores finais, roteadores e coordenadores. Os roteadores também podem atuar como nós sensores finais já que o protocolo ZigBee utiliza o padrão IEEE 802.15.4 para definir as camadas Física (PHY) e de Acesso ao Meio (MAC) - *Medium Access Control*. Portanto, a frequência, o sinal da banda larga e as técnicas de modulação e demodulação são idênticas para todos os dispositivos da rede.Na Figura 2.5 é possível visualizar as camadas implementadas no protocolo ZigBee.

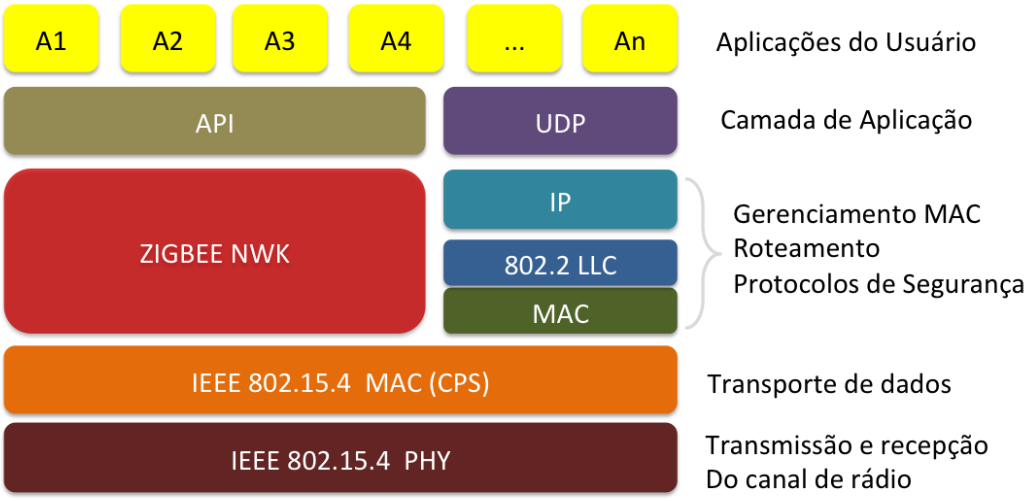


Figura 2.5 - Camadas do protocolo ZigBee. Fonte: adaptado de www.zigbee.org.

O protocolo ZigBee demonstra ser bastante interessante para implementação em aplicações de baixo consumo de energia, sistemas embarcados e em várias aplicações compromissadas com a confiabilidade e versatilidade. Vale lembrar que pelo fato da largura de banda não ser muito grande, a taxa de transmissão de dados tem um desempenho menor. Mesmo assim, esse desempenho é satisfatório para as aplicações que fazem uso do protocolo supracitado, especialmente o trabalho realizado no escopo desta tese.

2.4 Acelerômetros e Giroscópios

Fisicamente a aceleração pode ser definida como a medida de variação de velocidade pelo tempo. Os acelerômetros são dispositivos eletromecânicos que medem as forças de aceleração de um determinado corpo ou objeto. Tal medida pode ser feita sobre dois tipos de força: estática, como por exemplo a força da gravidade, ou dinâmica que é causada através da vibração e movimento do acelerômetro.

Os acelerômetros medem o que é chamado de aceleração própria (*proper acceleration*). Tal aceleração também pode ser conhecida como força g (*g-force*) visto que a mesma é medida em g, uma unidade que refere-se a constante gravitacional que é aproximadamente $9.81m/s^2$. A aceleração própria é um tipo de aceleração relativa que utiliza o valor 0 como referência. Assim, a gravitação e outras forças naturais não exercem efeito sobre esse tipo de aceleração.

Um acelerômetro parece um circuito pequeno e simples, mas possui vários componentes que trabalham de diversas maneiras. Este circuito pode trabalhar nos três eixos do plano cartesiano e é importante determinar quais os eixos que estão sendo utilizados para as aplicações. Os acelerômetros mais comuns usam apenas um eixo e são comumente utilizados para medir o nível de vibrações. Com a utilização de 2 eixos já é possível medir tanto vibração quanto aceleração nos eixos X e Y. Já os acelerômetros que utilizam 3 dimensões possuem a coordenada Z, permitindo medidas e localizações no espaço tridimensional. A Figura 2.6 mostra um diagrama de exemplo do circuito e eixos do acelerômetro.

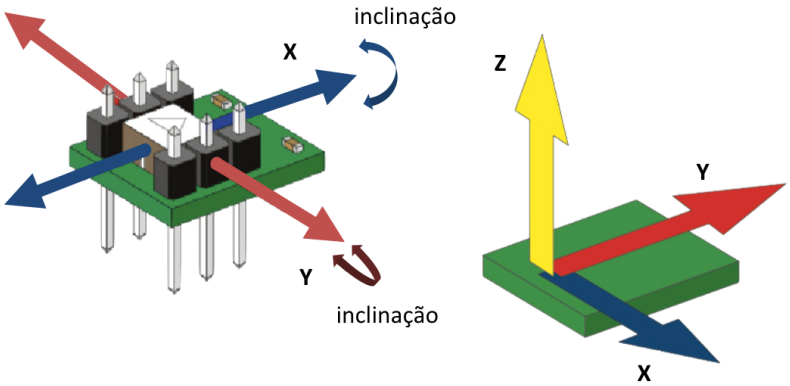


Figura 2.6 A figura ilustra um acelerômetro com dois eixos e um com três eixos. O acelerômetro de dois eixos pode inclusive medir a inclinação. Fonte: Parallax, Kerry Wong.

Há diversas tecnologias empregadas no desenvolvimento dos acelerômetros que podem ser selecionadas dependendo das necessidades do projeto. Os tipos mais comuns são os mecânicos seguidos pelos piezoelétricos, piezoresistivos, capacitivos e outros como: acelerômetros de fibra ótica, efeito hall, temperatura, etc. (André and Varum, 2013). A maioria dessas tecnologias seguem os padrões estabelecidos pela *BS IEC Semiconductor accelerometers* (BS IEC 60747-14-4, 2011), da *IEEE specification guide for single-axis non gyroscopic accelerometers*

(IEEE 1293, 2008) e da *MIL Aircraft accelerometers* (MIL-A-27261, 2011). Estes padrões determinam a forma de manufatura, testes e uso dos acelerômetros.

Por sua vez, os giroscópios foram desenvolvidos para substituir as conhecidas bússolas e utilizam o princípio da inercia como referncia para seu funcionamento. Giroscópios são fabricados incluindo estruturas rotacionais formadas por dois círculos articulados no qual o eixo de rotação fica livre, podendo assumir qualquer orientação (Passaro et al., 2017). Ao ser girado, a orientação do eixo não é modificada devido a conservação de seu momento angular.Um modelo pode ser visto na Figura 2.7. Dessa forma, este dispositivo pode ser utilizado para orientação, como por exemplo por embarcações, por aviões para funcionalidades como piloto automático e muito comunmente atualmente, seu uso em dispositivos móveis.

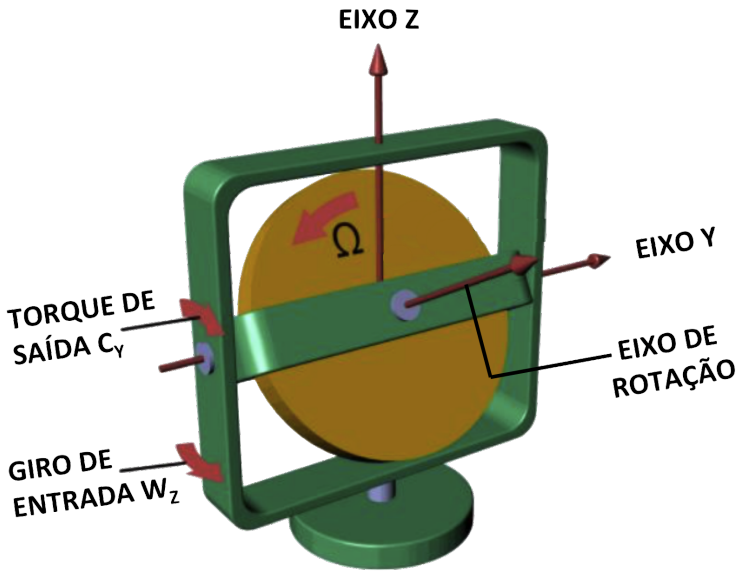


Figura 2.7 - Modelo de funcionamento de um giroscópio. Fonte: (Passaro et al., 2017)

Os rotores giratórios utilizados em giroscópios proveem a vantagem de produzir grandes sinais, fornecendo assim uma alta precisão nos dados (Peters et al., 2001). Contudo, os rolamentos utilizados podem impossibilitar a utilização de materiais de baixo custo disponíveis atualmente, principalmente porque podem se desgastar com facilidade (Hu et al., 2011). Mesmo assim, o uso de MEMS (sistemas micro elétricos mecânicos) tem barateado o custo no desenvolvimento de sensores de inércia, especificamente os giroscópios, principalmente em trabalhos de monitoramento, medições e análises em trens e linhas férreas como mencionado em(Heirich et al., 2011).

Para o projeto desenvolvido nesta tese utilizou-se um circuito chamado *Invensense* MPU-6050. Este circuito contém um acelerômetro MEMS e um giroscópio MEMS e conta com 16-bits de conversão analógico digital para cada canal. Portanto, ele captura X, Y e Z ao mesmo tempo e utiliza um barramento I2C para comunicações externas. O MPU-6050 não é caro, especialmente dado o fato de que ele combina tanto um acelerômetro e um giroscópio em um único chip.

2.5 Mineração de Dados

Atualmente, a maioria dos sistemas de software tendem a armazenar os dados obtidos, ou gerados por eles, em bancos de dados de forma que possam ser acessados sempre que necessário. Grande parte desse comportamento deve-se ao barateamento dos custos de armazenamento e a automatização das empresas (Rezende, 2003). Com isso, a quantidade de dados armazenados tem aumentado em uma velocidade muito alta, criando enormes bases de dados.

Esses dados e as informações obtidas e inferidas dos mesmos são considerados muito importantes e apreciados pelas organizações. Através dos dados e de processamentos computacionais é possível identificar perfis e padrões que não seriam identificados sem tal processamento (Rezende, 2003). Com a necessidade de extrair essas informações das bases de dados foi desenvolvido um processo para descobrimento de conhecimento em bases de dados, o KDD (*Knowledge Discovery in Databases*). O KDD consiste na geração de técnicas para a extração da informação e também para a descoberta de conhecimento (Araújo, 2009).

Existe na literatura uma divergência nas opiniões entre os conceitos de KDD e mineração de dados. Na mineração de dados o foco principal é de como transformar dados armazenados em conhecimento útil, expresso em termos de formalismos de representação, como regras e relações entre os dados. Já o KDD se preocupa com a extração de conhecimento previamente desconhecido, intrínseco e possivelmente útil (Rezende, 2003). Alguns autores consideram que os termos são sinônimos, enquanto outros consideram que a mineração é uma das etapas do processo de KDD, sendo a etapa mais importante do processo como descrito nos trabalhos de (Rezende, 2003), (Cornelius Junior, 2015) e (Fayyad et al., 1996a). Como este trabalho é focado na utilização de técnicas de mineração de dados os dois conceitos são abordados como sinônimos.

O processo de extração de conhecimento e informações de uma grande quanidade de dados é visto como um processo interativo e iterativo, pois é centrado na interação entre os usuários, especialistas do domínio dos dados e os responsáveis pela mineração. Existem diversas metodologias para a divisão de etapas durante o processamento da mineração, contudo o selecionado para ser utilizado como base nesta pesquisa é o processo sugerido em (Rezende, 2003), demonstrado na Figura 2.8, onde o processo de mineração é dividido em três etapas: pré-processamento, extração de padrões e pós-processamento. O modelo de processo apresentado ainda adiciona duas fases, uma anterior ao processo de mineração e uma após, sendo elas o conhecimento do domínio e a utilização do conhecimento adquirido.

Na fase de conhecimento do domínio são considerados aspectos relacionados aos objetivos da aplicação e a base de dados a ser estudada. É também nesta etapa que são identificados os dados da base que serão utilizados no processo de mineração, sendo uma alternativa também, a escolha de todos os dados da base (Rezende, 2003). A próxima etapa é a de pré-processamento onde o conjunto de dados escolhidos passa por um tratamento para que possa ser submetido aos métodos e ferramentas do processo de extração de padrões. Um dos processos dessa etapa é

Figura 2.8 - Etapas do processo de mineração de dados. Fonte: adaptado de (Rezende, 2003)

a limpeza de dados que contempla ações de verificação e limpeza de valores ausentes, valores fora do padrão e dados inconsistentes (Araújo, 2009), (Cornelius Junior, 2015).

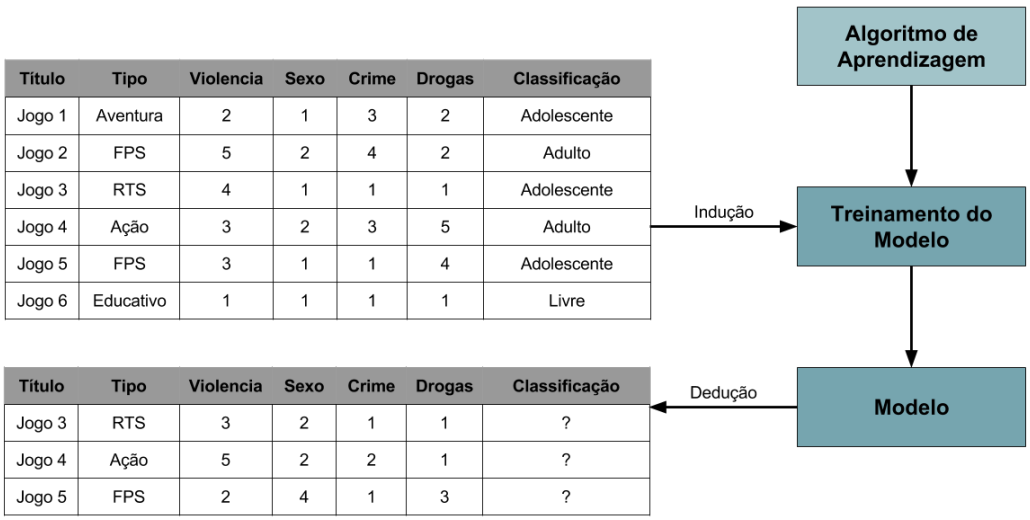
A etapa de extração de padrões é a etapa onde os algoritmos de mineração serão utilizados para a extração de conhecimento a partir dos dados. O resultado dessa etapa normalmente é composto de preditores, que a partir de um problema determinam a decisão (de Araújo, 2016). A última etapa do processo é denominada de pós-processamento, onde os conhecimentos obtidos serão avaliados quanto a sua qualidade e utilidade para a tomada de decisões. Como o processo é iterativo, é possível que todas as etapas ocorram várias vezes, pois qualquer mudança feita durante uma das etapas pode ser significativa para o processo (Cornelius Junior, 2015), (Fayyad et al., 1996a). Existe também a fase da utilização do conhecimento, que nada mais é que a fase do uso real do conhecimento obtido a partir de todo o processo de mineração, como uma informação útil para a tomada de decisões.

Como citado anteriormente os algoritmos utilizados durante a etapa de extração de conhecimento, são os responsáveis por encontrar padrões e informações dentro do grande volume de dados. Esses algoritmos podem ser classificados em diversas classes, sendo os algoritmos das classes de classificação, clusterização e associação os mais usados e explorados neste trabalho. A seguir, serão apresentados os principais conceitos dessas três classes de algoritmos.

2.5.1 Classificação

O processo de classificação está relacionado a associação ou classificação de objetos a uma determinada classe (Fayyad et al., 1996b). Dessa forma os algoritmos de classificação possuem duas etapas distintas: treinamento e classificação. Durante a execução do treinamento, um conjunto de dados de amostragem é utilizado. Estes dados já estão classificados de forma que o algoritmo passe por um aprendizado para construir um modelo que seja capaz de classificar o restante dos dados (Buss, 2011), (Araújo, 2009), (Cornelius Junior, 2015). Na segunda etapa, o modelo é então utilizado pelo classificador que terá função de receber dados e classificá-los a partir do modelo desenvolvido anteriormente. É necessário nesse momento a utilização de outras bases de dados, conhecidos como testes, para que seja possível estimar a precisão do classificador (Cornelius Junior, 2015). É importante ater-se ao uso de mais de uma base teste, pois, é necessário evitar-se o processo de *overfit*, onde o classificador se torna eficaz demais na classificação de uma base de dados, mas ineficaz para a classificação de outras (Buss, 2011). Alguns algoritmos utilizados para a classificação são as árvores de decisão e as regras de produção (Rezende, 2003).

Como exemplo, podemos imaginar um sistema de classificação de jogos em três faixas etárias de uso; livre, adolescente e adulto. Uma base de dados possui os dados do jogo e se este contém apologias à violência, sexo, crime e drogas, onde cada apologia é classificada entre um e cinco, sendo cinco a mais alta. Com base neste exemplo, a Figura 2.9 representa o processo de classificação de uma base de dados, partindo da indução do dados ao classificador até a dedução do mesmo (Cornelius Junior, 2015).



Título	Tipo	Violencia	Sexo	Crime	Drogas	Classificação
Jogo 1	Aventura	2	1	3	2	Adolescente
Jogo 2	FPS	5	2	4	2	Adulto
Jogo 3	RTS	4	1	1	1	Adolescente
Jogo 4	Ação	3	2	3	5	Adulto
Jogo 5	FPS	3	1	1	4	Adolescente
Jogo 6	Educativo	1	1	1	1	Livre

Título	Tipo	Violencia	Sexo	Crime	Drogas	Classificação
Jogo 3	RTS	3	2	1	1	?
Jogo 4	Ação	5	2	2	1	?
Jogo 5	FPS	2	4	1	3	?

Figura 2.9 Representação do processo de indução e dedução de um classificador. Fonte: Araújo, 2016

2.5.2 Clusterização

O processo de clusterização se baseia na premissa de que os dados não estão classificados e o algoritmo de clusterização deve agrupá-los a partir de suas semelhanças (Berkhin, 2006). Diferente da classificação, o processo de clusterização é feito sem a necessidade de um treinamento prévio, ou seja, o algoritmo não é supervisionado. A clusterização as vezes é uma alternativa às técnicas de classificação, principalmente quando se existe a necessidade de rotular e coletar informações para a criação dos grupos testes (Witten et al., 2016).

Os algoritmos de clusterização trabalham de forma interativa para formar os grupos, criando várias possibilidades e após um determinado número de interações, optar pela melhor formação. Subsequentemente o algoritmo escolhe diversos pontos de forma aleatória para representar os centróides dos grupos formados. A cada nova interação esses centróides são recalculados levando em consideração os elementos presentes dentro do grupo. Em seguida, os elementos são realocados para o grupo do centróide mais próximos. Ao final da execução, os dados estarão agrupados a partir das suas semelhanças, como mostrado na Figura 2.10 (Evandro Costa et al., 2013).

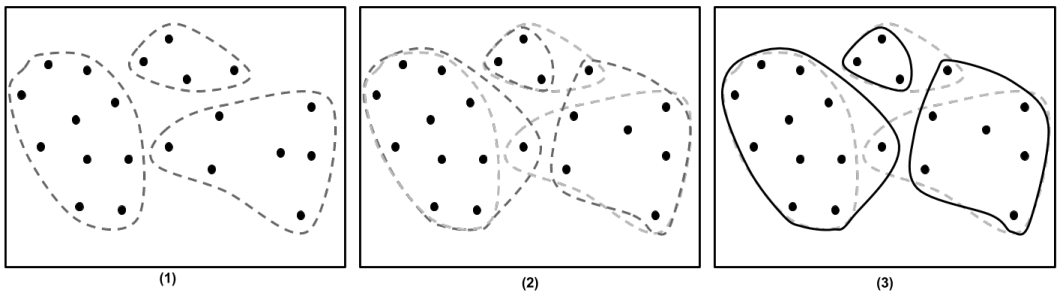


Figura 2.10 - Fases do processo de Clusterização. Fonte: adaptado de www.adobe.com

2.5.3 Associação

Por fim, o processo de associação, ou de regras de associação, está entre os processos mais utilizados na mineração de dados. A tarefa de associação tem como propósito encontrar regras, que mostram condições nos valores dos atributos com a finalidade de identificar padrões associativos. Tais padrões indicam quanto um grupo de atributos está relacionado a outro a partir da análise da relação entre seus itens (Cornelius Junior, 2015).

A associação pode ser utilizada em pesquisas baseadas em preferências, buscando pela afinidade entre os dados analisados. O principal objetivo desse processo é identificar conjuntos de itens ou eventos que acontecem juntos. Isto é possível baseando-se na teoria de que a presença de um item em uma determinada transação, implica diretamente na ocorrência de outro (Buss, 2011). Por exemplo um supermercado pode estar interessado em identificar qual produto é comprado por um cliente ao mesmo tempo que este compra um outro produto. Assim, é possível identificar quais produtos têm mais propensão a serem comprados juntos e determinar

medidas estratégicas para atrair os consumidores. A Figura 2.11 mostra um exemplo do roteiro mencionado.

Figura 2.11 - Modelo do processo de associação. Fonte: adaptado de www.brandidea.com

As regras de associação podem ser representadas como uma implicação lógica do tipo X -> Y, onde os conjuntos X e Y são de itens distintos (Rezende, 2003). Dessa forma os algoritmos de associação procuram induzir essas regras, utilizando regras de previsão, do tipo SE..ENTÃO, onde SE é a condição da regra e ENTÃO tende a prever algum atributo solicitado (Garcia, 2012). Dentre os algoritmos utilizados para esta técnica o mais comum é o algoritmo Apriori (Cornelius Junior, 2015). Este algoritmo itera um conjunto de *Item Sets*, conjunto de itens onde cada item é um par associado atributo-valor, procurando identificar associações que satisfaçam um mínimo de repetições (Pachiarotti, 2012).

CAPÍTULO 3

ESTADO DA ARTE

Atualmente os sistemas de monitoramento de trens mais modernos utilizados no mundo, empregam tecnologias de comunicação sem fio de longo alcance como GPS (Sistema de Posicionamento Global), GSM (Sistema Global de Telefonia Móvel) e a tecnologia por satélite Geographic Information Systems (GIS). No trabalho de (Nejikovsky and Keller, 2000) são utilizados receptores GPS combinados com transceptores sem fios que fornecem as coordenadas do trem para uma central de monitoramento, ao mesmo tempo em que sensores associados aos transceptores transmitem diferentes eventos relacionados ao trem em movimento.

No trabalho de (q. Ma et al., 2006) é proposto um sistema para monitoramento da velocidade de trens que integra tecnologias de sensores, redes Ethernet e GSM. O sistema utiliza uma rede sem fios e os dados de velocidade dos trens. Medidos por cada estação, os dados são enviados para a central de controle por meio de uma rede GSM. Já o trabalho de (Mirabadi et al., 1999) propõe um método denominado "Onboard Train Navigation System" (OTNS) que permite o diagnóstico de falhas e a determinação da posição de trens em uma via utilizando GPS, radar por efeito Doppler e outros sensores.

Um dos principais problemas que surge nos trabalhos citados anteriormente é que os sensores de GPS apresentam muitos problemas de localização, medição de velocidade e aceleração, devido a falta de precisão. Além disso, os sistemas propostos não mencionam redução das falhas de interferência eletromagnéticas, observadas em sistemas que utilizam receptores GPS, GSM e/ou GIS, ou por oclusões, quando o trem está trafegando em túneis, ou desastres climáticos como por exemplo em (Bertran and Delgado-Penin, 2004).

Em (Sharma and Vaidya, 2007) é proposto um sistema para monitoramento de vagões composto por etiquetas RFID acopladas aos trens e, leitores RFID instalados na ferrovia a distâncias fixas. Este sistema tem a vantagem do urso de RFID mas restringe-se ao monitoramento da posição dos trens. Além disso, os dados são enviados via rede celular no trem em movimento como mostrado no diagrama da Figura 3.1.

Através do uso de um acelerômetro pode-se obter valores de aceleração muito mais precisos e confiáveis. O acelerômetro é um transdutor eletrônico utilizado nas medições de

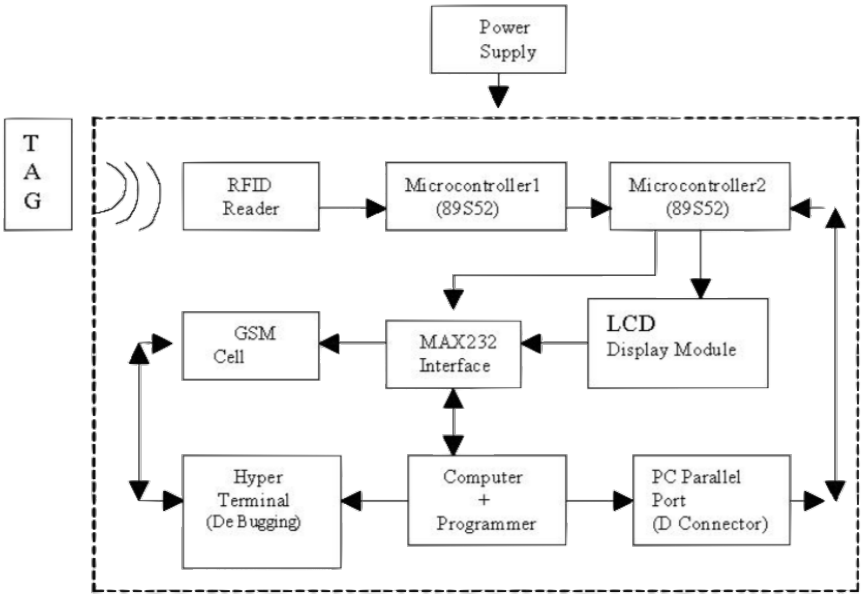


Figura 3.1 - Esquemático da arquitetura proposta por (Sharma and Vaidya, 2007).

movimentos de corpos ou objetos. Suas aplicações mais comuns são em vibrações, oscilações, movimentos sísmicos, choque, força e posição. Normalmente, o sensor se comporta com uma resposta elétrica a variação da posição do objeto em observação. Como exemplos, nos celulares, podem ser utilizados para determinar sua posição relativa e na indústria automobilística, através da terminação da força, para determinar vários parâmetros de desempenho dos veículos.

A medição através dos acelerômetros utiliza a unidade de medida linear g (translação). A quantidade g é é o valor de aceleração produzida pela força da gravidade terrestre. Posto que ela varia com a latitude e altitude, ela é padronizada para $9,80665m/seg^2$ e geralmente simplificado para $9,81m/seg^2$. A partir destes valores pode-se obter a aceleração angular (rotacional) através da unidade radiano por segundo quadrado (rad/seg^2). Por outro lado, para medidas de frequência utiliza-se a unidade Hertz. Nas especificações do acelerômetro tem-se a faixa em relação à g e faixa de frequência, além do seu peso e como deve ser encaixado no objeto. Em um exemplo simples da vantagem de utilização do acelerômetro em monitoramento de veículos, ao integrar-se o valor da aceleração tem-se a velocidade e ao integrar-se a velocidade tem-se a posição do veículo.

Com os avanços e melhorias nos acelerômetros, está cada vez mais comum encontrá-los nas mais diversas aplicações. Muitas delas utilizam os acelerômetros atrelados a redes de sensores sem fios para monitoramento de trens urbanos. No trabalho proposto por (Lai et al., 2012) o acelerômetro é utilizado como um mecanismo de adaptação de taxas de transmissão para aumentar a taxa de transferência e estabilizar o sinal wireless quando o trem chega ou sai da estação. Em outras palavras, os autores utilizam o acelerômetro principalmente para identificar qual o estado o trem encontra-se nas estações. Esses estados foram denominados de enter, stop, leave, start, e foram utilizado como uma forma de melhorar o gerenciamento dos dados transmitidos na rede. Um modelo do acesso mobile wifi pode ser visto na Figura 3.2.

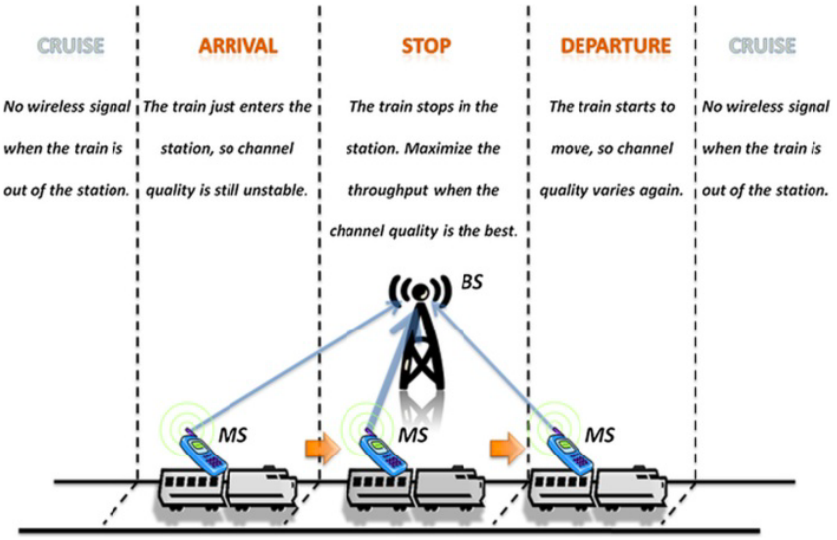


Figura 3.2 - Modelo da arquitetura do trabalho proposto em (Lai et al., 2012).

Outra utilização pode ser vista no trabalho proposto por (Stockx et al., 2014) chamado SubwayPS. Neste caso, os acelerômetros são utilizados para sanar um dos principais problemas mencionados anteriormente que a localização do trem submerso quando não há sinal de WiFi e GPS. Os autores utilizam a mesma técnica de posição que permite a localização em smartphones. Enquanto os trens estão no subsolo, partindo dos pontos de saída e chegada do trem, realiza-se uma aproximação do espaço percorrido através dos dados do acelerômetro e do giroscópio presentes nos aparelhos celulares. A Figura 3.3 mostra um exemplo de variabilidade diária medida na linha Piccadilly em Londres. Ambos os gráficos retratam os dados calculados pelo SubwayPS. As paradas A, B, e C são claramente visíveis. Estes gráficos ilustram um exemplo de como há muita variabilidade nas medições. As vezes mesmo entre as mesmas duas estações os valores podem variar muito (veja Track A-B) com uma diferença de quase um minuto. O período durante o qual um trem ainda está parado em uma estação pode também variar muito (ver B).

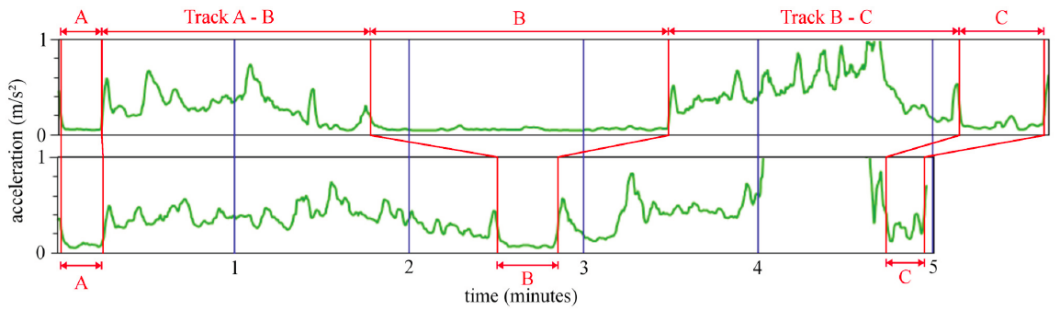


Figura 3.3 - Exemplo dos dados recebidos pelo SubwayPS Fonte: (Stockx et al., 2014).

O trabalho proposto por (Shah et al., 2014) também utiliza os acelerômetros dos smartphones para classificar o meio de transporte no qual se encontram. Além disso, o sistema também utiliza GPS e GIS. A classificação realizada pode ser: meios de transporte imóveis, veículos,

meios não motorizados e aleatórios. Os resultados demonstraram que os dados obtidos do GIS são úteis para classificar o modo de transporte, entretanto, os autores planejam utilizar as características das medições do acelerômetro para ajudar a diminuir as dúvidas e reduzir a latência de detecção. Além disso, o uso de acelerômetros provê uma redução no gasto energético do sistema inicial. Apesar da proposta ser interessante, ainda consta como trabalhos futuros desses autores.

Um uso bastante peculiar do acelerômetro foi proposto por (Eriksson et al., 2008). O trabalho desses autores propõe o uso das medidas obtidas pelos acelerômetros para detectar as condições de qualidade das superfícies em estradas. Os dispositivos foram acoplados juntos aos GPS de taxis e buscou avaliar as condições das estradas por onde esses veículos trafegavam. Os resultados mostraram que 90% dos casos detectados indicam anomalias na estrada, necessitando de reparos.

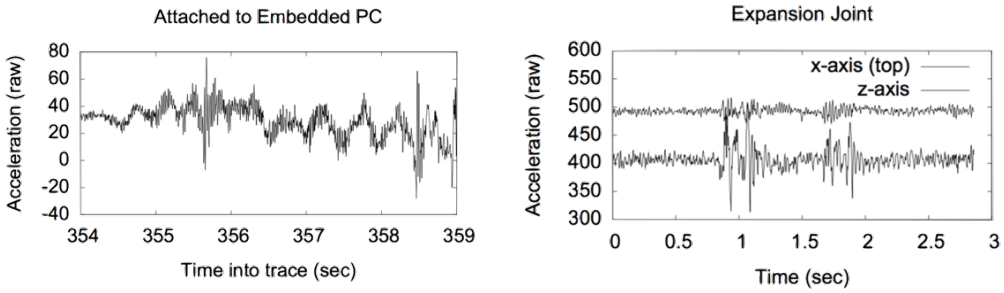


Figura 3.4 - Detecção de irregularidade em vias no trabalho de (Eriksson et al., 2008).

Vale ressaltar que os resultados obtidos são expansíveis e que podem ser adaptados para vias permanentes, como trilhos. Essa alternativa inclusive foi debatida anteriormente no trabalho de (Weston et al., 2006), que utiliza as vibrações geradas pelos acelerômetros acoplados na suspensão dos trens para o monitoramento da qualidade das linhas férreas. Devido a qualidade dos acelerômetros da época é notório que o mesmo projeto, se aplicado hoje, obterá uma precisão consideravelmente maior.

Esses resultados foram provados por (Bassetti et al., 2013) e (Tariq Abuhamdia and Davis, 2014). Em (Bassetti et al., 2013) é desenvolvido um sistema que utiliza o nó acelerômetro triaxial amplificado MEMS para obtenção de informações dos trens de carga. Com a falta de manutenção dos trilhos, principalmente daqueles nos quais trens de carga passam com cargas pesadas, a redução de qualidade pode ser catastrófica. O sistema realiza uma análise dos dados obtidos a fim propor meios de reduzir os danos causados nos trilhos.

Já o trabalho de (Tariq Abuhamdia and Davis, 2014) propõe uma solução mais prática e com menor custo. Um sistema de monitoramento montado no trem auxilia na detecção de falhas ferroviárias e trilhos quebrados. Os resultados mostram que os eixos x e y retornados pelo acelerômetro trazem informações úteis em casos de exceção, que ajudam a identificar possíveis defeitos nas vias.

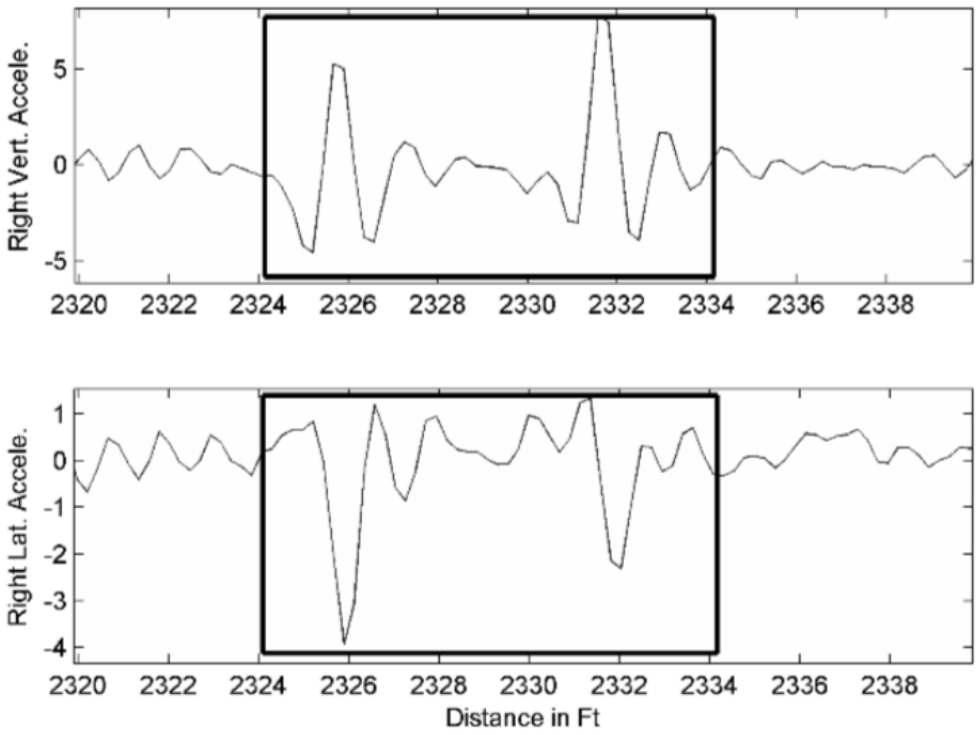


Figura 3.5 Aceleração vertical e lateral do lado direito para detecção de quebras de concreto nos trilhos. Fonte: (Tariq Abuhamdia and Davis, 2014).

Apesar dos diversos trabalhos comentados anteriormente utilizarem o acelerômetro, poucos utilizam uma rede sem fios bem definida e nenhum utiliza uma rede baseada no protocolo ZigBee para envio dos dados. O que é possível observar é que os protótipos desenvolvidos estão preocupados fortemente com os resultados locais do uso de acelerômetros mas não se preocuparam em como esses dados serão agrupados e monitorados em tempo real e com alta qualidade. Além disso, os trabalhos se limitam a analisar resultados que já eram previamente supostos e não buscam inferir novas informações a partir de técnicas de inteligência artificial a fim de detectar novos padrões e novas possibilidades de uso dos dados adiquiridos. Portanto, este trabalho visa o uso de uma rede ZigBee para a comunicação de dados (utilizada pelo projeto RailBee), devido ao baixo custo, baixo consumo energético e alta qualidade na transmissão, de modo que, de forma inovadora, a junção desta rede, o uso de acelerômetros e giroscópios forneçam importantes dados reais que ao ser analisados possa-se monitorar o estado atual do trem além de inferir-se novas informações sobre os mesmos.

O sistema RailBee utilizado neste trabalho utilizará diferentes sensores integrados a módulos IEEE 802.15.4 presentes nos trens em movimento. Estes módulos permitem o monitoramento de forma dinâmica de vários parâmetros dos veículos como condições do motor, localização na via, medidas de temperatura, além de detectar falhas operacionais como abertura da porta em movimento. Assim como em (Nejikovsky and Keller, 2000), o sistema apresenta a vantagem do monitoramento remoto de uma estação base que permite tomadas de decisões e

envio de sinais de controle baseado em dados de tráfego da via. Característica não observada em (Mirabadi et al., 1999), no qual as informações são disponibilizadas apenas no próprio trem em movimento, e em (Aboelela et al., 2006), (Delprete and Rosso, 2009) e (Browne et al., 2007) nos quais o monitoramento é feito por meio das linhas férreas.

O RailBee apresenta-se como uma solução de menor custo e maior flexibilidade do que as encontradas na literatura. Além disso, permite o monitoramento em tempo real dos parâmetros dos trens no METROREC, o que não acontece atualmente. O RailBee irá prover uma grande quantidade de informações valiosas, não se limitando a poucos parâmetros, como posição e velocidade, como nos sistemas descritos anteriormente. Trata-se de um sistema inovador, com domínio tecnológico nacional, com a titularidade do pedido de patente requerida em nome da UFPB e os inventores participantes deste projeto (Número do registro no INPI:, 01 dez. 2008, Patente PI0805974-8).

Um trabalho realizado com estudos iniciais do RailBee, ainda sem a implementação física da RSSF, foi laureado com o VI Prêmio Alstom de Tecnologia Metroferroviária, ano 2009 (Vencedor) e Menção Honrosa na V e VII edição do Prêmios Alstom de Tecnologia Metroferroviária, nos anos de 2008 e 2010, respectivamente. O Projeto também foi contemplado no Concurso Pró-Inovação Institutos Federais de Educação, Ciência e Tecnologia (CND/UNB). Os estudos científicos iniciais foram publicados em periódicos internacionais como (dos Santos et al., 2011) e em periódicos nacionais (Araujo, 2009; Araújo et al., 2010).

Além da alta qualidade dos dados adquiridos e a diminuição nos custos, para que o monitoramento de trens ocorra de forma ótima é necessário que os dados obtidos sejam analisados e que informações relevantes para a empresa/indústria estejam disponíveis. Em posse dessas informações, as empresas podem tomar melhores decisões de negócio e identificar as áreas mais frágeis de seu modelo de processos. Assim a área de inteligência artificial aliada à mineração de dados surge como um complemento ideal no estado da arte do uso de acelerômetros, RSSFs e monitoramento de trens.

Seguindo a tendência de encontrar padrões utilizando a mineração de dados, o trabalho desenvolvido em (Lipan and Groza, 2010) preocupa-se em recolher dados através de GPS implantados em ônibus que formam uma base de dados com os trechos percorridos por eles e as velocidades em determinado período de tempo. A base de dados é então minerada com o intuito de identificar padrões de tráfego que possam ajudar a criar planilhas de horários e rotas de ônibus mais precisos que os atuais já fornecidos pelas companhias gestoras.

O trabalho desenvolvido em (Ceapa et al., 2012) segue a premissa que os sistemas que criam rotas para o uso do transporte público não consideram a lotação de pessoas no transporte naquele momento, sendo preocupante pois eles só levam em consideração o tempo de movimentação do usuário do ponto A ao B sem levar em consideração o seu conforto. A partir dos dados recolhidos na cobrança de passagens em estações de metrô o trabalho utiliza a mineração de dados para descobrir padrões de lotação nas estações. Utilizando o metrô da cidade de Londres como base de testes, foi possível descobrir que a lotação de pessoas ocorre com mais

frequência durante os dias úteis da semana, com pequenos picos durante o dia. Além disso o trabalho demonstra que a análise dos dados pode ser feita até por uma técnica simples como a de histórico de médias, dessa forma podendo ser incorporado aos sistemas de rotas, fazendo com que esses possam levar a lotação de pessoas como um dos fatores para determinar rota.

No campo do monitoramento metroferroviário também há algumas pesquisas que utilizam bases de dados de trens para realização de mineração de dados. No artigo (Bin and Wensheng, 2015) existe um preocupação com a manutenção diária de trem unidade elétrica (TUE). Essa manutenção gera um grande número de dados que podem ser utilizados no diagnóstico de falhas dos TUEs. Através da mineração de dados e inferição de novas informações, foi possível identificar e criar potenciais regras para guiar a manutenção desses trens. Este trabalho propõe o uso de uma versão melhorada do algoritmo de associação FP-Growth que se encaixa nas necessidades do diagnóstico de falhas do TUE e também aumenta a eficiência e a precisão da mineração.

No trabalho apresentado nesta tese, buscamos integrar as duas vertentes abordadas neste estado da arte. Ao mesmo tempo que trabalha-se com a hipótese de que o uso de acelerômetros e girocópios podem melhorar o monitoramento de trens, técnicas de mineração de dados são utilizadas para mostrar que as possibilidades de obtenção de novas informações são maiores do que as já conhecidas. No capítulo de resultados é possível entender melhor os resultados destas técnicas.

CAPÍTULO 4

MATERIAIS E MÉTODOS

Este capítulo é dedicado a explicar a metodologia que foi adotada para desenvolvimento desta tese, bem como os materiais, ferramentas e tecnologias que foram e que serão utilizados para desenvolvimento dos protótipos e obtenção dos resultados. Inicialmente começamos pela metodologia aplicada no trabalho e como se deu o processo de levantamento bibliográfico que norteou as decisões de projeto. Em seguida descrevemos com mais detalhes o mapeamento sistemático realizado e as informações coletadas. Por fim, detalhamos o processo de desenvolvimento do dispositivo, teste e análise de dados necessários para execução deste trabalho.

A metodologia adotada para o desenvolvimento deste trabalho consiste de 4 etapas principais: levantamento bibliográfico através de um mapeamento sistemático, especificação e definição das hipóteses do projeto e desenvolvimento dos protótipos necessários, realização dos testes reais no METROREC CBTU/STU-REC e por fim, avaliação e análise dos resultados. A Figura 4.1 mostra um fluxograma da metodologia adotada para este projeto e as etapas mencionadas.

Figura 4.1 - Etapas da metodologia adotada para esta tese.

Vale salientar, que parte da última etapa, análise dos resultados, utiliza ferramentas e algorítmos com base na inteligência artificial para realização da mineração de dados, extração das informações mais relevantes e descoberta de novas informações.

4.1 MAPEAMENTO SISTEMÁTICO

De modo a construir-se as bases teóricas para este trabalho, optou-se pela realização de um mapeamento sistemático. Através do mesmo é possível realizar-se um levantamento das fontes bibliográficas mais importantes e relacioná-las às hipóteses do trabalho. O mapeamento sistemático promove uma visualização total da área, permitindo o acesso a muitas evidências que existem sobre o tópico de interesse (Kitchenham, 2012).

De acordo com (Petersen et al., 2008), o uso de mapeamento sistemático tem sido negligenciado por pesquisadores, principalmente das áreas de engenharia. Entretanto, este método proporciona a criação de um mapa com os resultados de pesquisa e revela-se como uma das melhores formas de construir um esquema de classificação e estrutura em um campo de interesse. No trabalho de (Kitchenham and Charters, 2007), o autor afirma que um mapeamento sistemático utiliza uma metodologia de revisão confiável, rígida e passível de revisão.

Baseando-se nas principais ideias de (Petersen et al., 2008; Kitchenham and Charters, 2007; Kitchenham and Brereton, 2013) acerca do mapeamento sistemático é possível compilar ideias em etapas essenciais. Estas etapas são: especificação do tema a ser abordado, criação de strings de busca baseadas nas palavras chaves dos trabalhos disponíveis nas bibliotecas digitais, extração das informações e por fim o mapeamento dos resultados.

4.1.1 Especificação do Tema

O tema abordado no mapeamento é o tema mencionado nos capítulos anteriores e diz respeito a área de utilização de acelerômetros para monitoramento de trens através de uma rede ZigBee bem como no processamento de informações provenientes de tal tecnologia. Este é o tema principal desta tese e a realização de um mapeamento bibliográfico permite o entendimento da área e a análise de tecnologias semelhantes que foram desenvolvidas ou estão em etapa de projeto.

Durante a definição do tema as questões de pesquisa são elaboradas. As questões de pesquisa são perguntas criadas com o objetivo de guiar o mapeamento sistemático, principalmente porque os resultados obtidos por este serão as respostas para essas perguntas. O objetivo desse mapeamento é identificar as pesquisas em execução na área de monitoramento que utilizam acelerômetros em trens e quais as tecnologias utilizadas. Uma pesquisa prévia dessas tecnologias é extremamente útil para futuros trabalhos que têm como objetivo criar novos sistemas de monitoramento com essa mesma finalidade, além de verificar os problemas e lacunas da área. Baseado nesse objetivo, nove questões foram criadas e foram respondidas com os resultados do mapeamento sistemático. Estas questões são mencionadas a seguir.

1. Quais as soluções propostas pelo artigo para o uso de acelerômetros no monitoramento de trens?

2. Quais as tecnologias frequentemente utilizadas?
3. Quais os principais problemas encontrados?
4. Qual as principais variáveis monitoradas com a utilização de acelerômetros em trens neste trabalho?
5. Quais tipos de inforações são inferidas?
6. Há uso de técnicas de mineração de dados?
7. Qual a maturidade do trabalho?
8. Quais os principais problemas enfrentados no desenvolvimento desta solução?
9. Qual a relevância deste trabalho comparado ao trabalho desta tese (Uso da escala Likert: 1 - 5)?

4.1.2 Strings de Busca

O segundo passo do mapeamento sistemático consiste na busca de documentos relacionados ao tema que foi definido anteriormente. Para isso são criadas strings de busca que serão utilizadas para levantamento dos trabalhos da área. Este levantamento é feito de forma automática abrangendo o maior número possível de bibliotecas digitais relevantes para a área.

Uma característica bastante peculiar do mapeamento sistemático que o difere de outras abordagens é que ele considera todos os tipos de estudos feitos a respeito do tema, tais como: análises teóricas, estudos de caso, experimentos, análises práticas, entre outros. Além disso avalia as diversas formas de publicações como artigos científicos, patentes, livros, teses, etc. Assim, garante-se uma completa análise da área na literatura mundial.

A definição das palavras chaves a serem utilizadas norteiam a criação da string. Elas foram divididas em dois grupos: palavras essenciais e palavras flexivas. Palavras do grupo essencial devem sempre estar contidas nos resultados obtidos, as do grupo flexivo não precisam obrigatoriamente estar contidas.

- **Palavras essenciais:** trem, acelerômetro, zigbee.
- **Palavras flexivas:** monitoramento de trens, sistema inteligente de transporte, Redes Inteligentes de trens.

Com base nas palavras chaves foi possível a definição da string de busca. Para a elaboração e utilização correta de uma string é imprescindível a utilização de conectivos lógicos tais como OR e AND. Esses conectivos permitem relacionar os termos da string de forma apropriada. A língua utilizada para as strings é o inglês, visto que é a língua comum da maioria dos

trabalhos acadêmicos mundiais e mesmo quando não é, há a inclusão de um abstract. Para este trabalho a string criada foi a seguinte:

String de Busca: (train OR metro OR subway) AND accelerometer AND ZigBee

Apesar da string ser objetiva e conter os tópicos essenciais para a pesquisa, a carência de trabalhos na área não permitiu um apanhado suficientemente grande para análise do tema. Portanto, a string passou por refinamentos e eliminação de algumas palavras para que pudéssemos obter o maior número possível de trabalhos que fossem relevantes. Por isso, a busca acabou removendo o termo ZigBee das pesquisas, embora no momento da leitura o termo tenha sido levado em consideração para comparações.

4.1.3 Extração das Informações

A extração das informações relevantes para análise ocorre através da revisão dos resultados da busca incluindo os critérios de inclusão e exclusão. Tais critérios filtram os trabalhos, resultando somente os que respondem satisfatoriamente às questões de pesquisa. Uma definição clara desses critérios é importante porque diversas bibliotecas retornam muitos resultados e grande parte não são relevantes para o mapeamento. Além disso, trata-se um conceito central na área, já aprovado entre os estudiosos, por não gerar erros de classificação de resultados. Neste caso, foram definidos três critérios: o artigo utiliza acelerômetros em sua solução, descreve um sistema para ser utilizado em trens, realizou testes da sua solução.

Para seleção das bibliotecas digitais utilizadas na extração das informações foram consideradas o campo de atuação das mesmas e a relevância para as áreas deste projeto no campo da ciência da computação e das diversas engenharias. Todos as bases são de alcance mundial e podem ser facilmente acessadas através de uma interface Web. Portanto, 5 engenhos de busca foram utilizados e são mencionados a seguir:

- Association for Computer Machinery (ACM)[1]: A ACM Digital library é uma plataforma Web que contém os textos com as publicações da ACM como: revistas, anais de conferências, boletins e livros.

- Institute of Electrical and Electronics Engineers (IEEE)[2]: A IEEE Explore Digital Library é um recurso poderoso para a descoberta e acesso ao conteúdo científico e técnico publicado pelo IEEE e seus parceiros. Contém mais de três millhões de documentos de texto disponíveis.

- Springer[3]: Contém artigos, revistas eletrônicas de periódicos, séries de livros digitais e jornais de alta qualidade através do seu serviço online. A biblioteca Springer contém mais de 1.250 jornais.

[1] http://dl.acm.org/
[2] http://ieeexplore.ieee.org/xpl/aboutUs.jsp
[3] http://www.digitallibrary.edu.pk/springer.html

- American Society of Mechanical Engineers (ASME)[4]: Realiza operações de publicações técnicas do mundo inteiro, oferecendo milhares de títulos, incluindo os principais jornais da área de engenharia mecânica e correlatas, anais de conferências e livros. Reúne publicações desde o ano de 1960 até o presente.

- Google Scholar[5]: Oferece de forma simples uma pesquisa ampla na literatura acadêmica, no qual pode-se pesquisar em várias bases de dados artigos, teses, livros, resumos e outros tipos de trabalhos científicos.

Após a execução das etapas supracitadas, foi realizada a busca nas bibliotecas retornando diversos artigos. Posteriormente, é verificado se existem publicações repetidas entre as bibliotecas; se sim, é feita a eliminação dos artigos repetidos mantendo somente um na base de análise do mapeamento.

Após a realização da busca inicial em todas as bases mencionadas no protocolo anterior obteve-se um total de: 3205 trabalhos relacionados ao tema. A filtragem dos resultados deu-se em três etapas principais. Todas as etapas foram realizados por 3 pesquisadores para dar um maior poder de credibilidade ao processo. Inicialmente analisou-se os títulos e resumos dos artigos, um a um, verificando qual a relevância dos mesmos baseando-se nos critérios de inclusão e exclusão. Na segunda etapa todos os artigos foram listados, removendo-se os duplicados. Nesta etapa um outro pesquisador fez-se necessário para ajudar no julgamento dos casos de divergência. Esses casos são os que somente um ou dois dos pesquisadores selecionou o artigo. Após o debate era decidido se o trabalho deveria ser selecionado para leitura ou não. Por fim, os artigos foram lidos em sua totalidade pelos pesquisadores e definidos quais eram relevantes para pesquisa, restando apenas 14 artigos com o foco do tema do mapeamento. A Tabela 4.1 mostra a evolução destas etapas e a quantidade de artigos resultante de cada uma delas.

Tabela 4.1 - Informações da filtragem de trabalhos do mapeamento

Acervo	Encontrados	Descartados	Selecionados
Springer Link	250	250	0
IEEE Explore	41	33	8
ACM	84	78	6
Google Scholar	2490	2490	0
ASME (busca 1)	150	149	1
ASME (busca 2)	190	188	2

4.1.4 Mapeamento dos Resultados

A análise dos resultados é focada em apresentar a quantidade e qualidade dos resultados para as categorias estabelecidas. Assim, é possível ver quais tipos de estudo são mais

[4]http://asmedigitalcollection.asme.org/

[5]https://scholar.google.com.br/intl/pt-BR/scholar/about.html

enfatizados no tema a ser estudado, e quais outros estão em menor quantidade, abrindo uma possibilidade para análises e estudos futuros.

Existe o costume do mapeamento de resultados ser feito por meio de "mapas de bolhas" (bubble charts), para mostrar melhor a frequência de uma união de ângulos do trabalho, seja a relação entre os contextos e os tipos de pesquisa feitos, ou entre contextos e tipos de contribuições, entre outras relações. Entretanto, pode ser feito por quaisquer gráficos, que sejam adequados às visões propostas pelo autor.

O trabalho de (Budgen et al., 2008) mostra as vantagens de se basear uma pesquisa em um mapeamento sistemático previamente realizado, bem como se esta técnica é aplicável para suportar futuras pesquisas relacionadas a algum tema específico. Segundo os autores, conclui-se que a partir deste tipo de estudo, pode-se determinar quais trabalhos são essenciais para responder às questões levantadas no início da pesquisa e também onde estes dados foram publicados e quais foram as saídas obtidas através deles (Kitchenham and Charters, 2007). A análise dos artigos encontrados encontra-se no estado da arte deste trabalho e já foi debatida no capítulo anterior.

4.2 DESENVOLVIMENTO DO PROTÓTIPO

Nesta etapa, as informações colhidas são evidenciadas na criação da hipótese central do trabalho. Partindo das informações geradas através do mapeamento sistemático e as necessidades do projeto o esforço foi direcionado para o desenvolvimento do protótipo que foi utilizado para a etapa de testes.

Ainda seguindo os resultados encontrados no mapeamento, foram realizados estudos mais direcionados às ferramentas e tecnologias adotadas já adotadas. Tais resultados nortearam a definição e aquisição de componentes essenciais para criação do protótipo, desenvolvido para aquisição dos dados provenientes do acelerômetro. As leituras dos manuais dos dispositivos ajudou na obtenção de informações como tensões, métodos de configuração, etc. Com o conhecimento necessário e componentes adquiridos, foram feitas as ligações físicas entre pinos, módulos e placas.

Por fim, foi executada a montagem do protótipo utilizando uma protoboard. Uma protoboard é uma placa de ensaio que possui furos e conexões condutoras utilizadas para a montagem de circuitos experimentais, uma vez que não necessita de soldagem. Após a confirmação de que as ligações entre os módulos estavam corretas e os dados chegavam em um nó coordenador experimental, ligado a um computador, foi realizada a soldagem em uma placa perfurada para que as trepidações do trem não desconectem nenhum componente. Contudo, uma placa de circuito impressa de melhor qualidade ainda pode ser desenvolvida com o intuito de melhorar ainda mais o protótipo. Mais detalhes sobre o protótipo desenvolvido serão mencionadas no próximo capítulo.

4.3 TESTES E COLETA DE DADOS

A etapa de testes foi dividida em 3 ciclos. No primeiro foram realizados testes em laboratório para garantir que o protótipo estava sendo desenvolvido como os requisitos, que seus componentes estavam calibrados e que os dados recebidos eram válidos. Nesta etapa, alem de uma rede mestre escravo desenvolvida e implementada localmente em labratório, foram realizadas simulações para verificar o comportamento da rede de sensores sem fios em casos de funcionamento com muitos nós. Programas de simulação de RSSF foram utilizados para esta finalidade.

Na segundo ciclo da etapa de testes, colocado dentro do trem enquanto o mesmo fazia algumas viagens diárias. Durante os testes armazenou-se localmente dados do acelerômetro e giroscópio, e também, os dados já presentes na cabine do maquinista, como velocidade, pressão nas bolsas de ar, etc.

Por fim, o terceiro ciclo contemplou uma bateria de testes mais completa. O protótipo foi instalado dentro da parte elétrica de um dos vagões do TUE e ficou em funcionamento por vários dias coletando dados do acelerômetro e giroscópio. Durante este ciclo de testes diversos contratempos foram vivenciados. Dentre eles o que mais afetou a aquisição contínua de dados foi a quebra em partes mecânicas dos trens em testes. Isso levou a equipe a realizar a migração dos dispositivos para outro trem a fim de continuar-se os testes.

Após o último ciclo todos os dados foram coletados e levados de volta ao laboratório na UFPB. Devido ao sigilo necessário no METROREC, nesta etapa foram permitidos somente os dados de nosso protótipo. Com isso alguns dados relevantes para possíveis comparações como a velocidade, peso das bolsas de ar e abertura de oprtas ficaram de fora da segunda análise de dados e extração de informações.

Os dados coletados para este trabalho foram obtidos a partir da colaboração entre o Laboratório de Energia Solar (LES - UFPB), do Laboratório de Computação Ubíqua e Móvel (LUMO-CI-UFPB), do IFPE e do METROREC, empresa que gerencia o metrô da cidade do Recife. O metrô do Recife é composto por duas linhas movidas à eletricidade, centro e sul, e duas linhas mais antigas movidas à diesel. Este estudo teve como foco de monitoramento as linhas em que os veículos utilizam eletricidade, os chamados TUEs. Para o recolhimento de dados foi utilizado um protótipo instalado no trem 21 da linha sul, realizando o percuso da estação Recife até a estação Cajueiro Seco, como mostrado na figura 4.2.

É importante destacar que o Metrô da Cidade do Recife possui a característica de ser um metrô de superfície, o que pode ocasionar diversos problemas de manutenção dos trens por estarem mais suscetíveis as intempéries. Essa é outra vantagem de uso de acelerômetros que pode ser utilizada em veículos deste tipo, melhorando a manutenção preditiva dos mesmos através de detecção de irregularidades nas vias ou outras situações que levem o metrô à quebra.

A coleta de dados deu-se em dois momentos distintos. O primeiro momento de coleta (CODA1 - Coleta de Dados 1) foi planejado para ser realizado entre o dia 24 de outubro de 2016

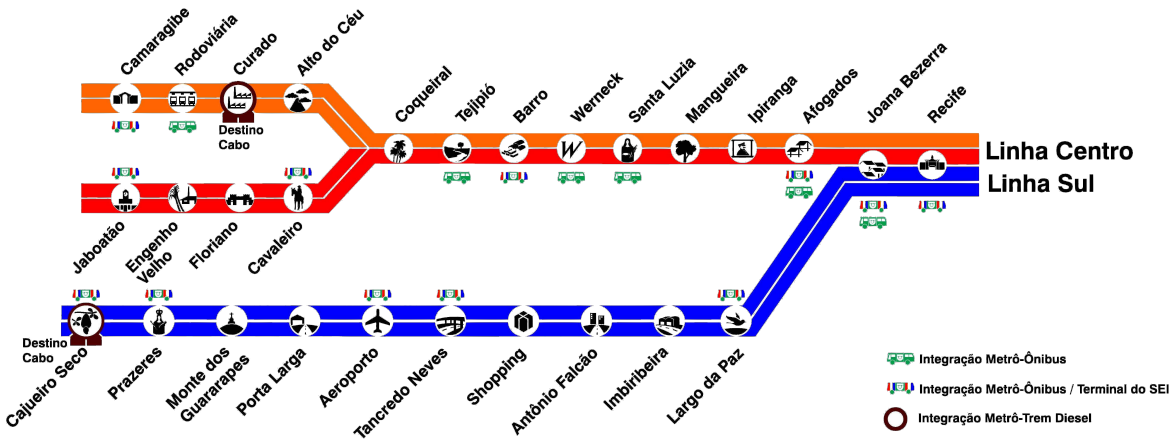


Figura 4.2 - Mapa do metrô de Recife. Fonte: http://mapa-metro.com/.

e 02 de novembro. Infelizmente, devido a problemas técnicos de ordem mecânica, o trem com o protótipo acoplado passou parte do período de coleta sem realizar viagens. O grande esforço por parte da equipe de manutenção e o pessoal do IFPE fez com que o trem ainda estivesse ativo alguns dias de testes. Assim, para o CODA1 tivemos como dados brutos completos os dias 26 e 28 de outubro e parte dos dados dos dias 24, 27, 29 e 31 de outubro, ficando o trem parado por completo nos dias 25 de outubro e 1 de novembro.

Durante o CODA1 dois protótipos estavam acoplados ao trem: o protótipo deste trabalho e um protótipo do MetroRec com o Projeto RailBee. Ao serem analisados, os dados do acelerômetro possuíam irregularidades pois apresentavam variações extremas em alguns momentos. Estima-se que isso possa ter ocorrido por duas razões: um pequeno bug no código, que foi reparado para a segunda bateria de testes, ou por interferência eletromagnética, também corrigida no segundo momento de testes com o uso de uma caixa e materiais de proteção. Os dados do outro protótipo do MetroRec incluíram pressão das bolsas de ar, velocidade, corrente elétrica recebida pelo trem e informações sobre os freios. Após analisados pela direção, somente os dados de velocidade foram repassados para nossa equipe a fim de correlacionarmos estes com os dados obtidos em nosso protótipo. O uso dos dados de velocidade fez com que pudéssemos melhorar os dados de aceleração recebidos e também inferir outras informações como dados relacionados às chegadas, paradas e saídas dos trens nas estações

No segundo momento de coleta de dados denominado de CODA2, todas as precauções necessárias para evitar os problemas anteriores foram tomadas. Esta coleta deu em diversos momentos e compreende as datas seguintes: 02/11/2016 à 05/11/2016, 05/11/2016 à 03/12/2016, 05/12/2016 à 26/12/2016. Como durante esse período o veículo 21 que estava sendo utilizado continuou apresentando problemas mecânicos, no ano de 2017 a MetroRec permitiu a mudança de nosso protótipo para outro veículo. Sendo assim, ainda durante a CODA2, o trem 23 da linha sul capturou dados entre o período de 24/02/2017 a 04/03/2017. Nesta segunda etapa de coleta de dados, a gerência do MetroRec não permitiu a passagem das outras informações do

trem para comparação com nosso projeto. Os dados ficaram sob análise da própria empresa. Portanto, nesse momento a mineração foi realizada somente com os dados de nosso protótipo.

4.4 ANÁLISE DE DADOS

Nesta etapa, os dados coletados nos processos anteriores são analisados utilizando técnicas de inteligência artificial, mais especificamente, mineração de dados, para buscar-se padrões e inferir-se novas informações a partir dos dados. Depois, as informações são comparadas, analisadas e estudadas, para que possamos validar a hipótese inicial de que além de inovador, o sistema proposto permite o uso de mineração de dados como forma de buscar-se novas informações que auxiliem no monitoramento, controle e manutenção dos sistemas metro-ferroviários.

Diversas ferramentas utilizadas para a mineração de dados estão disponíveis tanto para uso comercial como acadêmico. Estas ferramentas possuem uma grande gama de funcionalidades, como: vários algoritmos de mineração, técnicas de visualização das informações extraídas, algoritmos de pre-processamento e pós-processamento, entre outras. Entre as mais comuns podemos citar: IBM Intelligent Miner, DBMiner, Clementine, Oracle Data Miner, RAPIDMINER e o WEKA (Cornelius Junior, 2015).

Neste trabalho foi utilizada a ferramenta WEKA, versão 3.8.0, para o processo de mineração de dados. O WEKA é uma ferramenta de código livre, desenvolvida em Java, que utiliza algoritmos de aprendizagem de máquina para o processo e as tarefas de mineração (Araújo, 2009). Foi construída para atender diversos requisitos incluindo uma alta qualidade requerida por sistemas alvo, oferecendo assim um meio para a experimentação, comparação e testes de diversos modelos de aprendizagem. Pode ser utilizada para a preparação dos dados através de filtros de maneira sumarizada ou em detalhes. Apresenta os resultados graficamente através de uma interface simples e bom nível de usabilidade (Pachiarotti, 2012). O WEKA foi escolhido por ser a ferramenta mais utilizada por pesquisas que procuram identificar padrões em um conjunto de dados do tipo que se adequa ao objetivo deste trabalho, somado a isso o sistema é de fácil entendimento e uso, além de prover uma vasta documentação.

Ao inicializar, o WEKA apresenta uma janela, apresentada na figura 4.3, composta por cinco opções de ambientes para serem selecionados: *Explorer*, *Experimenter*, *KnowlodgeFlowI*, *Workbench* e *SimpleCLI*. Essas opções significam, respectivamente, explorador, experimentador, fluxo de conhecimento, mesa de trabalho e linha de comando simples (Pachiarotti, 2012).

A opção Explorer contém o ambiente onde são disponibilizados as técnicas para a realização da mineração de dados. Sendo o principal ambiente do WEKA o *Explorer*, demonstrado na figura 4.4 possui abas que permite diversas funções, como o pré-processamento, realizado através de diversos filtros tais como: discretização, normalização e reamostragem, seleção de atributos, transformação, combinação de atributos, entre outros (Pachiarotti, 2012), e a extração de informação a partir das técnicas de classificação, clusterização e associação. A opção *Experimenter* é um ambiente onde é possível comparar o desempenho de diferentes modelos

Figura 4.3 - Janela Inicial do WEKA. Fonte: Imagem gerada pelo autor

de aprendizagem, além de fornecer a possibilidade de realizar testes estatísticos entre os modelos (Araújo, 2009). O ambiente *KnowledgeFlow* oferece funções semelhantes ao ambiente *Explorer* utilizando um diagrama de fluxo de dados para mostrar as etapas, sendo esse diagrama simples de ser criado a partir de uma interface drag and drop. O ambiente de *WorkBench* somente agrega todas as funcionalidades dos outros ambientes em uma só janela, facilitando o processo de execução de outras tarefas. E por último o ambiente *SimpleCLI* que é uma interface de linha de comando que fornece um ambiente para a execução de comandos WEKA (Araújo, 2009).

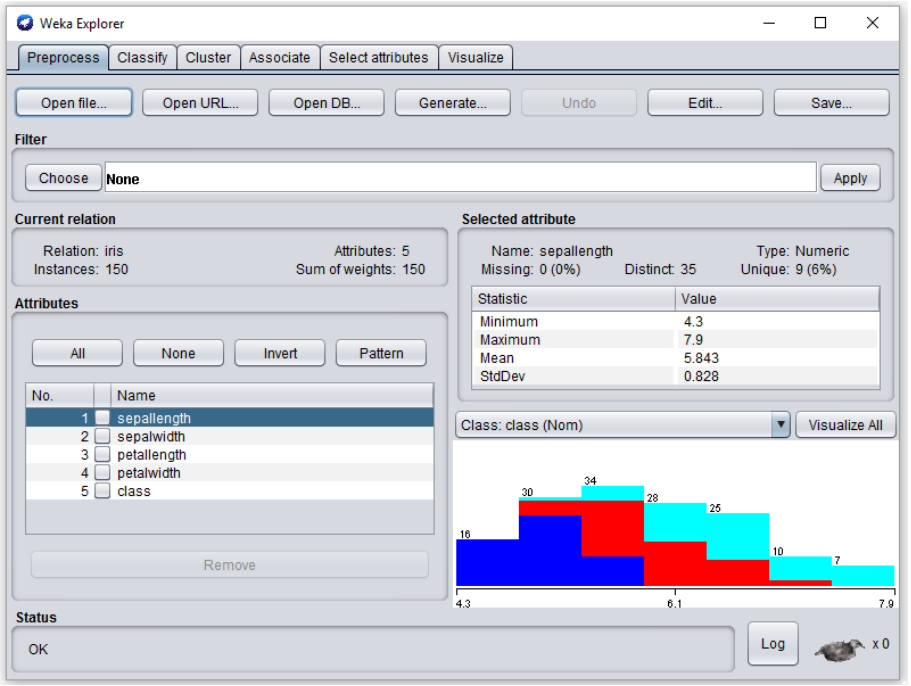


Figura 4.4 Ambiente *Explorer* do WEKA, aba de pré-processamento. Fonte: Imagem gerada pelo autor

Como dito anteriormente o ambiente *Explorer* é o ambiente principal do WEKA e também o único ambiente utilizado neste trabalho. No topo é possível observar diversas abas que

representam diversos processos como, pré-processamento, classificação, clusterização, entre outros. No pré-processamento é escolhido a base de dados que será utilizada no processo. É possível depois dessa escolha se realizar diversas operações com a base, de forma a prepará-la para a passagem pelos algoritmos de extração. Aplicar algum filtro descrito anteriormente e escolher os atributos que serão utilizados no processo são algumas das operações possíveis.

Por último chegamos nas abas relacionadas a execução dos algoritmos, *Classify*, *Cluster* e *Associate*, na figura 4.5 é apresentada a aba *Classify* como exemplo. Estas abas têm interfaces parecidas e possuem elementos como a escolha do algoritmo que deve ser utilizado, as opções para os testes e uma visualização em forma de texto com os resultados do processo. As abas de *Select Attributes* e *Visualize* são abas responsáveis respectivamente por um processo de seleção dos atributos dos dados utilizando alguns algoritmos e pela visualização gráfica da base de dados. Por isso não serão utilizadas neste trabalho.

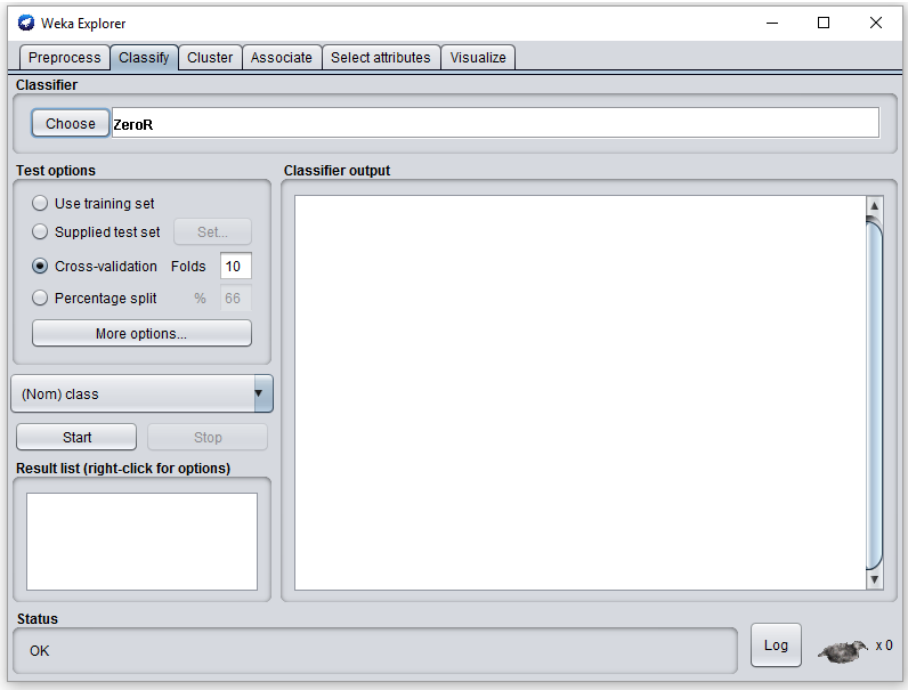


Figura 4.5 - Aba *Classify* do ambiente *Explorer* do Weka. Fonte: Imagem gerada pelo autor

As fases realizadas nesta etapa são: pre-processamento, extração de padrões, pós-processamento e utilização do conhecimento. Estas fases são melhor explicadas a seguir.

Pré-Processamento

Pelas características e problemas apresentados no tópico anterior o pré-processamento ocorreu em duas etapas. Durante a primeira etapa foi necessária a extração dos dados que poderiam ser obtidos a partir da análise da velocidade, a principal motivação neste ponto era a obtenção da parada do trem em uma estação e as possíveis informações que podem ser encontradas a partir deste dado. Primeiramente foi necessário a criação de um código em Java para a análise do arquivo que continha esses dados. O programa criado procurava por indicações que

mostrassem pela velocidade a parada do trem, ao encontrar uma parada o código era capaz de obter o horário em que o trem parou e o horário que começou a se mover, a partir disto foi possível identificar o tempo de permanência durante a parada. Com isso, foi possível determinar o tempo de frenagem, sendo este o tempo que leva para um veículo em determinada velocidade decresce-lá até chegar à zero, e também os valores dos picos de velocidade e o tempo até esta ocorrência.

Com os dados obtidos foi possível gerar diversas linhas que representavam paradas do trem durante a execução do seu percurso. Contudo, os resultados apresetaram anomalias provenientes da variação de velocidade, indicando a necessidade de limpeza dos dados. Por exemplo, não necessariamente uma parada do trem indica que isso está ocorrendo para o embarque ou desembarque de passageiros, podendo ser uma parada de espera entre estações, conserto, ou paradas por eventos externos (como quando há uma irregularidade detectada no trilho). Dessa forma, através de debates com especialistas no domínio, qualquer parada com tempo menor que 10 segundos e maior que 400 foi eliminada da base de dados, formando assim 328 paradas identificadas durante o período de recolhimento de dados em CODA1. Cada linha da base de dados é formada por diversas colunas de atributos e os valores associados a eles, essas linhas são chamadas de instâncias pelo WEKA. Portanto, o nome instância substituirá o termo linha da base de daos no dercorrer do trabalho.

Na segunda etapa do pré-processamento os dados gerados são adicionados a um banco de dados em MySQL com o propósito de guardá-los para uma futura necessidade de reavaliação. A partir do banco de dados é necessário exportá-lo para um arquivo de extensão CSV (*Comma Separated Values*) já que a partir do mesmo é possível gerar um arquivo no formato ARFF (*Attribute-Relation File Format*) permitindo a leitura dos dados pelo Weka. Com a base de dados transformada é possível então prosseguir para o processo de extração de padrões.

Extração de Padrões

Durante o processo de extração de padrões diversos algoritmos das classes de classificação, clusterização e associação foram utilizados sobre a base de dados. Alguns dos algoritmos utilizados necessitavam de formas específicas de uso das bases de dados, como por exemplo os algoritmos de classificação, que necessitavam que a base de dados fosse dividida em duas partes, para formar um conjunto usando na criação de um modelo de classificação e um conjunto teste para validá-la. Além disso a classificação também requer uma coluna base para a criação de um etiqueta classificatória, que neste caso foi utilizado o tempo de parada na estação, como mostrabo na Tabela 4.2.

Através da interface do WEKA a base de dados é carregada e é possível a manipulação dos dados presentes nela, inclusive através da criação de filtros para melhorar a qualidade dos dados. Neste trabalho não foram necessários a utilização desses filtros, contudo dependendo de novos dados para análise, tais filtros podem ser requeridos.

Tabela 4.2 - Classificação das paradas através do tempo

Etiqueta	Tempo parado na estação
1	$\leq$ 15s
2	$>$ 15s e $\leq$ 30s
3	$>$ 30s e $\leq$ 60s
4	$>$ 60s

A primeira extração feita nos dados vem a partir dos gráficos mostrados pelo WEKA sobre os elementos que compõem a base de dados, mostrados em gráficos de colunas e apresentando se necessário a interação entre dados. A partir deste ponto os algoritmos são aplicados e suas saídas são guardadas para análises futuras na etapa de pós-processamento, devendo-se sempre analisar quais algoritmos serão aplicados sobre os dados, pois dependendo do resultado esperado e dos dados da base alguns algoritmos não retornaram dados úteis.

Durante o processo de extração alguns resultados dos algoritmos foram considerados incorretos em uma primeira análise, dessa forma foi necessário consecutivas execuções dos algoritmos, mudando-se diversos parâmetros e entradas, para que as afirmações sobre os resultados pudessem ser apresentadas com alto grau de certeza.

Pós-Processamento

Após a aplicação dos algoritmos é necessário que os resultados sejam analisados de forma a identificar se os modelos criados, as informações e os padrões extraídos são realmente úteis e estão corretas. Muitos dos algoritmos utilizados podem não gerar informações relevantes, isso se deve ao fato que existem algoritmos que se adaptam melhor a certo tipos de dados de entrada.

A partir das três classes de algoritmos definidas anteriormente no trabalho, sendo elas classificação, clusterização e associação, os resultados de saída de cada uma foram analisados de forma diferente. Para os resultados dos algoritmos de classificação é necessário se avaliar a eficiência do modelo criado e sua eficácia em classificar novas instâncias de dados. Nos algoritmos de clusterização é necessário se observar o valor dos atributos do centróide de cada cluster formado e a partir dele inferir padrões entres as instâncias agrupadas em cada cluster. Por último nos algoritmos de associação é essencial se analisar as regras criadas a partir da base de entrada e verificar a veracidade e lógica das regras criadas e como serão utilizadas.

Como citado anteriormente no capítulo 2 o processo de mineração de dados é um processo iterativo, pode ser que os resultados gerados pela execução dos algoritmos sejam questionados e seja necessário mudar atributos nos algoritmos para que em uma nova execução sejam retirados dados mais concretos. Dessa forma as etapas de extração de padrões e de pós-processamento podem acontecer diversas vezes antes de gerar informações relevantes e ter resultados com maior grau de certeza.

Utilização do Conhecimento

No processo de utilização do conhecimento é discutido como as informações, padrões e modelos criados serão utilizados pelos gestores dos serviços de onde foram extraídos os dados para construção da base. A partir dos resultados encontrados foram identificados os algoritmos mais propensos a extraírem bons resultados da base de dados utilizada. Também pode ser identificado diversos parâmetros sobre a base construída, principalmente sobre seu tamanho e os atributos presentes nela. E por último os resultados encontrados mostraram tendências de comportamento dos dados apresentados podendo identificar futuros problemas que podem ser resolvidos para melhoras a prestação dos serviços.

CAPÍTULO 5

PROTÓTIPO

Neste capítulo serão descritos com mais detalhes o protótipo e infraestrutura desenvolvidos para o projeto. Durante o processo de desenvolvimento diversas medidas precisaram ser tomadas para contornar os problemas enfrentados, seja por ordem econômica, com vistas a reduzir o custo do produto final, seja por incompatibilidade de tecnologias. Como resultado do protótipo, tivemos um modulo completo que foi integrado ao sistema RailBee e que inclui os sensores mencionados anteriormente: acelerômetro e giroscópio, bem como, todos os outros circuitos necessários para que o módulo atendesse as suas necessidades. Vale ressaltar que o protótipo de testes não se restringe somente ao módulo remoto de aquisição, mas também, a um sistema de monitoramento e uma base de dados remota para armazenamento das informações, além de uma base local utilizada como backup.

5.1 FUNCIONAMENTO DO SISTEMA

O sistema é composto por 3 partes principais: a Parte I, contempla a rede de sensores sem fios localizada nos trens com os nós remotos, nos postes, que servem como roteadores, e nas estações de metrô, neste caso os nós base. A parte II que é composta pela parte do servidor, onde está localizada a base de dados principal, e a base do sistema que será acessado através de *Web-Sevices*. E por fim a Parte III que são os dispositivos que podem acessar os serviços disponíveis no servidor, tais como celulares, computadores, relógios inteligentes, totens com informações localizados nas estações, etc. A figura 5.1 mostra o funcionamento completo do sistema com as três partes integradas.

De acordo com o diagrama mostrado na figura anterior percebe-se a existência dos dispositivos localizados nos trens. Estes são os nós finais da RSSF e medem constantemente as diversas grandezas de sensoriamento ou capturam sinais necessários para monitoramento. Estes dados são enviados para o nó base (sink) na estação de metrô mais próxima. Este envio pode ocorrer diretamente do nó final para o base ou através de nós roteadores dispersos ao longo das

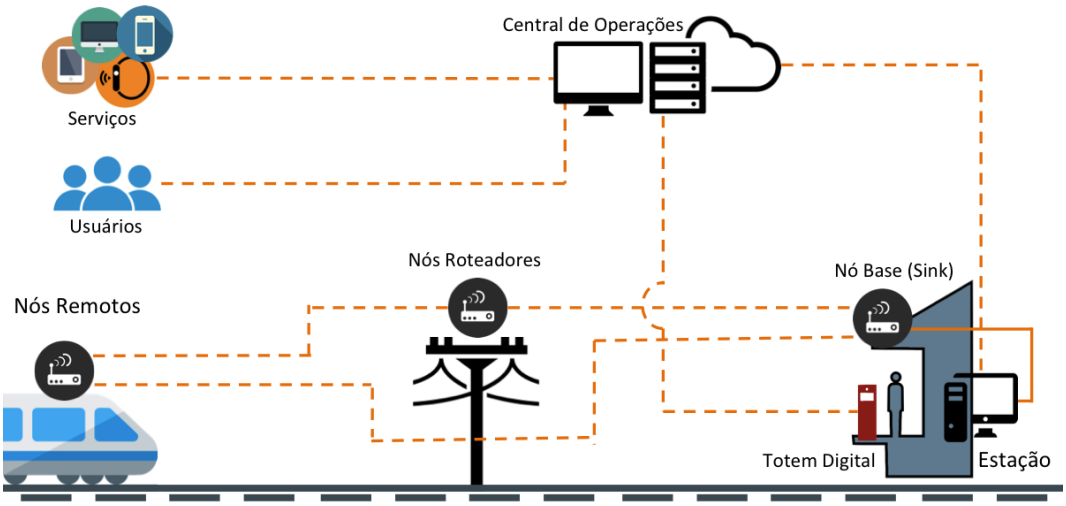


Figura 5.1 - Modelo de funcionamento da versão final do projeto RailBee.

vias permanentes. Esses passos fazem parte das atividades realizadas pela parte I do projeto. Nesta etapa, a transmissão de dados entre os dispositivos é completamente sem fios.

Ao receber os valores das medições realizadas, o nó base os envia para o servidor através de uma rede cabeada que utiliza fibra óptica. As informações são então armazenadas na base de dados e disponibilizadas na nuvem de aplicações que estará disponível. A partir de então, o monitoramento pode ocorrer tanto para as características monitoradas automaticamente, quanto para o monitoramento pelo usuário via sistema.

Além do monitoramento, diversas informações estarão disponíveis para vários tipos de usuários: maquinistas, controladores, população em geral, pessoal de manutenção, etc. Isso ocorrerá através dos serviços implementados no servidor central após a análise e disponibilização das informações. Portanto, todas essas partes são integradas através de redes, a primeira utilizando uma rede de sensores sem fios, a segunda uma rede cabeada de fibra-óptica e por fim as informações na nuvem que podem ser acessadas via Internet.

Como mencionado anteriormente, o sistema desenvolvido neste projeto vem melhorar e incrementar as funcionalidades do atual projeto RailBee. Sendo assim, o dispositivo contendo o acelerômetro e giroscópio ficará na parte interna dos trens acoplado como módulo extra ao sistema de aquisição de sinais do RailBee, unindo-se em um só sistema de aquisição com poder de processamento local de dados.

5.2 COMPONENTES DO PROTÓTIPO

O módulo remoto desenvolvido consiste na junção do circuito acelerômetro/giroscópio a um módulo XBee e um microcontrolador. O uso do microcontrolador deve-se principalmente ao fato de que o acelerômetro adotado (MPU 6050) trabalha com o protocolo I2C (Inter-Integrated Circuit). O I2C é bastante utilizado desde os anos 80 facilitando a comunicação entre componentes e aumentando a flexibilidade e a simplicidade de desenvolvimento de hardware. Contudo

o empacotamento pelo módulo XBee não adapta diretamente o protocolo I2C, havendo a necessidade de uma transformação local de dados. Para isso utilizou-se inicialmente um Arduino Uno que transforma os dados de X, Y e Z do acelerômetro em informações que podem ser lidas e transmitidas pelo módulo ZigBee. A Figura 5.2 a seguir mostra um diagrama esquemático do dispositivo desenvolvido. A seguir descreveremos as tecnologias utilizadas e como foram adaptadas (ou configuradas) para utilização no projeto.

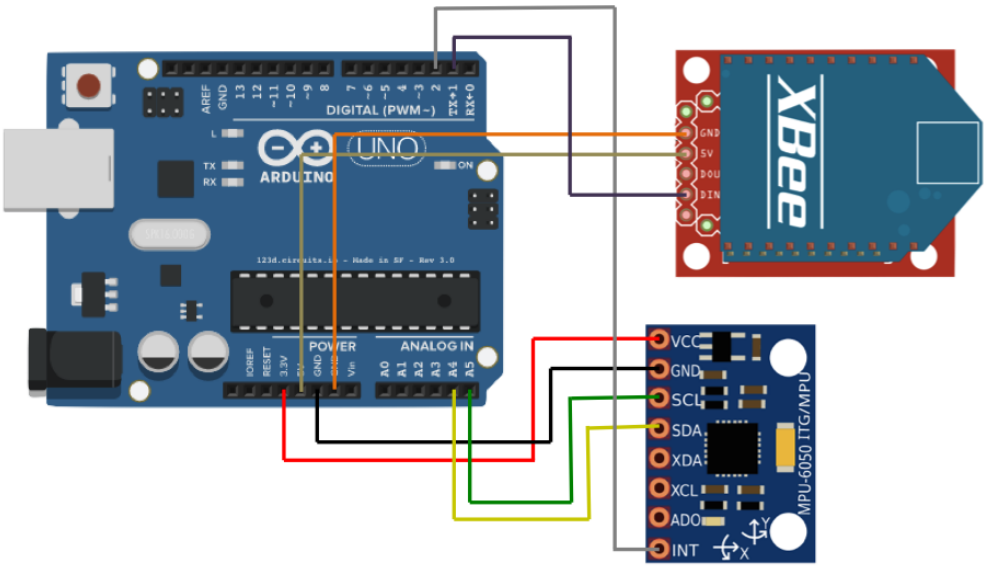


Figura 5.2 - Projeto do módulo desenvolvido para utilização do acelerômetro.

5.2.1 Arduino

Arduino é uma plataforma de prototipagem voltada tanto para o público experiente quanto para o público inexperiente por conta principalmente, da facilidade no manuseio e programação. Nos últimos anos, surgem, a cada dia, placas com essa mesma proposta, algumas mais poderosas, inclusive. Apesar disso, o Arduino continua sendo uma das principais plataformas, principalmente por conta do seu custo reduzido, além de possuir inúmeros dispositivos compatíveis, dada a sua natureza *open-source*, e ter uma comunidade de desenvolvedores extensa, contribuindo para a documentação. Outro ponto positivo é a possibilidade de desenvolver em várias plataformas diferentes, como, por exemplo, Windows, Macintosh OSX e Linux. O projeto feito em um sistema é facilmente portado para outro.

O Arduino é uma plataforma para prototipação eletrônica que possui uma única placa de hardware livre, projetada com um cristal (ou oscilador), relógio simples que envia pulsos de tempo em frequência, um regulador linear de 5 volts e um microcontrolador Atmel AVR previamente configurado com suporte à entrada e saída. Essa junção de tecnologias permite a criação de ferramentas acessíveis, flexíveis, de fácil manipulação e baixo custo. Tanto o

projeto de hardware quanto de software são de código aberto facilitando o uso e aumentando a aplicabilidade do mesmo.

Atualmente, existem diversos modelos disponíveis no mercado, como: Uno, Duemilanove, Mini, Nano, Mega. Suas variações consistem basicamente no tamanho de memória, quantidade de pinos de entrada/saída e tipo de bootloader. Para este projeto, utilizou-se inicialmente o Arduino Uno, por ser o mais popular, comum no mercado e barato. O diagrama da placa do Arduino e seus principais componentes, utilizado no primeiro protótipo pode ser visto na Figura 5.3 e as especificações técnicas são detalhadas na Tabela 5.1.

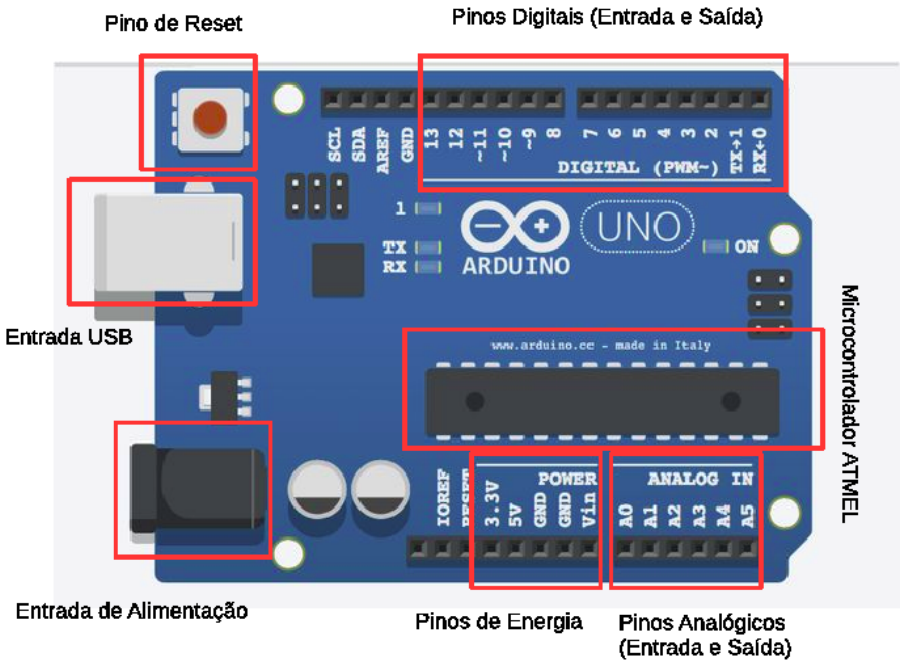


Figura 5.3 - Arduino Uno - Principais Componentes

Ao implementar-se o primeiro protótipo notou-se o grande poder de processamento e memória do mesmo. Devido a isso, testamos uma versão menos poderosa do Arduino, a versão Nano. Esta versão contém os mesmo componentes mencionados anteriores mas com uma capacidade mais reduzida, menor custo e tamanho reduzido. A Figura 5.4 mostra o chip utilizado na segunda versão do protótipo.

O Arduino Nano é uma placa pequena, completa e fácil de se trabalhar, que utiliza microcontroladores do tipo ATmega328 ou ATmega168. Sua tensão de operação é de 5V e conta com uma entrada Mini-B USB. A placa possui 14 pinos de entrada e saída digitais dos quais 6 são saídas PWM e 8 pinos de entrada analógica. Suas dimensões são bastante reduzidas: 45x18mm pesando apenas 5g. As especificações do Arduino Nano podem ser vistas na Tabela 5.2. Observe que apesar da redução de tamanho as funcionalidades se assemelham bastante com o Arduino UNO.

Tabela 5.1 - Especificações técnicas do Arduino UNO

Atributo Técnico	Especificação
Microcontrolador	ATmega328P
Tensão de Operação	5 V
Tensão de Entrada (rec)	7 V - 12 V
Tensão de Entrada (max)	6 V - 20 V
Pinos Digitais de E/S	14 (6 PWM)
Pinos PWM de E/S	6
Pinos Analógicos de E/S	6
Corrente DC por Pino de E/S	20 mA
Corrente DC para 3,3 V	50 mA
Memória Flash	32 kB (0.5 kB para bootloader)
SRAM	2 kB
EEPROM	1 kB
Clock	16 MHz
Comprimento	68.6 mm
Largura	53.4 mm
Peso	25 g

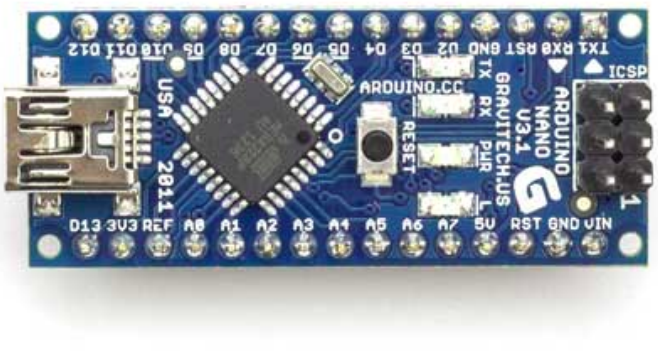


Figura 5.4 - Arduino Nano utilizado no protótipo 2 do projeto. Fonte: Dados produzidos pelo autor.

Neste projeto, o Arduino será responsável por servir de interface entre os sensores de aquisição de dados e um outro dispositivo, baseado no protocolo de comunicação ZigBee, que enviará os dados para a rede. Embora dois protótipos tenham sido desenvolvidos para atender os requisitos do projeto, foi interessante perceber a capacidade de cada um, permitindo planos para trabalhos futuros. Por exemplo, caso deseje-se realizar um processamento local dos dados de forma mais robusta, diminuindo o peso do tráfego de rede, o Arduino Uno do protótipo 1 deve ser a escolha ideal.

5.2.2 Nós Sensores e Módulo XBee

Os nós sensores utilizados neste projeto são do modelo XBee, um módulo devidamente regulamentado pela ANATEL. Embora os nomes sejam semelhantes, é importante deixar clara a

Tabela 5.2 - Especificações técnicas do Arduino Nano

Atributo Técnico	**Especificação**
Microcontrolador	ATmega168 or ATmega328
Tensão de Operação	5 V
Tensão de Entrada (rec)	7 V - 12 V
Tensão de Entrada (max)	6 V - 20 V
Pinos Digitais de E/S	14 (6 PWM)
Pinos PWM de E/S	6
Pinos Analógicos de E/S	8
Corrente DC por Pino de E/S	40 mA
Memória Flash	16 kB-ATmega168 / 32 kB-ATmega328 (2 kB bootloader)
SRAM	1 kB-ATmega168 ou 2 KB-ATmega328
EEPROM	512 B-ATmega168 ou 1 kB-ATmega328
Clock	16 MHz
Comprimento	45 mm
Largura	18 mm
Peso	5 g

diferença entre XBee e ZigBee. O termo ZigBee, como mencionado no capítulo 2, refere-se ao protocolo de comunicação que segue os padrões IEEE 802.15.4 e pertence à ZigBee Alliance. Já o XBee é o módulo físico, fabricado pela Digi International e mostrado na Figura 5.5. O módulo possui diversas variações, como o XBee-PRO Series 2, utilizado neste trabalho, e são fabricados para serem utilizados em uma rede de sensores sem fio que utilizem o protocolo ZigBee. Os princípios que norteiam o XBee são baixo custo e baixo consumo energético, já que os módulos exigem um mínimo de energia ao mesmo tempo que proporcionam uma comunicação de dados confiável entre os dispositivos da rede.

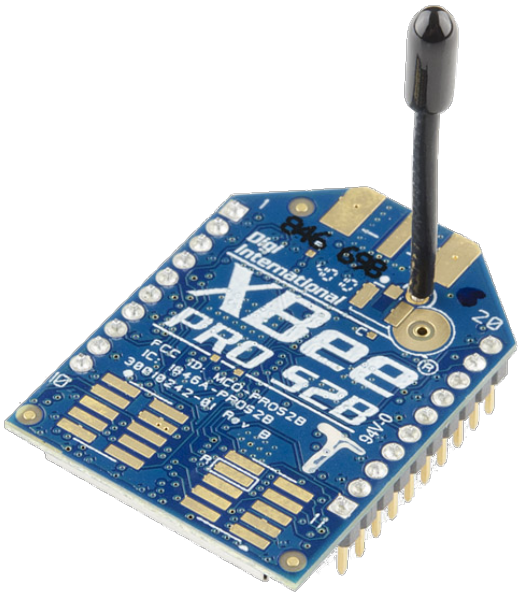


Figura 5.5 - XBee Pro Series 2. Fonte: www.sparkfun.com.

Segundo o manual do usuário fornecido pelo fabricante, o XBee PRO 2 [1], utilizado neste projeto, possui um alcance de 60m a 90m em ambientes fechados e de 1500m a 3200m em lugares abertos. Possui uma taxa de transferência de 250.000 bits/s e necessita de uma tensão de alimentação variando de 3V a 3.4V. No Brasil, o módulo opera em uma frequência de banda de 2,4GHz, aprovado pela Anatel (Norma: 2256-14-1209) e fazendo parte dos sistema de rádio ISM (Industrial, Science and Medical). A Tabela 5.3 mostra as especificações do módulo utilizado e o quanto o mesmo é potente e robusto para aplicações industriais deste tipo.

Tabela 5.3 - Especificações técnicas do XBee Pro S2 utilizado

Atributo Técnico	**Especificação**
Alcance *Indoor*	90m (60m - Padrão Internacional)
Alcance *Outdoor*	3200m (1500m - Padrão Internacional)
Potência de Transmissão	50 mW (+17 dBm) ou 10 mW (+10 dBm) Internacional
Taxa de Transmissão RF	250.000 b/s
Vazão Dados	Até 35000 b/s
Sensitividade do *Receiver*	-102 dBm
Voltagem	3.0 - 3.4 V
Corrente de Operação (enviando)	295mA (170 mA padrão internacional)
Corrente de Operação (recebendo)	45mA
Corrente de Operação (idle)	15mA
Corrente de Operação (sleep)	$3.5\,\mu A$
Temperatura de operação	-40 to 85º C (industrial)
Dimensões	2.438 cm x 3.294 cm
Frequência de Operação	ISM 2.4 GHz
Topologias	ponto-ponto, ponto-multiponto, peer-to-peer e mesh
Número de Canais	14 (11 ao 24)

Para o uso deste módulo, faz-se necessária a realização de algumas configurações iniciais utilizando um software, o XCTU, fornecido pelo próprio fabricante. Nesse mesmo software podemos determinar se o nó sensor entrará em modo *sleep* ou não, qual será a sua função na rede (coordenador, roteador ou nó sensor final), em qual canal ele operará, qual será a sua ID, em qual modo funcionará (AT ou API), entre outros diversos parâmetros. No próprio firmware que o módulo carrega já existem configurações iniciais que fazem com que ele opere como nós finais.

Como a proposta inicial de testes é flexível para questões de comunicação, o parâmetro ID foi configurado de modo que o módulo possa se comunicar de forma dinâmica com o endereço da rede que o XBee encontrar. O endereço de destino dos dados deve ser configurado com o valor 0, que corresponde ao endereço do nó coordenador da rede. O parâmetro IR define o intervalo de tempo, em milissegundos, para a aquisição dos dados que, neste projeto, foi definido com o valor 7D, em hexadecimal (125ms em decimal). Ou seja, a cada 125ms uma amostra será adquirida e enviada, resultando em uma taxa de amostragem de 8Hz, assim como proposto

[1] http://ftp1.digi.com/support/documentation/90000976.pdf

em (Santos, 2009) e atuando na mesma frequência de aquisição que o projeto RailBee. Após essa configuração, o módulo poderá ser conectado ao Arduino e posto em operação.

Após todas as configurações iniciais, para facilitar o processo de prototipagem e a fim de realizar a conexão entre sensores externos, nós sensores e Arduino, foi utilizado um dispositivo chamado de XBee Explorer, que pode ser visualizado na Figura 5.6. Ele serve basicamente para comunicar o Xbee com os componentes conectados à uma placa de prototipagem com o Arduino. Ao receber os dados, o XBee os empacota e disponibiliza na rede que está conectado.

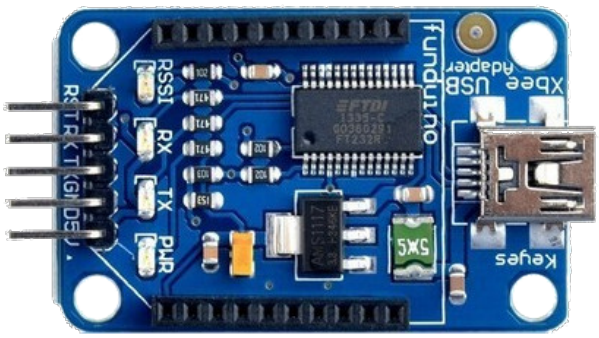


Figura 5.6 - Módulo XBee Explorer USB. Fonte: Dados produzidos pelo autor.

5.2.3 Circuito MPU6050

Os dispositivos da família MPU-6050 são um dos primeiros do tipo MotionTracking do mundo projetados para o baixo consumo de energia, baixo custo e requisitos de miniaturização como smartphones, tablets e sensores portáteis de alto desempenho. O MPU-6050 incorpora o InverSense MotionFusion e calibração de firmware em tempo de execução que permite aos fabricantes eliminar a seleção, qualificação e integração ao nível do sistema de dispositivos discretos em produtos habilitados ao movimento. Isso garante que os algoritmos de fusão de sensores e procedimentos de calibração tenham um desempenho ótimo para quem os utiliza. Um diagrama e uma imagem do MPU6050 que foi utilizado para este projeto podem ser vistos na Figura 5.7.

Os dispositivos MPU-6050 combinam um giroscópio de 3 eixos e um acelerômetro de 3 eixos no mesmo *chip*, juntamente com um processador digital de movimento (DMP) que processa algoritmos MotionFusion complexos de 6 eixos. Todo o circuito contém pequenas dimensões de 4x4x0.9mm. O módulo opera a 400kHz em uma tensão de 3-5V e possui um sensor de temperatura interno, que realiza leituras entre -40 e +85°C. Um exemplo da arquitetura do módulo MPU é mostrado na Figura 5.8. O MPU-6050 provê alto desempenho, baixo ruído e um menor custo de uso se comparado a outros circuitos com acelerômetros. Para projetos futuros, o uso do acelerômetro MMA8452Q também foi considerado.

O MPU-6050 trabalha com o protocolo I2C. Através do barramento I2C o dispositivo pode acessar magnetômetros externos ou outros sensores, permitindo outros dispositivos para reunir um conjunto completo de dados de sensores sem a intervenção do processador do sistema.

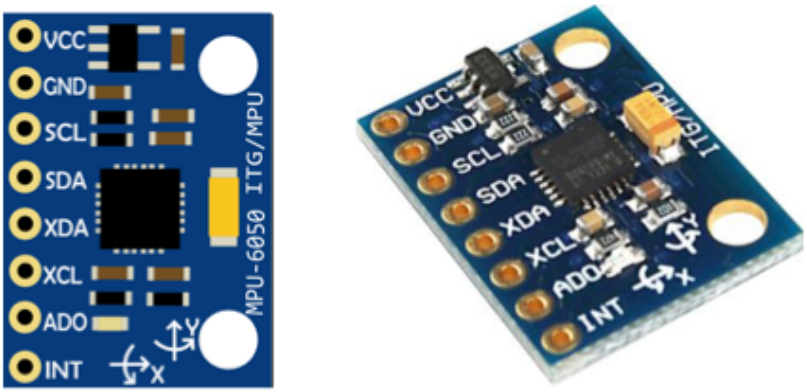


Figura 5.7 - Acelerômetro MPU6054. Fonte: Dados produzidos pelo autor.

Figura 5.8 - Arquitetura do dispositivo MPU6050. Fonte: Dados produzidos pelo autor.

Originalmente desenvolvido em 1982, o I2C permitia apenas comunicação de 100kHz e era fornecido apenas para endereços de 7 bits, limitando o número de dispositivos no barramento. Dez anos depois tornou-se pública a versão que operava a 400kHz e com endereços de 10 bits, amplamente utilizada até hoje. Cada barramento I2C é composto por dois sinais: SCL e SDA, onde o SCL é o sinal do relógio e o SDA o de dados. O sinal de relógio é sempre gerado pelo dispositivo mestre do barramento. Ele permite uma flexibilidade na conexão dos dispositivos com diferentes tensões de entrada e saída, o que significa que podem ser conectados dois dispositivos através do protocolo I2C sem que haja um deslocamento do nível de tensão. É possível observar esse conceito sendo utilizado no presente trabalho, onde o Arduino opera em 5V enquanto o acelerômetro atua em 3.3V, em média.

Podemos observar na figura 5.7 anterior todos os pinos presentes no MPU6050 onde:

- VCC: valores possíveis para a entrada da alimentação de 3 a 5v;
- GND: o negativo da tensão;
- SCL / SDA: Interface I2C, utilizada para transmissão dos dados;
- XDA / XCL: I2C Auxiliar (não foi utilizado para este trabalho);

- ADO: Endereço (não utilizado para o protótipo visto que teremos apenas um acelerômetro);
- INT: pino para uso de interrupções.

5.3 DISPOSITIVO DESENVOLVIDO

Após a análise, estudos e aquisição dos dispositivos, agrupamos os mesmos no circuito final que foi destinado aos testes. O diagrama com a ideia inicial desse módulo foi mencionado no início do capítulo na Figura 5.2. Para isso, foi necessário acoplarmos em um só dispositivo o acelerômetro MPU-6050, um microcontrolador, que neste caso foi o Arduino Nano e o módulo XBee. Os dados adquiridos pelo acelerômetro e giroscópio são processados pelo microcontrolador e em seguida empacotados e enviados pelo Módulo ZigBee.

O primeiro passo para o desenvolvimento do dispositivo foi a calibração do processador DMP do acelerômetro com base em medidas da aceleração gravitacional. Com isso, foi possível diminuir os possíveis erros, obter o máximo do desempenho do módulo e os valores de offset a serem utilizados no programa da captura dos dados. Após calibrado o acelerômetro, foi utilizado o sketch padrão do MPU6050 no Arduino, alterando apenas o valor do offset gerado na calibração e a maneira como são escritos os dados na serial. As ligações entre o acelerômetro e o Arduino Nano foram feitas da forma descrita na Tabela 5.4

Tabela 5.4 Ligações realizadas entre o Arduino e o Acelerômetro. Fonte: Dados produzidos pelo autor.

Acelerômetro	Arduino
VCC	5V
GND	GND
SCL	A5
SDA	A4
INT	D2

Para ligar o XBee e o acelerômetro ao Arduino procedeu-se da seguinte forma: a alimentação do módulo XBee foi conectada diretamente no pino 3.3V do Arduino e o pino 3 (para recepção de dados – RX) ao pino de transmissão (TX) do Arduino, visto que todas as configurações de firmware já haviam sido realizadas. Na Figura 5.9 é possível ver a imagem do protótipo com esses três componentes.

Devido ao fato de que o dispositivo ficará acoplado ao trem e que o mesmo sofre vibrações, decidimos desenvolver uma placa de circuito impresso para que todos os componentes fiquem soldados à placa. O uso do circuito impresso ajuda a dar ao sistema uma característica mais profissional e de produto final ao módulo desenvolvido. Além dos três componentes essenciais citados anteriormente, foi acoplado ao módulo um relógio do tipo RTC (*Real Time Clock*) e um módulo SD para leitura e escrita em cartões de memória. Esses dois dispositivos foram

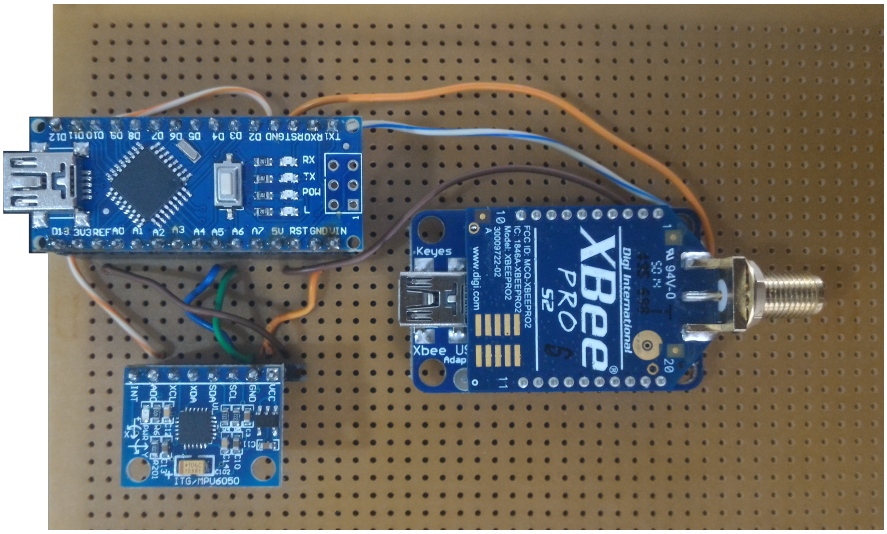


Figura 5.9 - Protótipo final desenvolvido. Fonte: Dados produzidos pelo autor.

utilizados para a graação de dados localmente com a hora e data de aquisição, e posteriormente comparar com os dados recebidos via rede zigbee. Por fim. O protótipo final foi embutido dentro de uma caixa protetora e instalado na parte interna do trem, próximo ao módulo já existente do projeto RailBee como pode ser visto na Figura 5.10.

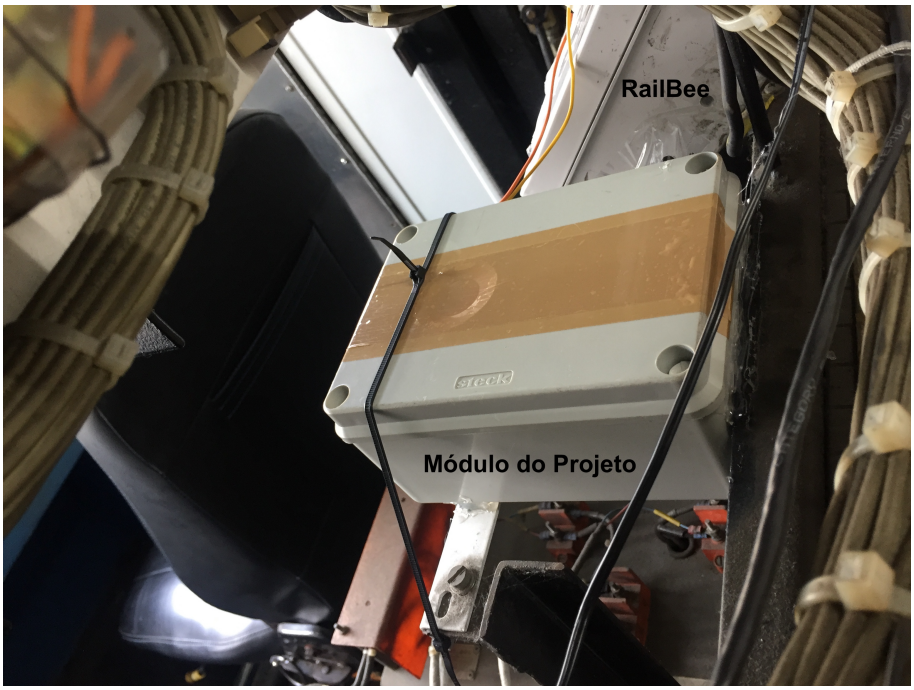


Figura 5.10 Caixa com módulo final acoplado ao trem para realização dos testes. Fonte: Dados produzidos pelo autor.

Como forma de verificar se o dispositivo estava funcionando corretamente após a montagem do protótipo, realizamos testes locais. Para isso, configuramos um módulo XBee como coordenador e acoplamos o mesmo ao computador usando o *explorer* e monitoramos a recepção de dados com o software XCTU (disponível pela próprio fabricante). Embora nessa etapa inicial apenas dados em hexadecimal estavam sendo recebidos, foi possível analisar os valores

enviados pelo acelerômetro verificando que os mesmos condiziam com o esperado. A Figura 5.11 mostra um exemplo de como procederam os testes iniciais com o módulo instalado no trem.

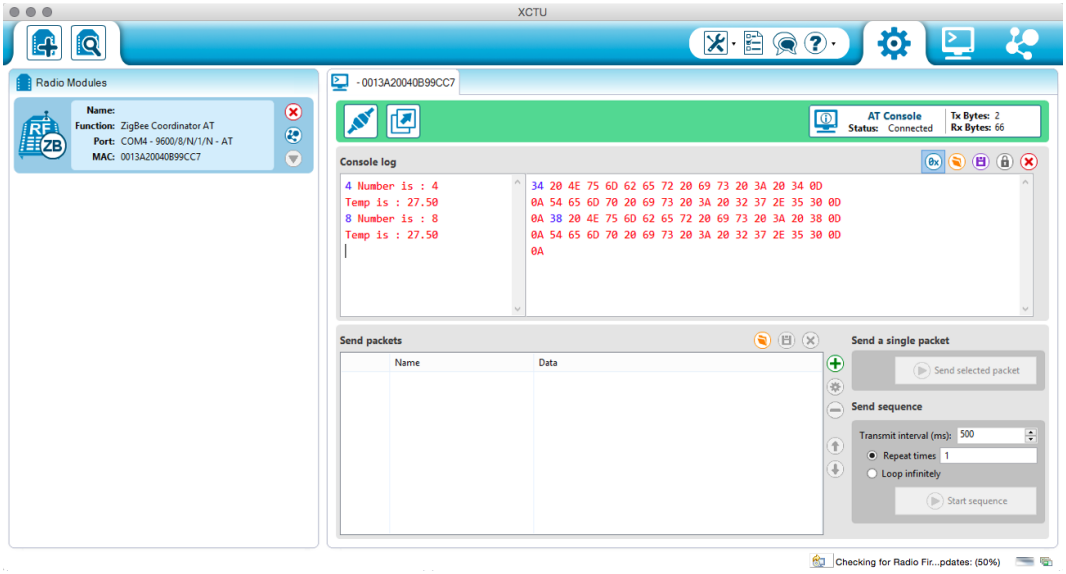


Figura 5.11 - Testes de recepção de dados com o XCTU. Fonte: Dados produzidos pelo autor.

CAPÍTULO 6

RESULTADOS

Até o capítulo anterior foi mostrado que a idéia de se utilizar acelerômetros e uma rede de sensores sem fios de alta qualidade é viável, de baixo custo e que melhora consideravelmente as diversas atividades já desenvolvidas nas centrais de controle de sistemas metroferroviários, além da possibilidade de uma nova gama de atividades baseadas em novas informações. Neste capítulo, o objetivo é ir além das pesquisas, desenvolvimentos, protótipos, implementações e testes. Aqui, serão mostradas as possibilidades que podem surgir ao se inferir novas informações e conhecimento a partir dos dados obtidos. Portanto, neste capítulo de resultados, as três fases de análise de dados são debatidas e apresentadas descrevendo os passos executados, as inferências de novas informações e abrindo questionamentos para diversos tópicos de pesquisa e desenvolvimento possibilitado a partir do projeto aqui desenvolvido.

6.1 TRATAMENTO DOS DADOS

Ao se carregar os dados do arquivo de período 24/10 à 02/11 o primeiro problema encontrado foi relacionado ao número extremamente grande de instâncias geradas. Isso ocorreu devido a taxa de amostragem trabalhada no sistema que foi a mais elevada possível. Vale destacar que em um sistema final esta taxa será menor, mas o objetivo de uma grande frequência de aquisição foi avaliar as possíveis perdas de pacotes na RSSF. Para os 9 dias de aquisição, mesmo com os problemas técnicos enfrentados, foram gerados 590822 instâncias, sendo impossível utilizá-las desta maneira, pois a capacidade de processamento disponível não seria suficiente. Esse problema esteve presente durante todas as análises realizadas. A Figura 6.1 mostra um exemplo do carregamento das diversas instâncias no Weka.

A primeira estratégia realizada para redução dos dados foi baseada nos horários de funcionamento. Como o horário máximo de funcionamento dos trens é de 05:00 horas da manhã às 23:00 horas da tarde, os dados após este horário foram removidos para esta análise. É importante chamar atenção que esses dados não deixam de se importantes, caso queira se saber como o trem se comporta durante o repouso do mesmo, se há oscilações de valores, se há diferenças

	Dia	Mês	Ano	Hora	Minutos	Segundos	acelX	acelY	acelZ	giroX	giroY	giroZ
1	24	10	2016	13	12	48	-0.01677	0.02635	9.82677	0.00000	-0.04580	0.00000
2	24	10	2016	13	12	49	-0.01677	0.02635	9.82677	0.00000	-0.04580	0.00000
3	24	10	2016	13	12	50	-0.01677	0.02635	9.82677	0.00000	-0.04580	0.00000
4	24	10	2016	13	12	51	0.00419	0.00898	9.79443	-0.02290	0.00000	0.00000
5	24	10	2016	13	12	52	0.00419	0.00898	9.79443	-0.02290	0.00000	0.00000
6	24	10	2016	13	12	53	-0.00898	0.00359	9.80880	0.03053	0.01527	0.01527
7	24	10	2016	13	12	54	-0.00898	0.00359	9.80880	0.03053	0.01527	0.01527
8	24	10	2016	13	12	55	-0.01557	-0.01736	9.80581	0.03053	0.04580	0.00763
9	24	10	2016	13	12	56	-0.01557	-0.01736	9.80581	0.03053	0.04580	0.00763
10	24	10	2016	13	12	57	-0.00599	-0.00958	9.81479	0.05344	0.01527	0.04580
11	24	10	2016	13	12	58	-0.00599	-0.00958	9.81479	0.05344	0.01527	0.04580
12	24	10	2016	13	12	59	-0.01138	0.00060	9.81419	-0.00763	-0.01527	0.02290

Showing 1 to 12 of 590,822 entries

Figura 6.1 - Exemplo das instâncias da base de dados. Fonte: Dados produzidos pelo autor (2017)

na pressão das bolsas de ar, se algum parâmetro não está de acordo para que o trem inicie seu funcionamento, entre outras medidas. Após essa primeira poda de dados o número de instâncias reduziu para 445.600.

Como o número de instâncias ainda era muito alto para análise no Weka, outras duas opções foram abordadas como possíveis para redução: criação de um subset de dados que pudessem representar a distribuição geral, e, transformar os dados em quadros para que cada instância passasse a representar um conjunto de 5 segundos, diminuindo assim em $\frac{1}{5}$ o número de instâncias. Não obstante, a criação de quadros de 5 segundos demonstraram afetar a identificação de informações durante os algoritmos de clusterização, portanto, a criação de um subset destacou-se como abordagem ideal.

Antes da criação do subset foi necessário realizar algumas operações. Primeiramente analisou-se a presença de *outliers* nos dados. Para isto a linguagem R foi utilizada com o intuito de criar-se um gráfico de *boxplots* para os atributos do acelerômetro e do giroscópio. Esse gráfico é mostrado na Figura 6.2. Como é possível perceber pelo gráfico o atributo **acelZ** tem altos valores, contudo esse atributo tende a medir sempre a gravidade que o acelerômetro sofre, portanto foi possível eliminar valores irreais.

Após a limpeza do atributo acelZ foi gerado um novo gráfico mostrado na Figura 6.3 que mostra que os atributos acelX e acelY também possuem *outliers*. Realizou-se portanto uma nova limpeza para eliminar os dados incorretos. Por fim, a base de dados ficou formada por dados cujo os valores fazem sentido para o ambiente de mineração. Um novo gráfico para esses dados é mostrado na imagem 6.4

Depois da eliminação de *outliers*, a linguagem R foi utilizado para gerar estatísticas sobre o conjunto de dados. Tais estatísticas sobre o conjunto de dados ajudaram principalmente em dois pontos. O primeiro foi determinar os requisitos e especificidades que o subset deve ter para representar bem o conjunto de dados de que foi retirado, por vezes até determinando o algoritmo que será usado para gerar o subset. Em segundo, para verificar se aquele subset é válido. Para isso, é necessário verificar se as características dele correspondem a do conjunto

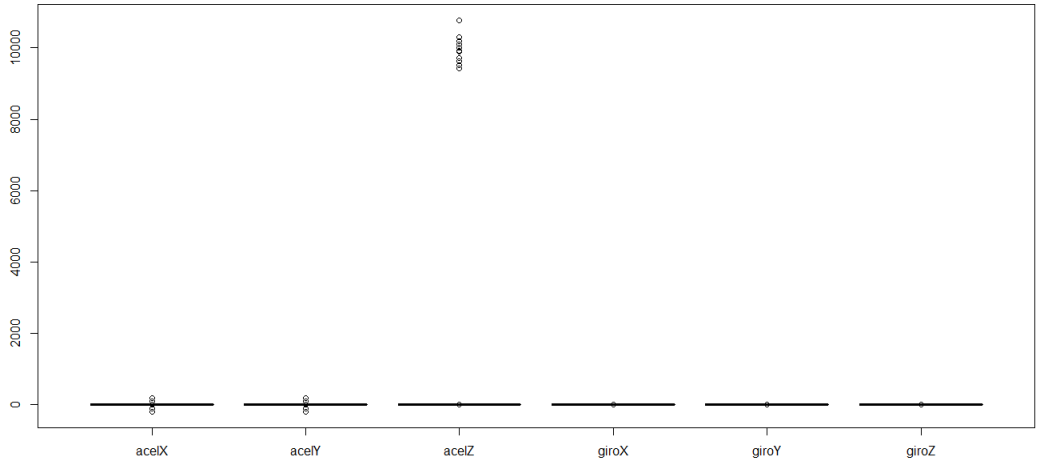


Figura 6.2 Boxplot dos atributos do acelerômetro de giroscópio. Fonte: Dados produzidos pelo autor (2017).

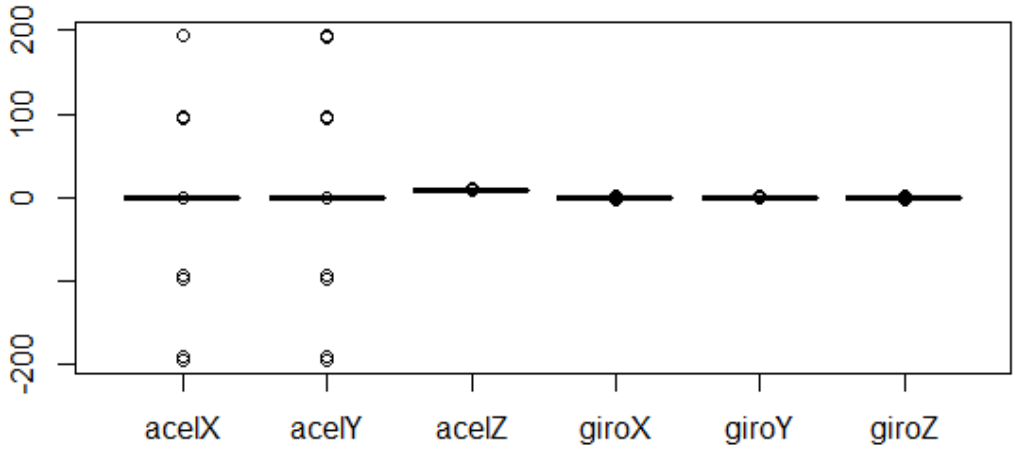


Figura 6.3 - BBoxplot dos atributos sem *outliers*. Fonte: Dados produzidos pelo autor.

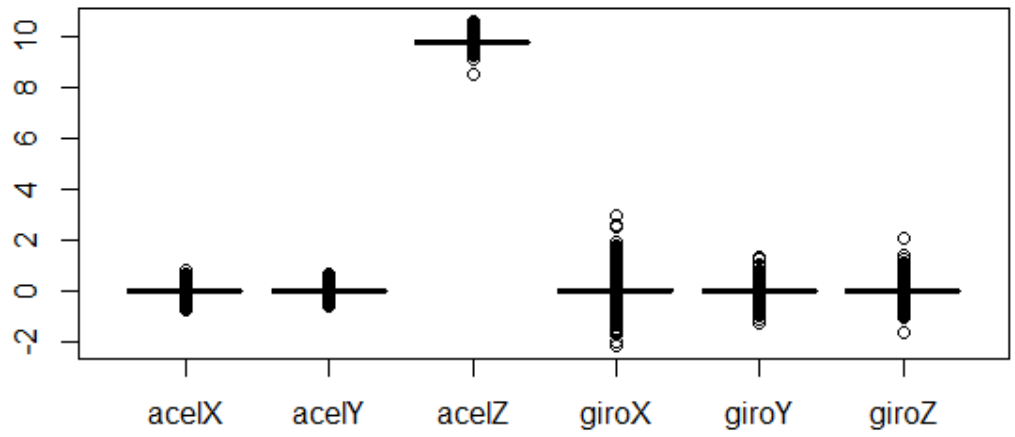


Figura 6.4 - Boxplot dos atributos sem *outliers*. Fonte: Dados produzidos pelo autor (2017).

de dados original, se sim ele pode ser utilizado, se não ele deve ser descartado e um novo subset gerado. A Figura 6.5 apresenta as estatísticas geradas.

Durante a análise percebeu-se que existiam valores negativos para a aceleração em X e Y. Isto é possível pois o sinal só representa para eles a direção para onde o trem está indo em relação a calibração do acelerômetro. Como no MetroRec o trem faz o percurso de ida e volta sem realizar retornos esses dados acontecem em cada retorno de viagem. O problema é para a criação das estatísticas esses dados poderiam estar sendo vistos de forma errônea, admitidos como uma aceleração negativa, sendo assim decidiu-se por colocar as colunas acelX e acelY em seus valores absolutos e uma nova estatística foi gerada e utilizada para comparar com a do subset. Apesar do mesmo acontecer para o giroscópio a informação do sinal é importante para a análise das suas informações.

```
      acelX              acelY              acelZ             giroX
 Min.   :-0.75323   Min.   :-0.631690   Min.   : 8.551   Min.   :-2.152670
 1st Qu.:-0.01976   1st Qu.:-0.000600   1st Qu.: 9.787   1st Qu.:-0.007630
 Median :-0.01078   Median : 0.005390   Median : 9.796   Median : 0.007630
 Mean   :-0.01041   Mean   : 0.004699   Mean   : 9.796   Mean   : 0.008476
 3rd Qu.:-0.00299   3rd Qu.: 0.010180   3rd Qu.: 9.806   3rd Qu.: 0.022900
 Max.   : 0.81970   Max.   : 0.611330   Max.   :10.592   Max.   : 2.984730
      giroY              giroZ
 Min.   :-1.30534   Min.   :-1.67939
 1st Qu.:-0.03817   1st Qu.:-0.04580
 Median :-0.03053   Median :-0.02290
 Mean   :-0.02674   Mean   :-0.02428
 3rd Qu.:-0.01527   3rd Qu.:-0.00763
 Max.   : 1.33588   Max.   : 2.06870
```

Figura 6.5 Estatísticas da base de dados, como maior e menor valores, media e mediana para cada atributo. Fonte: Dados produzidos pelo autor (2017).

Para criar a amostra foi utilizada a linguagem R, a base de dados utilizada uma amostra de 100000 instâncias foi gerada de forma aleatória e sem que uma mesma instância aparecesse mais de uma vez. Para verificar se a amostra foi comparada as suas estatísticas com a da base completa, além disso foi utilizado foi gerado histogramas de alguns atributos para verificar se a distribuição continuava a mesma, se compararmos então a imagem 6.6 com a imagem 6.7 podemos ver que a alteração foi muito pequena e que a amostra representa bem a base.

```
      acelX              acelY              acelZ             giroX
 Min.   :0.00000   Min.   :0.00000   Min.   : 8.551   Min.   :-2.152670
 1st Qu.:0.00599   1st Qu.:0.00359   1st Qu.: 9.787   1st Qu.:-0.007630
 Median :0.01317   Median :0.00719   Median : 9.796   Median : 0.007630
 Mean   :0.01685   Mean   :0.01517   Mean   : 9.796   Mean   : 0.008476
 3rd Qu.:0.02156   3rd Qu.:0.01377   3rd Qu.: 9.806   3rd Qu.: 0.022900
 Max.   :0.81970   Max.   :0.63169   Max.   :10.592   Max.   : 2.984730
      giroY              giroZ
 Min.   :-1.30534   Min.   :-1.67939
 1st Qu.:-0.03817   1st Qu.:-0.04580
 Median :-0.03053   Median :-0.02290
 Mean   :-0.02674   Mean   :-0.02428
 3rd Qu.:-0.01527   3rd Qu.:-0.00763
 Max.   : 1.33588   Max.   : 2.06870
```

Figura 6.6 - Estatísticas da amostra da base de dados. Fonte: Dados produzidos pelo autor (2017).

```
      acelX             acelY             acelZ             giroX
Min.   :0.00000   Min.   :0.00000   Min.   : 9.107   Min.   :-2.152670
1st Qu.:0.00599   1st Qu.:0.00359   1st Qu.: 9.787   1st Qu.:-0.007630
Median :0.01317   Median :0.00778   Median : 9.796   Median : 0.007630
Mean   :0.01686   Mean   :0.01517   Mean   : 9.796   Mean   : 0.008277
3rd Qu.:0.02156   3rd Qu.:0.01377   3rd Qu.: 9.806   3rd Qu.: 0.022900
Max.   :0.81970   Max.   :0.61133   Max.   :10.592   Max.   : 2.984730
      giroY             giroZ
Min.   :-1.30534   Min.   :-1.09160
1st Qu.:-0.03817   1st Qu.:-0.04580
Median :-0.03053   Median :-0.02290
Mean   :-0.02682   Mean   :-0.02435
3rd Qu.:-0.01527   3rd Qu.:-0.00763
Max.   : 1.33588   Max.   : 1.40458
```

Figura 6.7 Segunda amostra de estatísticas da base de dados. Fonte: Dados produzidos pelo autor (2017).

6.2 ANÁLISE DOS DADOS - PRIMEIRO CONJUNTO DE TESTES

Esta seção tem como foco os dados referentes a primeira aquisição durante os meses de outubro e novembro de 2016. Estes dados foram explicados na metodologia e utilizam informações de velocidade do trem. Os testes tiveram como principal objetivo analisar as paradas do trem relacionando-as com a velocidade e verificar a possibilidade de estimar a quantidade de entrada de passageiros de acordo com o tempo. Se os dados de pressão nas bolsas de ar houvessem sido disponibilizados para estes algoritmos, seria possível encontrar valores mais exatos e aumentar o poder de correlação encontrado. A subseção está dividida para representar cada algoritmo utilizado. No final, um espaço é destinado para a discussão e comparação dos resultados.

6.2.1 Algoritmo de Classificação - IBk

O algoritmo IBk é um algoritmo que implementa um modelo de mineração de dados considerando a técnica do vizinho mais próximo *Nearest Neighbor*. Esta técnica de classificação procura utilizar as instâncias antigas armazenadas em bases de dados, onde se é conhecido os valores de saída, para predizer uma saída desconhecida de um uma nova instância de dados (Ali and Smith, 2006). Diferente de outras técnicas, o vizinho mais próximo pode ser usado com diversos tipos de dados, não ficando restrito a dados numéricos. Além disso, a técnica é escalável para bases de dados de diferentes tamanhos, sendo mais eficiente para bancos de dados maiores do que técnicas que utilizam árvores.

Por ser um algoritmo de classificação, o mesmo requer que a base de dados seja dividida em duas partes: a primeira composta preferencialmente por uma quantidade de 60% a 70% dos dados, de forma a ser usada para o treinamento do modelo. A segunda, com menor número de dados, é utilizada para composição do conjunto teste, utilizado para testar o modelo. Para este, utilizou-se a opção de *percentage split* que divide a base aleatoriamente em uma determinada

porcentagem para criar o modelo, e o restante dos dados são utilizados para testes. Para o IBk, 70% da base foi selecionada automaticamente para ser utilizada na criação do modelo e 30% para os testes, sendo 230 e 98 instâncias dos dados respectivamente. Utilizando as configurações iniciais do algoritmo foi selecionado somente um vizinho, ou seja somente o dado mais próximo irá contar para a dedução do desconhecido. Os resultados são mostrados na Figura 6.8.

```
=== Evaluation on test split ===

Time taken to test model on training split: 0.01 seconds

=== Summary ===

Correctly Classified Instances          64               65.3061 %
Incorrectly Classified Instances        34               34.6939 %
Kappa statistic                          0.4572
Mean absolute error                      0.1769
Root mean squared error                  0.413
Relative absolute error                 55.0475 %
Root relative squared error            102.3094 %
Total Number of Instances               98

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0,800    0,036    0,800      0,800    0,800      0,764    0,882     0,671     1
               0,694    0,347    0,667      0,694    0,680      0,347    0,673     0,616     2
               0,500    0,194    0,481      0,500    0,491      0,302    0,653     0,373     3
               0,625    0,000    1,000      0,625    0,769      0,778    0,813     0,656     4
Weighted Avg.  0,653    0,231    0,665      0,653    0,655      0,434    0,711     0,563

=== Confusion Matrix ===

  a  b  c  d   <-- classified as
 12  3  0  0 |  a = 1
  3 34 12  0 |  b = 2
  0 13 13  0 |  c = 3
  0  1  2  5 |  d = 4
```

Figura 6.8 Resultado do algoritmo IBk após a construção e teste do modelo. Fonte: Dados produzidos pelo autor

A partir dos resultados é possível perceber que o algoritmo não obteve muito sucesso. A taxa de classificação correta das instâncias alcançou a marca dos 65% o que é uma taxa baixa e que tende a mostrar que as decisões tomadas a partir das informações geradas pelo modelo poderiam estar erradas, causando diversos problemas no futuro. Como pode ser visto na *Confusion Matrix* da figura, que é a matriz que mostra como os elementos foram classificados nas etiquetas, a classificação na etiqueta do tipo 3 foi a mais afetada por erros, mostrando que o algoritmo teve dificuldades em classificar instâncias que possuíam tempo superior a 30s e menor que 60s de parada na estação.

Para se verificar a pouca eficiência do modelo de forma definitiva foi realizada uma mudança nos parâmetros iniciais do algoritmo. A mudança focou no aumento dos números de vizinhos necessários para a classificação dos dados com resultados desconhecidos. Foram testados com três valores diferentes, sendo eles dois ,três e quatro vizinhos. Os resultados obtidos foram bem interessantes, com dois vizinhos a taxa de classificação correta subiu 3%, com três a taxa subiu em 5%, contudo ao se testar com quatro a taxa de classificação caiu em 1%. Para ter um maior grau de certeza nos resultados os testes foram replicados com o número de vizinhos variando de cinco a dez. Os resultados obtidos se mantiveram estáveis, com taxas

de classificação correta subindo inicialmente para 69% e depois voltando a cair. Os resultados indicaram que o melhor número de vizinhos para este conjunto de dados são três.

Uma observação importante a se fazer em relação aos resultados destes testes é que mesmo sendo o vizinho mais próximo considerado um dos melhores algoritmos de classificação, o modelo construído para este específico conjunto de dados não é tão eficiente, mostrando que os tipos de dados e as relações entre eles são fator chave para determinar os algoritmos e modelos a serem utilizados. Vale ressaltar também que essa ineficiência encontrada neste algoritmo pode ser alterada a partir da entrada de novos dados ou atributos.

6.2.2 Árvore de Decisão - Algoritmo J48

O algoritmo J48 foi criado a partir de uma recodificação do algoritmo C4.5 escrito em C para uma versão escrita em Java. Este algoritmo de classificação trabalha na construção de árvores de decisão através de um conjunto de dados de treinamento criando assim o modelo, que por sua vez é utilizado para classificar os dados em um conjunto teste (Librelotto, 2014). Uma árvore de decisão é uma estrutura baseada em uma tabela de decisão estruturada em forma de árvore. Esta estrutura é utilizada como uma maneira de expressar quais conjuntos de condições são necessários ocorrer para que determinadas ações aconteçam (Kohavi, 1995).

O J48 é considerado o algoritmo de melhor resultado para a construção de árvores a partir de um agrupamento de dados de treinamento. Utilizando a abordagem dividir-para-conquistar, o algoritmo divide um problema complexo em subproblemas mais simples, aplicando este padrão de maneira recursiva em cada subproblema e dividindo o espaço definido pelos atributos em subespaços, associando a eles uma classe (Librelotto, 2014), citepj48.2.

Por ser um modelo de classificação a base de dados também precisa ser dividida em duas. Uma parte para a construção do modelo composta por preferencialmente 60% a 70% dos dados, de forma a ser usada para o treinamento, nesse caso a construção da árvore. Outra, com menor quantidade de dados, para realização dos testes do modelo. Dessa forma, nossa base de dados composta por 328 instâncias foi subdividido em um conjunto com aproximadamente $\frac{1}{3}$ dos dados composto de 110 instâncias, sendo portanto o conjunto testes, e um conjunto com $\frac{2}{3}$ dos dados, compostos por 218 instâncias que serão utilizadas para compor o conjunto de treinamento. Além dessa divisão da base, foi necessária a inserção de um novo atributo chamado de flow_estimation. Esse atributo funciona como classificador do tempo em que o trem passa em uma parada, considerando que quanto mais tempo ele permanece na parada, maior é o fluxo de pessoas entre a plataforma e o trem (entrada e saída de passageiros).

Na primeira execução do algoritmo utilizando o conjunto de dados para a construção dos modelos, foi retornando que as instâncias foram classificadas corretamente com uma taxa de 100%. Isso indicaria que o modelo é incrivelmente potente para classificar dados da base com relação a etiqueta apresentada, como mostrado na figura 6.9.

```
=== Evaluation on training set ===

Time taken to test model on training data: 0.02 seconds

=== Summary ===

Correctly Classified Instances         218               100      %
Incorrectly Classified Instances         0                 0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances              218

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1,000    0,000    1,000      1,000    1,000      1,000    1,000     1,000     1
                 1,000    0,000    1,000      1,000    1,000      1,000    1,000     1,000     2
                 1,000    0,000    1,000      1,000    1,000      1,000    1,000     1,000     3
                 1,000    0,000    1,000      1,000    1,000      1,000    1,000     1,000     4
Weighted Avg.    1,000    0,000    1,000      1,000    1,000      1,000    1,000     1,000

=== Confusion Matrix ===

   a   b   c   d   <-- classified as
  25   0   0   0 |   a = 1
   0 126   0   0 |   b = 2
   0   0  48   0 |   c = 3
   0   0   0  19 |   d = 4
```

Figura 6.9 - Resultado do algoritmo J48 na etapa de construção. Fonte:Dados produzidos pelo autor

Contudo, uma precisão tão alta gerou dúvidas sobre a eficiência do modelo criado, já que existem algoritmos de classificação que podem acabar criando modelos perfeitos para uma base de dados, mas que não conseguem classificar corretamente novos dados. Apesar das dúvidas sobre o modelo criado, aplicou-se o conjunto de dados testes para verificar a eficiência do classificador ao analisar estes dados. Nesta execução cerca de 99% das instâncias foram classificadas corretamente, sendo somente 1% não classificadas.

Neste caso os excelentes resultados de classificação continuaram sendo obtidos após a passagem do conjunto teste pelo modelo. Contudo, ainda desconfiou-se de tão bons resultados e foi decidido realizar uma análise mais minuciosa da árvore gerada pelo modelo, que é mostrada na Figura 6.10. A árvore mostra que o modelo conseguiu encontrar que o atributo time_at_station era usado para categorizar as etiquetas e achou suas delimitações, podendo observar que existe uma diferença na delimitação da etiqueta 3 determinada na Tabela /refparadas que deveria ser valores menores de 60 e não 54, como encontrado pelo modelo.

A partir da análise da árvore e dos resultados obtidos pelo teste, foi possível afirmar que o modelo está correto e funcionando da forma esperada. Contudo, as informações resultadas poderiam ter ainda maior relevância para a pesquisa se o número de instâncias e atributos disponíveis fosse maior. Isso possibilitaria a construção de um modelo e de um conjunto de testes mais preciso.

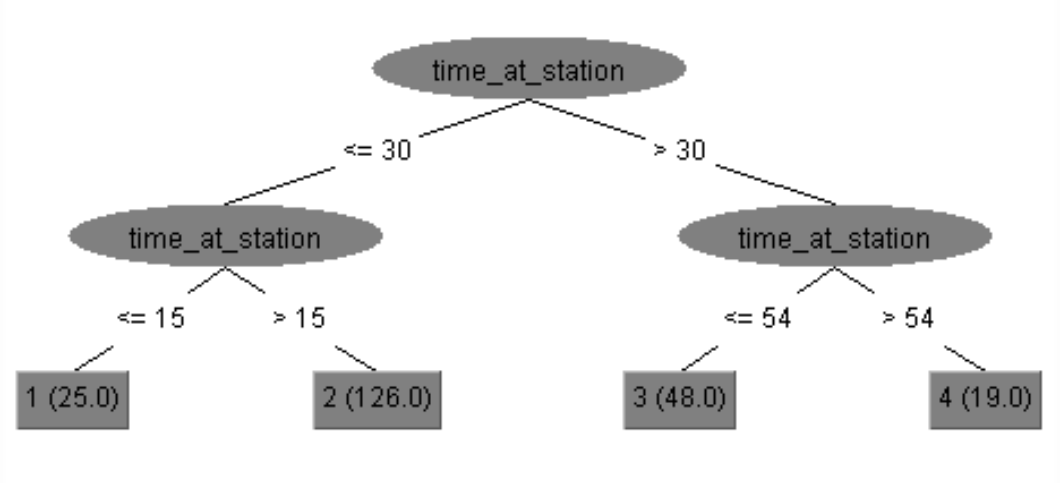


Figura 6.10 - Árvore de decisão criada pelo WEKA. Fonte: Dados produzidos pelo autor

6.2.3 Clusterização - Algoritmo *SimpleKMeans*

Algoritmos de clusterização tem como principal tarefa agrupar as instâncias em grupos a partir das similaridades dos dados dos seus atributos. Uma vantagem da sua utilização é que na clusterização cada atributo da base será utilizado para analisar os dados (Jain, 2010). Uma das maiores desvantagens do processo é que é necessário que o utilizador forneça o número de grupos a serem formados, sendo desta forma um pouco mais difícil para iniciantes em mineração identificar um número de grupos ótimo na primeira execução do algoritmo, partindo para a execução de inúmeras tentativas com quantidades de grupos diferentes. Após o algoritmo receber o número de grupos, ele seleciona da base de dados instâncias iguais ao número de grupos requeridos. Essas instâncias serão utilizadas como os centros dos grupos. A partir daí, a clusterização realiza a análise da distância entre uma instância e os centros dos grupos e a insere no grupo com a menor distância. Contudo, o centro do grupo não é estático e a cada nova alocação de uma nova instância é preciso realizar novos cálculos para os centróides dos grupos (Jain, 2010).

Para a utilização do algoritmo *SimpleKMeans* algumas mudanças são requeridas na base de dados. O algoritmo não permite que atributos não-numéricos sejam utilizados como entrada. Dessa forma, a ferramenta *Ignore attributes*, presente na interface de clusterização do WEKA, foi utilizada para retirar os atributos que continham datas e horários. Outra alteração feita foi a retirada do atributo de estimativa de fluxo. Este atributo funcionava como uma etiqueta, categorizando as instâncias. Esta categorização é extremamente necessária para algoritmos de classificação mas não é útil para a clusterização da nossa base de dados. Como dito anteriormente existe uma dificuldade inicial em se determinar o número de grupos a serem formados, foi então determinado a realização de três testes, seguindo o seguinte número de grupo: dois, três e quatro. Todos os testes e grupos são exibidos na imagem 6.11.

Analisando o primeiro resultado com dois grupos podemos perceber que dos cinco atributos três deles possuem valores próximos em ambos os grupos. Contudo, é possível perceber

```
Final cluster centroids:
                                    Cluster#
Attribute               Full Data          0          1
                          (328.0)    (111.0)    (217.0)
=========================================================
time_at_station           35.6189     34.973    35.9493
braking_time              44.5793    49.9369    41.8387
braking_speed              48.128     49.009    47.6774
acceleration_time         30.5549    50.3694    20.4194
speed_at_acceleration     26.3659    51.3243    13.5991
```

```
Final cluster centroids:
                                    Cluster#
Attribute               Full Data          0          1          2
                          (328.0)    (106.0)    (108.0)    (114.0)
=================================================================
time_at_station           35.6189    34.8868    34.1204    37.7193
braking_time              44.5793       51.5     33.037    49.0789
braking_speed              48.128    50.8302    34.9907    58.0614
acceleration_time         30.5549    50.3868    19.5278    22.5614
speed_at_acceleration     26.3659    51.9811    14.9537    13.3596
```

```
Final cluster centroids:
                                    Cluster#
Attribute               Full Data          0          1          2          3
                          (328.0)    (104.0)     (99.0)     (91.0)     (34.0)
============================================================================
time_at_station           35.6189      35.25    35.0606    37.5275    33.2647
braking_time              44.5793    52.2308    38.4444    50.2967    23.7353
braking_speed              48.128    51.3365    41.7879    60.6154    23.3529
acceleration_time         30.5549    47.4327    11.5354    26.1429    46.1176
speed_at_acceleration     26.3659    52.1635    13.1111    13.5495    20.3529
```

Figura 6.11 Grupos formado pelo algoritmo *SimpleKMeans* na ordem de dois, três e quatro grupos. Fonte: Dados produzidos pelo autor

uma grande diferença entre os grupos nos atributos acceleration_time e speed_at_acceleration. O atributo acceleration_time representa o tempo de aceleração, ou seja, o tempo que um trem leva da saída da parada até alcançar um pico de velocidade e voltar a desacelerar. Já o atributo speed_at_acceleration representa este pico de velocidade. É possível perceber que existe um padrão ocorrendo entre esses dados e que será descrito no próximo parágrafo. Nas subsequentes execuções do algoritmo é possível perceber que outro padrão vai surgindo a partir dos atributos braking_time, tempo levado desde um pico de velocidade até que a mesma chegue à zero, e braking_speed, velocidade em que o ytrêm se encontrava no momento em que iniciou a desaceleração.

Analisando o padrão reconhecido pelo pico de velocidade e o tempo de aceleração, podemos perceber que a diferença entre os grupos no que diz respeito ao pico de velocidade não se altera muito, mesmo com o aumento do número de grupos. Assim é possível usar o resultado da clusterização de dois grupos para demonstrar o padrão encontrado. É perceptível que 217 instâncias, equivalente a 66% dos dados, tem como pico de velocidade médio 13,5Km,

esta informação é importante pois mostra que na maior parte das vezes que um trem sai de uma parada ele não acelera até uma velocidade alta, isso pode indicar que existe algo que faça com que os trens sejam obrigados a diminuir sua velocidade, podendo ser problemas nos trilhos, nas estações ou no percurso em si. Além disso outra conclusão que pode ser elencada a partir dessa informação é que o fato de o trem não alcançar uma velocidade mais alta pode levar a um maior tempo entre estações, provocando atrasos e causando desconforto aos passageiros.

Outros pequenos padrões também foram identificados pelo processo. A partir da clusterização em quatro grupos foi possível concluir que os trens passam em média 30s em cada parada e possuem uma desaceleração média de 1Km/s. O grupo 3 é formado por instâncias que possuem um tempo de velocidade muito alto para chegar a um pico de velocidade baixo. Este fato pode indicar que essas paradas são as que sofrem com os possíveis problemas descritos no padrão explicitado no parágrafo anterior.

Como mencionado, os padrões encontrados podem indicar uma série de informações, mas não garantem um elevado grau de certeza. Para que se tenha certeza sobre a veracidade dos padrões encontrados seria necessária uma base de dados muito maior que a utilizada, composta por mais dias e também por mais trens. Portanto as informações retiradas neste momento servem como indicativos para que os gestores analisem se esses padrões realmente ocorrem e possam tomar providências sobre eles.

6.2.4 Regras de Associação - Algoritmo Apriori

As regras de associação representam padrões que podem ser identificados a partir de transações já armazenadas na base de dados. Seguindo a premissa da tarefa de associação, a construção de regras procura relacionamentos entre atributos da base de dados com determinado grau de certeza (De Vasconscelos, 2004). Isto é, as regras buscam encontrar elementos que implicam na presença de outros elementos na transação.

O algoritmo apriori, proposto em 1993, é um dos algoritmos de associação mais utilizados na literatura. O mesmo funciona a partir da realização de múltiplas passagens na base de dados obtendo diversas alternativas de combinação entre os atributos. O algoritmo realiza buscas sucessivas, pois utiliza um método com a mesma lógica da técnica dividir-para-conquistar executando um procedimento de indução de regras para todas as combinações possíveis de atributos (Librelotto, 2014).

O algoritmo requer que as instâncias da base de dados representem transações, sendo então necessário que a base de dados passasse por algumas mudanças. Primeiramente foi observado que seria impossível transformar os dados da base em transações, portanto a abordagem encontrada foi transformar todos os atributos para se tornarem etiquetas, como realizado com o atributo de estimativa de fluxo. Dessa forma, o algoritmo apriori, que não aceita dados numéricos, poderia ser executado sem problemas. Na primeira execução do algoritmo geradas dez regras de associação, podendo ser observadas na figura 6.12.

```
Apriori
=======

Minimum support: 0.1 (33 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 18

Generated sets of large itemsets:

Size of set of large itemsets L(1): 9

Size of set of large itemsets L(2): 14

Size of set of large itemsets L(3): 6

Best rules found:

 1. speed_at_acceleration=13 93 ==> data_moment=2016-11-26 93    <conf:(1)> lift:(1.67) lev:(0.11) [37] conv:(37.43)
 2. acceleration_time=7 67 ==> data_moment=2016-11-26 67    <conf:(1)> lift:(1.67) lev:(0.08) [26] conv:(26.96)
 3. acceleration_time=6 66 ==> data_moment=2016-11-26 66    <conf:(1)> lift:(1.67) lev:(0.08) [26] conv:(26.56)
 4. speed_at_acceleration=12 flow_estimation=2 42 ==> data_moment=2016-11-26 42    <conf:(1)> lift:(1.67) lev:(0.05) [16] conv:(16.9)
 5. speed_at_acceleration=13 flow_estimation=2 41 ==> data_moment=2016-11-26 41    <conf:(1)> lift:(1.67) lev:(0.05) [16] conv:(16.5)
 6. acceleration_time=7 flow_estimation=2 39 ==> data_moment=2016-11-26 39    <conf:(1)> lift:(1.67) lev:(0.05) [15] conv:(15.7)
 7. acceleration_time=6 speed_at_acceleration=12 38 ==> data_moment=2016-11-26 38    <conf:(1)> lift:(1.67) lev:(0.05) [15] conv:(15.29)
 8. acceleration_time=6 flow_estimation=2 34 ==> data_moment=2016-11-26 34    <conf:(1)> lift:(1.67) lev:(0.04) [13] conv:(13.68)
 9. acceleration_time=7 speed_at_acceleration=13 34 ==> data_moment=2016-11-26 34    <conf:(1)> lift:(1.67) lev:(0.04) [13] conv:(13.68)
10. speed_at_acceleration=12 79 ==> data_moment=2016-11-26 77    <conf:(0.97)> lift:(1.63) lev:(0.09) [29] conv:(10.6)
```

Figura 6.12 - Resultado do algoritmo apriori. Fonte: Dados produzidos pelo autor

Como é possível observar, as regras criadas pelo algoritmo são formadas pelas ligações entre atributos. Geralmente o primeiro passo é a análise do grau de certeza da regra, demonstrado pelo campo **conf** em cada regra. Contudo é possível perceber que as regras não possuem sentido lógico para as regras de associação dos atributos para os dias em que os dados foram colhidos. Dessa forma, é possível identificar que a utilização de tais dados geram regras inúteis como resultado.

De forma a tentar extrair novas regras o atributo data_moment foi retirado da base de dados e o algoritmo foi executado novamente. Os novos resultados demonstraram que com as alterações o algoritmo não conseguiu produzir nenhuma regra de associação. Um terceiro teste foi realizado eliminando-se os atributos arrive_time, live_time e flow_estimation, deixando somente os antigos dados numéricos da base e, novamente, nenhuma regra de associação foi gerada.

A partir dos resultados encontrados é possível afirmar que a base atual não possui dados interessantes para serem usados no algoritmo apriori. Não obstante, a obtenção e utilização de novos dados pode permitir que o algoritmo possa produzir regras de associação.

6.3 ANÁLISE DOS DADOS - SEGUNDO CONJUNTO DE TESTES

Para a segunda análise, apenas os dados de aceleração pertencentes ao nosso protótipo foram permitidas pelo MetroRec. As outras informações inerentes aos testes (velocidade, pressão nas bolsas de ar, etc.) ficaram em posse da equipe de gerência do metrô para análise e futura liberação. Devido a grande quantidade de dados obtidos e as inúmeras possibilidades de relacionamento entre os dados, os mesmos não foram liberados até o momento. Portanto, os algoritmos utilizados nesta segunda etapa de testes são essencialmente algoritmos de clusterização

que foram os que demonstraram resultados mais satisfatórios para os valores do acelerômetro e giroscópio.

Nesta segunda etapa os dados utilizados foram os mencionados no capítulo metodologia designados por CODA1 e CODA2. Os dados de CODA1 foram utilizados para validar as ideias encontradas na primeira etapa de testes com um número maior de instâncias. Já os dados do CODA2 compreendem o período entre 24/02/2017 e 04/03/2017 e são compostos dos 3 valores do acelerômetro e 3 valores do giroscópio obtidos do TUE 23 que funciona na linha sul.

A importância da análise dos dados do CODA2 foi devido a o mesmo ter sido realizado durante o período de carnaval. Com isso espare-se mostrar que a grande quantidade de pessoas, provocando a superlotação do trem, faz com que as variáveis das análises fiquem ainda mais em evidência. Além disso, demonstra a necessidade de um monitoramento constante e em tempo real, devido a essas modificções que as variáveis podem sofrer em situações peculiares, como feriados, dias festivos, campeonatos futebolísticos, etc.

Quatro algoritmos foram utilizados para realizar a clusterização das amostras, sendo eles: Hierárquico, Mean Shift, K-means e X-means. Para o Hierárquico tanto o software WEKA como a linguagem R foram utilizados. Para ambos o tamanho da amostra era muito grande para ser executado sendo que no WEKA o algoritmo não rodava, enquanto que através da linguagem R, o processamento era tomado por inteiro para a execução, causando o desligamento ou travamento da máquina de testes. Para o mean shift somente a linguagem R foi utilizada, já que não foi encontrado nenhum pacote para o WEKA que o adicionasse. Já os algoritmos K-means e X-means foram testados utilizando-se a ferramenta WEKA e permitiram a obtenção de dados mais consistentes, como será mostrado nos resultados a seguir. Os atributos de mês, ano, minuto, segundo e dia, assim como nos testes realizados na etapa 01, não foram utilizados durante a clusterização.

6.3.1 X-means

Como o X-means é um algoritmo que funciona sem a necessidade de explicitar o número de clusters ele foi executado primeiro para poder fornecer exatamente quantos clusters são ideais. Isso facilitou a estimativa de clusters necessários para o k-means, de forma que fosse possível executá-lo de forma ótima.

O X-means gerou 83 clusters que podem ser verificados na Figura 6.13, a partir da análise desse gráfico é possível verificar que os maiores cluters estão posicionados nos horários de pico dos metrô, sendo esse horário de 6hrs às 9hrs e de 17hrs às 19:30hrs. Isso é um curiosidade interessante, pois identifica que há a necessidade de uma análise mais profunda dos dados desses horários para estes clusters. Principalmente quando seja possível compará-los com outras informações do trem.

Analisando a tabela apresentada na Figura 6.14 a seguir é possível perceber que os maiores clusters além de estarem em um horário de pico também concentram uma acelX e

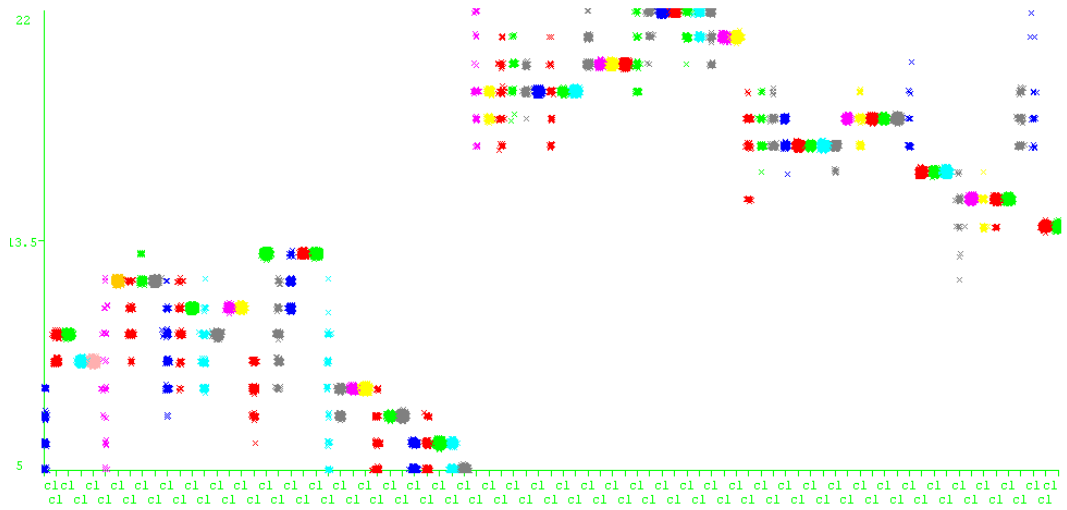


Figura 6.13 - Clusters do X-means analisados em relação a hora. Fonte: Dados produzidos pelo autor

acelY extremamente baixas se comparados a outros clusters, o que com as informações que foram encontradas podem levar a entender que o peso do vagão, possivelmente com maior número de usuários, dificulta a aceleração do trem.

	Cluster 26	Cluster 29	Cluster 32	Cluster 34
Hora	8	7	6	5
acelX	0.01422999382	0.01352761806	0.0138048535	0.01344568866
acelY	0.008038247529	0.007968560677	0.007754712709	0.00810838565
acelZ	9.791787753	9.789856499	9.789208588	9.791064854
giroX	0.01013447488	0.009418590781	0.008530964612	0.011636735
giroY	-0.02794819811	-0.02735141298	-0.02430050038	-0.02404304506
giroZ	-0.0259156425	-0.02189504233	-0.01855486301	-0.01913800128

Figura 6.14 - Clusters com o maior número de instâncias. Fonte: Dados produzidos pelo autor

O dado mais importante se encontra no gráfico da Figura 6.15. Nele podemos observar a acelZ em relação às horas de funcionamento. A importância desse dado se dá devido ao fato de que a acelZ mede sempre a gravidade e quando sofre alguma mudança por pequena que seja indica que ela está captando a vibração do trem, portanto nessa imagem podemos ver que existe uma maior alteração em horários de pico e isso é extremamente preocupante. O que era esperado era identificar que a acelZ fosse modificada pela a aceleração nas outras coordenadas, mas se a indicação for correta pode ser que o número de pessoas no trem esteja desestabilizando seu eixo o que pode levar a acidentes, ao desgastes das rodas e possíveis problemas ao realizar curvas, reduzindo a segurança e comodidade dos passageiros.

6.3.2 K-means

Como citado anteriormente, o número de clusters encontrado pelo x-means foi então utilizado no K-means, gerando assim também 83 clusters. Contudo, diferentemente do x-means, o

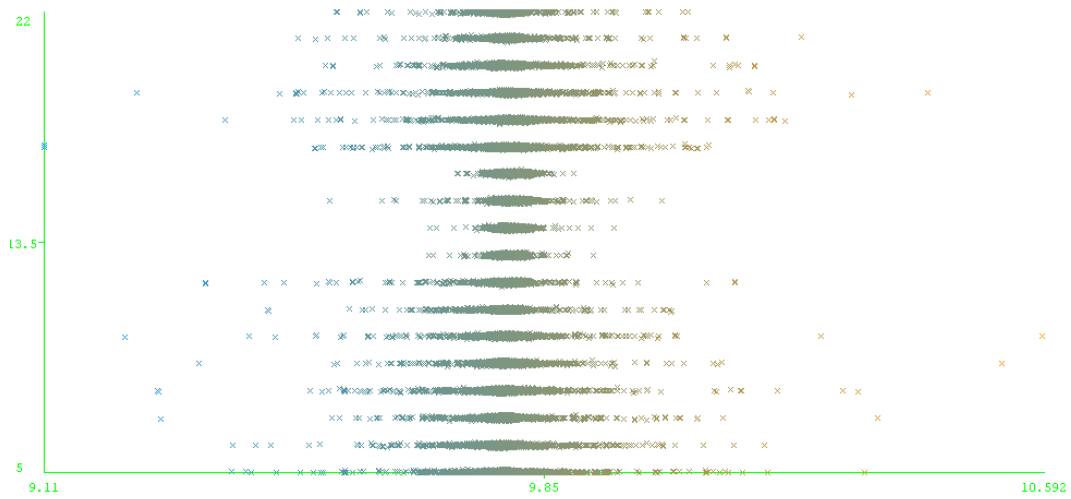


Figura 6.15 Clusters representados em Y pela Hora e em X pelo valor de acelZ. Fonte: Dados produzidos pelo autor

K-means criou alguns clusters com um número muito pequeno de instâncias e que demonstram ser extremamente interessantes para identificar os padrões entre esses dados, como mostrado na Figura 6.16. Na imagem 6.17 podemos ver que os clusters estão bem mais espalhados em relação a hora Diferentemente do X-means, seria complicado retirar alguma informação dessa relação portanto as análises foram mais concentradas nos clusters pequenos.

		Cluster#							
Attribute	Full Data	4	9	18	35	60	61	77	78	80
	(100000.0)	(86.0)	(54.0)	(48.0)	(57.0)	(27.0)	(47.0)	(76.0)	(22.0)	(82.0)
Hora	13.4445	6.0698	19.2222	7.9167	18.7719	9.8148	7.6383	10.8158	6.6818	14.7317
acelX	0.0169	0.1135	0.1378	0.1479	0.088	0.1702	0.3223	0.1261	0.413	0.1165
acelY	0.0152	0.0458	0.3144	0.2629	0.1008	0.1045	0.1143	0.0383	0.2086	0.0404
acelZ	9.7964	9.9533	9.9195	9.5864	9.5518	10.0633	9.8249	9.7888	9.9296	9.8015
giroX	0.0083	0.1778	0.2696	-0.5695	-0.4375	0.287	0.1551	0.034	0.0507	-0.019
giroY	-0.0268	-0.0109	-0.0164	-0.1298	0.0354	-0.0622	-0.129	0.007	0.1086	-0.0376
giroZ	-0.0243	0.3132	0.1373	-0.0988	-0.2214	0.0119	-0.3469	0.2439	0.483	-0.095

Figura 6.16 - Centróides dos clusters de menores instâncias. Fonte: Dados produzidos pelo autor

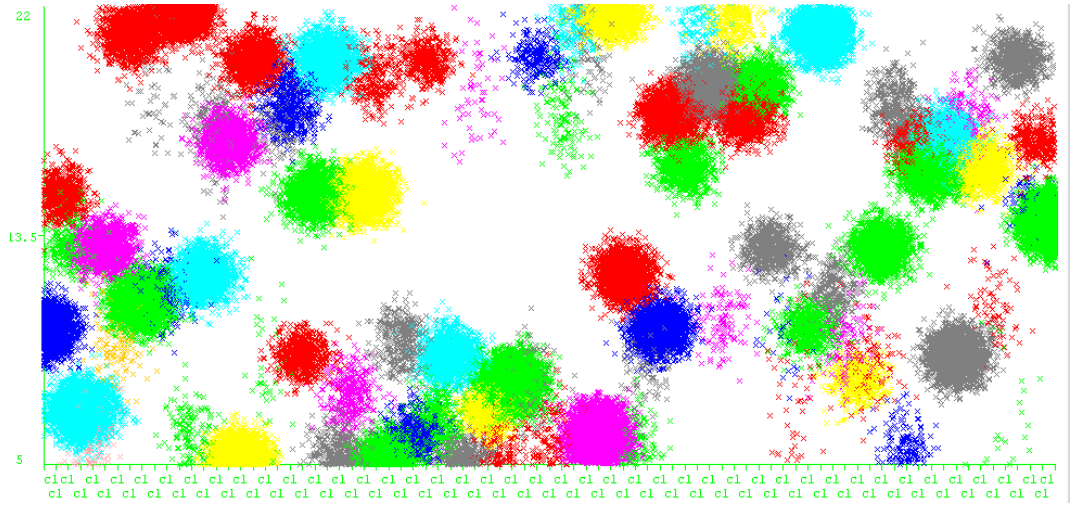


Figura 6.17 Clusters representados em Y pela Hora e em X pelo valor de acelZ. Fonte: Dados produzidos pelo autor

Após a análise dos clusters é possível perceber algumas características, como por exemplo o 9 e 78, mostram que a acelZ está acima do valor da gravidade o que como já foi citado pode indicar a vibração do trem. Contudo, ambos compartilham uma característica bem específica que é a de estarem com o valor de acelX ou de acelY bastante elevados. Isso pode identificar que ao realizar uma curva, a vibração nas outras dimensões também aumenta, causando esse efeito. A maior preocupação dessa informação é que se isso aumentar muito pode causar riscos de acidentes ao trem. Para análises melhor do limite de vibração, estudos mais aprofundados levando em consideração as características da curva precisam ser realizados.

Por outro lado, ao analisar-se os clusters 18 e 35 percebe-se o oposto. O valor de acelZ baixa em relação a gravidade, contudo a aceleração nos outros eixos parece normal, contudo se analisarmos giroX podemos perceber que ele está extremamente alto, isso é uma preocupação altíssima pois pode indicar problemas relacionados a manutenção do trilho que causa uma inclinação do trem ou até mesmo algum tipo de irregularidade no terreno onde o trilho está localizado.

Por último, temos a avaliação do cluster 60. Este cluster apresenta valores bastante intrigantes pois ele possui uma elevação dos valores de acelZ em relação à gravidade, mas não apresenta nenhuma justificativa que possa ter sido levantada até o momento. Este fato é de extrema importância porque permite perceber que há uma razão para a formação do cluster mas não se consegue analisar o porquê. Isso leva a acreditar-se que existe mais algum dado ou conjunto de dados que influenciam no atributo acelZ, mas que não pôde ser elencado nos dados recolhidos pelo nosso protótipo. Isso porque falta algum dado ali dentro que não temos e que está causando a formação do cluster. Além disso, há a segurança de que estes valores não fazem parte do conjunto de outliers porque os mesmos já foram removidos durante a recriação do subset e por isso o mesmo representa fielmente o conjunto de dados.

6.3.3 Testes realizados durante a CODA2

Para que fosse possível avaliar com mais precisão a necessidade de monitoramento constante e de criação de modelos de classificação, decidiu-se realizar os testes finais durante uma época festiva. Com isso, esperava-se que os valores fossem diferentes dos parâmetros e padrões já encontrados nos diversos outros dias adquiridos no segundo semestre de 2016. Por isso, o período do CODA2 foi particularmente escolhido como sendo o período de Carnaval do 2017. Este é um período que a cidade do Recife recebe diversos turistas e que a população utiliza os transportes públicos com maior intensidade.

Apesar de o protótipo ter estado ligado durante todo o período de carnaval, alguns dias apresentaram dados irregulares. No total, o sábado (24/02/2017), segunda-feira, terça-feira, quarta-feira e quinta-feira (01/03/2017) foram passíveis de análise. Em média 75K instâncias foram obtidas por dia de funcionamento. A seguir, serão mostradas e discutidas as imagens dos

clusters gerados em relação à variável acelZ seguindo o exemplo dos algoritmos mostrados na subseção anterior.

No sábado de carnaval é notório que a vibração do trem foi bastante alta. Esta informação fica perceptível pela grande variação do acelZ. A Figura 6.18 mostra os resultados mencionados. É possível ver a grande densidade dos valores com alguns dados esparsos para maior ou menor gravidade. Parte destes dados podem indicar outliers.

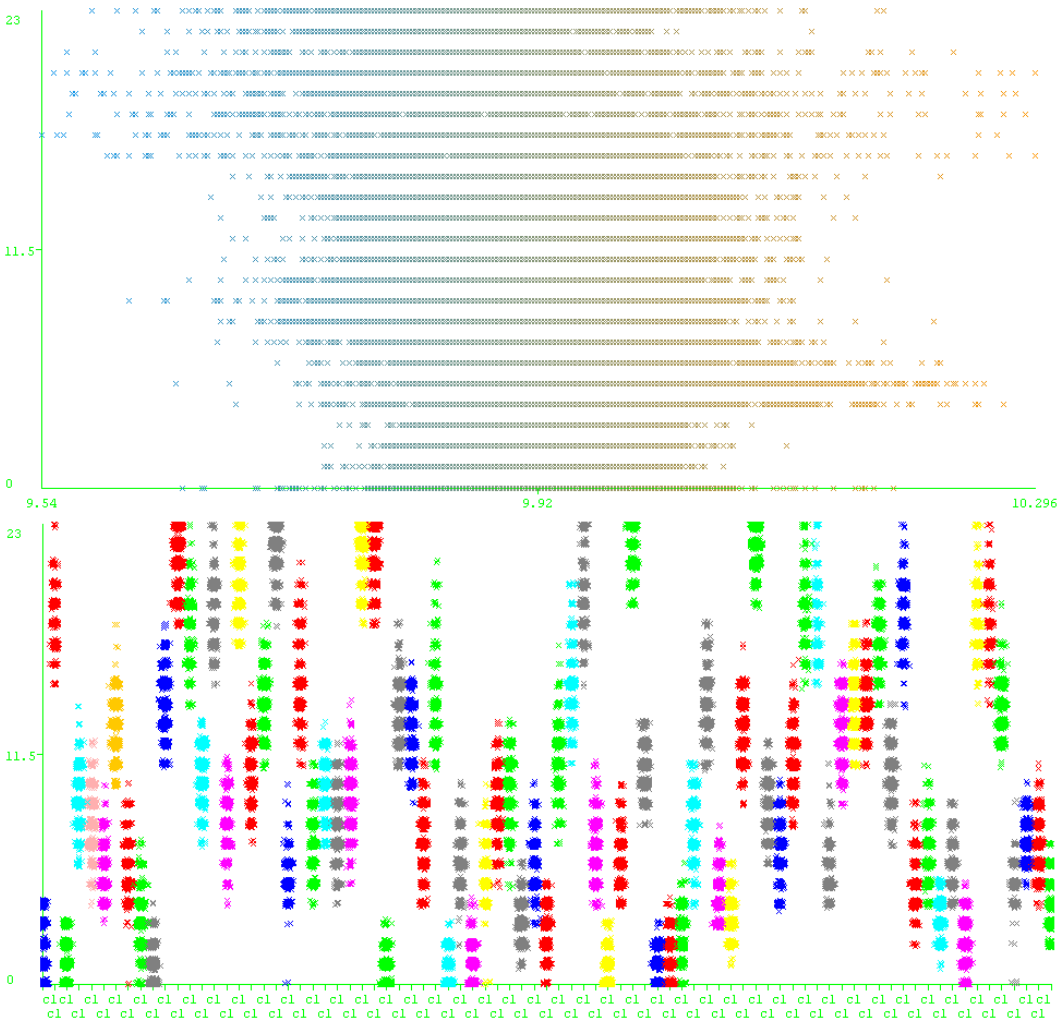


Figura 6.18 Clusters representados em Y pela Hora e em X pelo valor de acelZ durante o funcionamento no sábado de carnaval. Fonte: Dados produzidos pelo autor

A segunda-feira foi marcada por valores bastante curiosos. Percebe-se que a variação no turno da tarde foi muito mais marcante que no resto do dia como na Figura 6.19. Em entrevistas com especialistas os mesmos informaram que isso era normal de acontecer visto que na segunda muitas pessoas voltam para rotinas de trabalho, voltando a utilizar o MetroRec pela tarde para irem para as festas que ocorrem na cidade.

Já os dados da terça, quarta e quinta-feira demonstraram características importantes e que corroboram com as ideias e premissas levantadas anteriormente. Na terça-feira a densidade do gráfico foi muito alta, vide Figura 6.20. É possível perceber que de todos os dias este foi o que o trem sofreu mais vibrações e com maior intensidade. Esta intensidade vai reduzindo aos

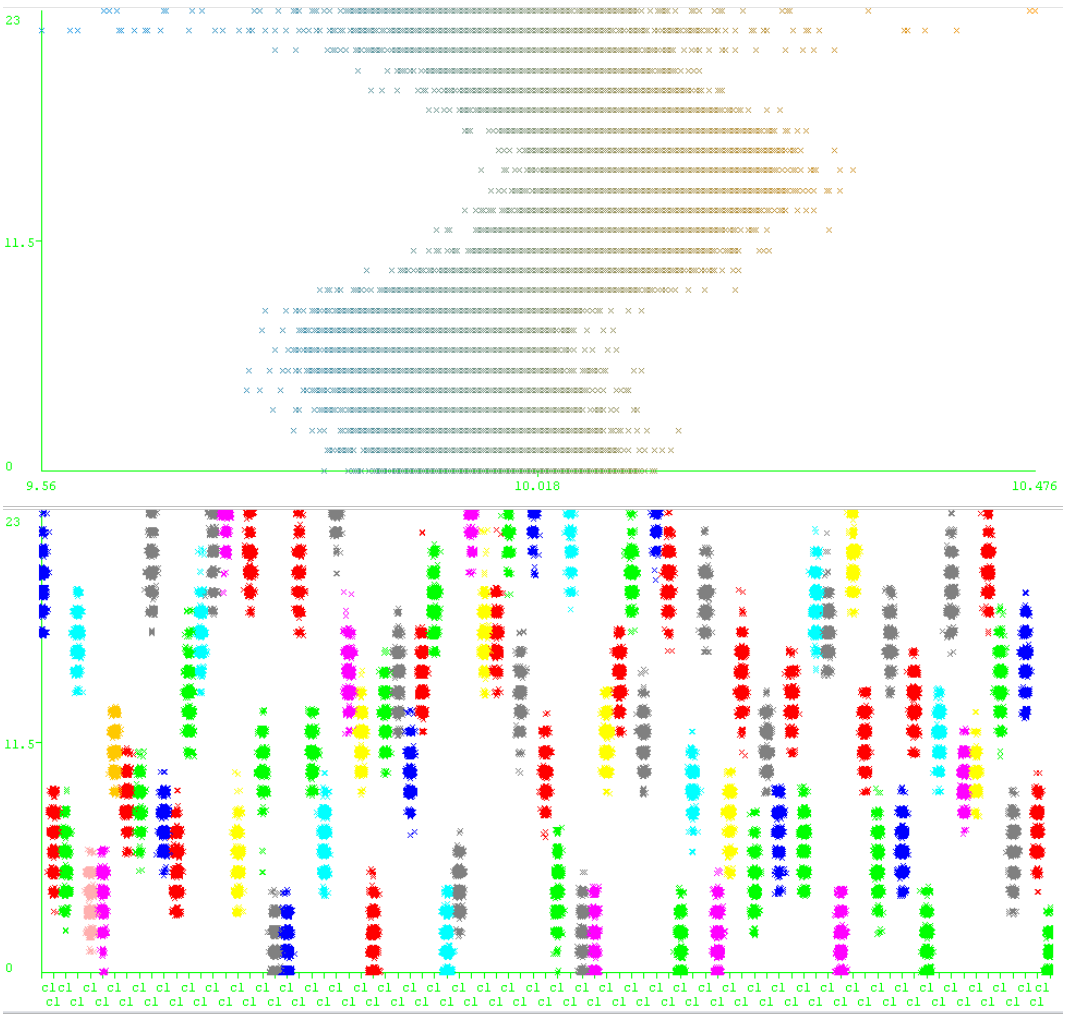


Figura 6.19 Clusters representados em Y pela Hora e em X pelo valor de acelZ durante o funcionamento na segunda de carnaval. Fonte: Dados produzidos pelo autor

poucos à medida que os dias da semana passam, ficando evidenciado nos gráficos da quarta-feira (Figura 6.21) e quinta-feira (Figura 6.22).

A grande quantidade de clusters formados atenta para a necessidade de uma análise de dados e formação de classes. Nos dados coletados em datas de uso normal do metrô os clusters foram muito maiores e bem definidos. Durante o carnaval, como nas imagens anteriores, a quantidade de clusters formados foi elevadíssima. Isto mostra que o trem não segue um padrão diário e que um especialista é necessário para entendimento mais profundo dos padrões encontrados. Com um sistema de monitoramento como o desenvolvido no escopo desta tese, torna-se possível aumentar a acurácia e entendimento de tais padrões ao longo dos anos e das repetidas iterações das análises de dados.

É importante levar em consideração que a medida que o trem leva muitos passageiros e com uma alta carga de vibração, o mesmo pode entrar em degradação, o que pode levar a quebra. Este fato também fica evidenciado pela continuidade da alta vibração do trem 23 após a terça-feira. Talvez isso tenha influenciado a quebra e parada deste trem que ocorreu no dia 02 de março.

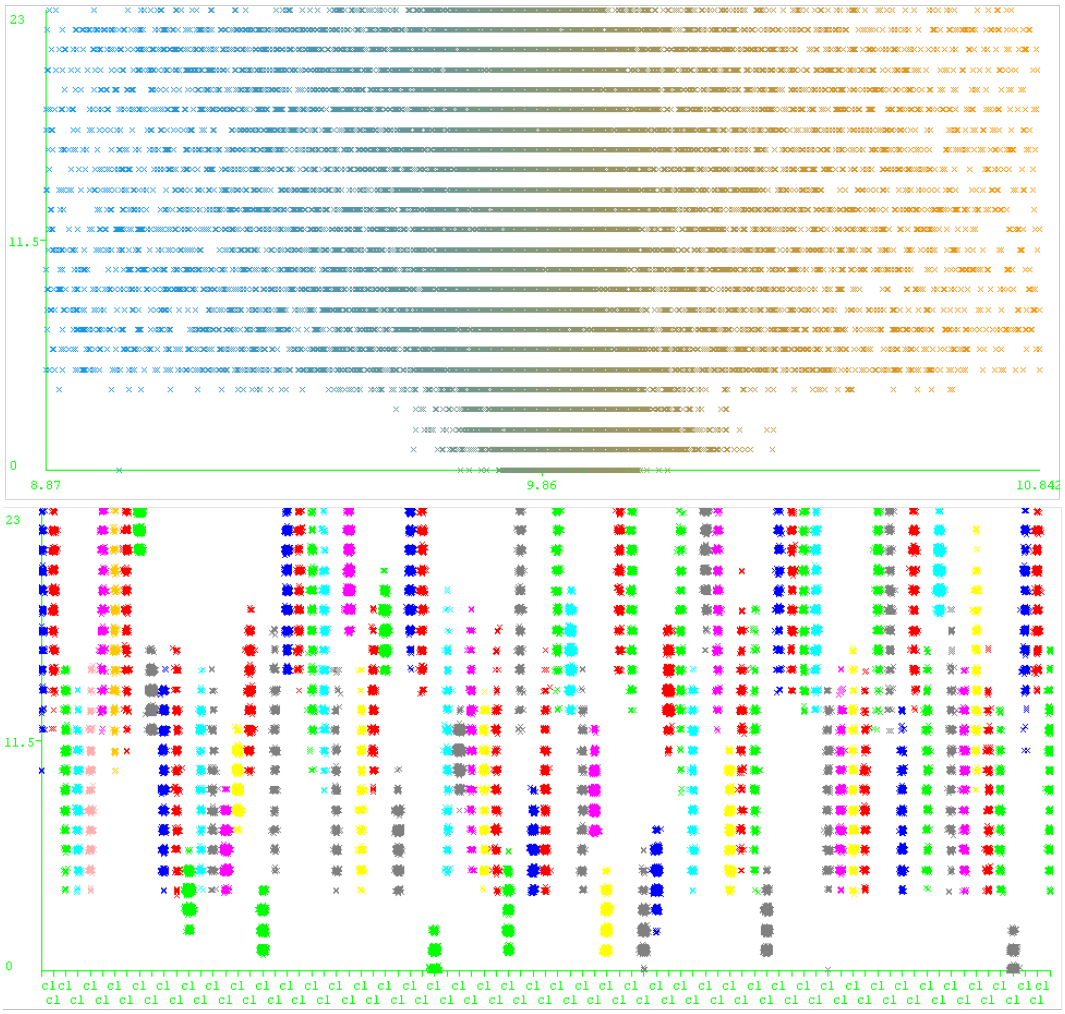


Figura 6.20 Clusters representados em Y pela Hora e em X pelo valor de acelZ durante o funcionamento na terça de carnaval. Fonte: Dados produzidos pelo autor

Também, vale ressaltar aqui que as calibrações necessárias para que os testes sejam realizados com sucesso não puderam ser repetidas durante o carnaval. A grande demanda pelo serviço de transportes e o elevado número de passageiros inviabilizaria a manutenção do protótipo nessa fase de testes. Por isso, o mesmo foi calibrado no primeiro dia e deixado em funcionamento no trem por todo o tempo.

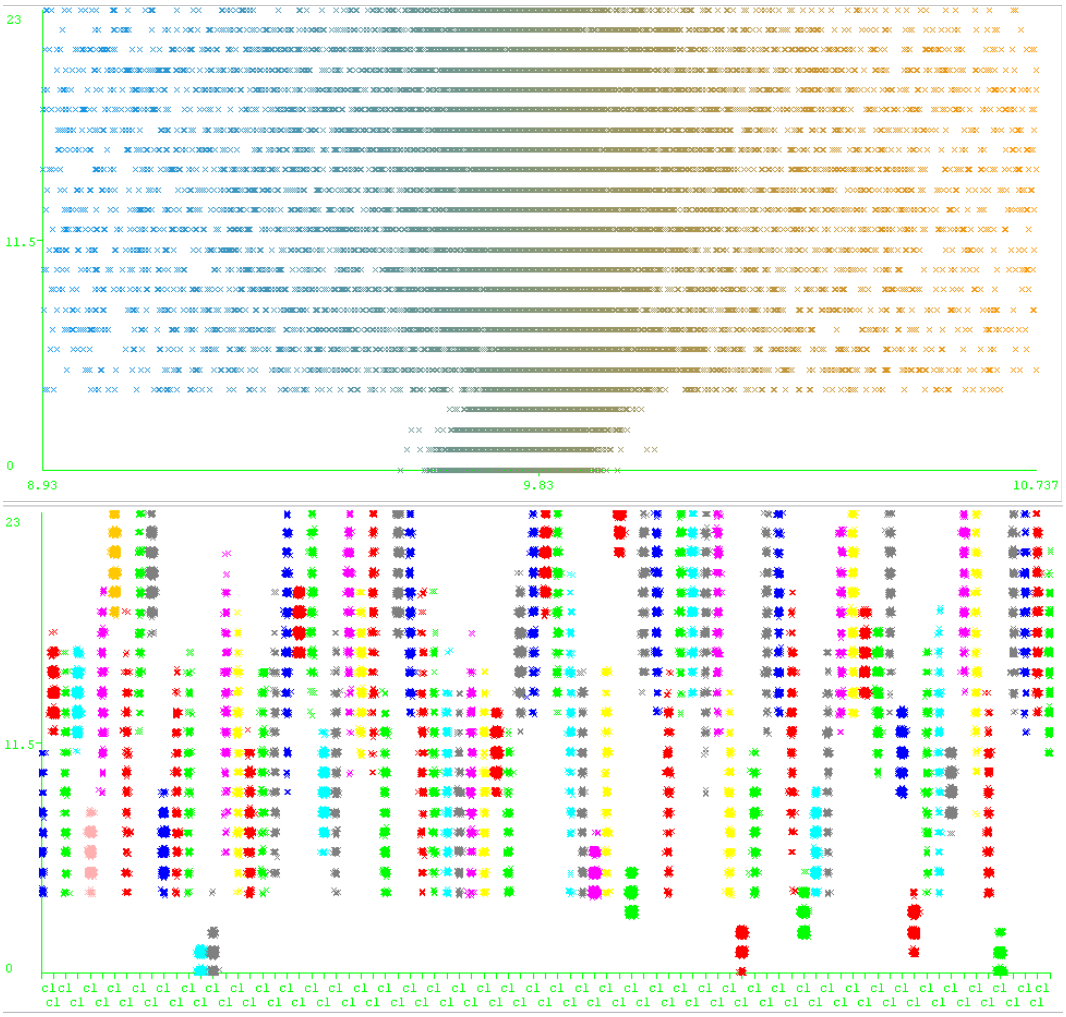


Figura 6.21 Clusters representados em Y pela Hora e em X pelo valor de acelZ durante o funcionamento na quarta de carnaval. Fonte: Dados produzidos pelo autor

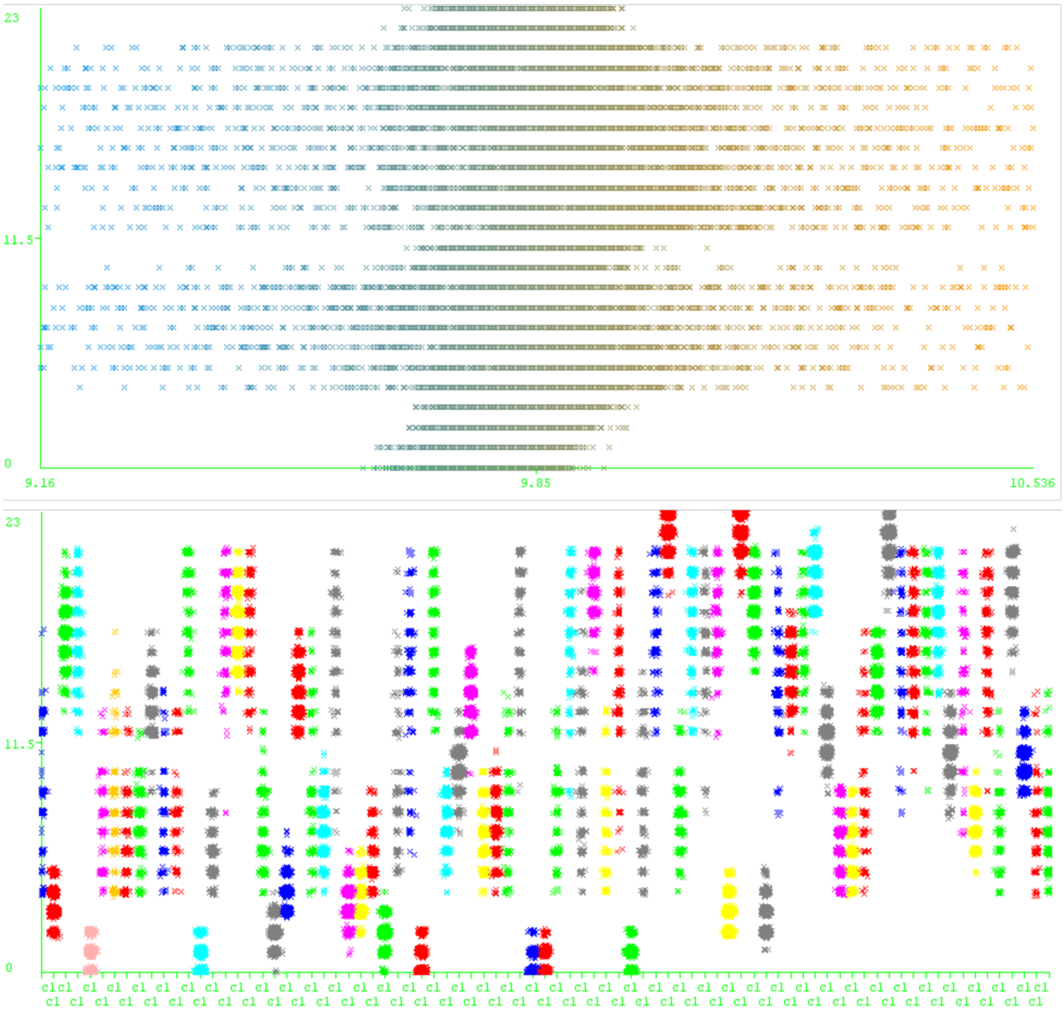


Figura 6.22 Clusters representados em Y pela Hora e em X pelo valor de acelZ durante o funcionamento na quinta de carnaval. Fonte: Dados produzidos pelo autor

CAPÍTULO 7

CONCLUSÕES E TRABALHOS FUTUROS

A pesquisa desenvolvida no escopo desta tese demonstra as potencialidades em utilização de acelerômetros em junção com uma rede de sensores sem fios, de preferência de alta qualidade, para monitoramento de trens. Além disso, a hipótese central parte do pressuposto de que existem diversas informações e relações entre dados que não estão sendo evidenciadas e que podem trazer conhecimento extremamente útil no monitoramento e segurança em trens. Tais suposições ficam evidenciadas e comprovadas ao longo das diversas análises e testes realizados neste trabalho.

Através do estado da arte ficou claro a necessidade de pesquisas na área. Contudo, embora bastante discutido recentemente, o uso de acelerômetros ainda possui uma grande gama de possibilidades a serem abordadas para as mais diversas finalidades. Por isso, os resultados encontrados, ao comprovar nossa hipótese, abordam duas facetas principais: a primeira, demonstrando as inúmeras possibilidades de conhecimento que podem ser extraídas de dados do acelerômetro e giroscópio e a segunda, levantando diversas questões de pesquisas que podem ser alavancadas com tais resultados, aumentando ainda mais os tópicos de pesquisas na área. Vale destacar que os resultados encontrados através do mapeamento sistemático são de grande utilidade não somente para este projeto mas para qualquer um que trabalhe no campo proposto no tema do mapeamento.

A proposta de utilização de acelerômetros e giroscópios com tal finalidade foi divulgada no *17 International Conference on Computational Science and Its Applications, 2017* através do artigo: *Using Accelerometers to Improve Real Time Railway Monitoring Systems Based on WSN* (Anjos et al., 2017). Atualmente, os resultados encontrados alertam os controladores de trens para possíveis problemas que existem e que não estão sendo monitorados de forma direta e constante. Estes resultados se evidenciam em informações como: tempo de abertura de portas, e análises de possíveis atrasos; grandes vibrações no eixo Z do trem, quando o mesmo está com grande lotação; e, grande vibração nos vários sentidos (principalmente eixos X e Y) durantes

as curvas, podendo levar a descarrilamentos. Estes novos resultados estão sendo compilados e melhorados para uma submissão de artigo para revista.

O objetivo mais importante com os trabalhos realizados aqui é atentar e direcionar para a necessidade de criação de um classificador. Contudo, somente o acelerômetro por si só não consegue dar informações suficientes para criação de um modelo de classificação. é preciso que tenhamos outros tipos de dados. Por isso como ainda não sabemos a relação entre os dados, a utilização de algoritmos de clusterização demonstraram ser de extrema importância para criação de padrões e agrupamento dos mesmos em clusters. Com esse entendimento e em posse de outros tipos de dados é possível realizar-se a criação de rótulos para alimentação do classificador. Assim, a avaliação de uma nova instância pode ser realizada de forma automática pelo classificador. Também, é importante destacar que o uso de algoritmos de clusterização requer a participação de um especialista que atua na interpretação do clusters formados o que é custoso para as empresas, destacando ainda mais a necessidade de um modelo de classificação.

Ainda que seja notória a importância das informações levantadas a partir dos algoritmos de inteligência artificial (com ênfase em mineração de dados) utilizados neste projeto, algumas limitações impediram a obtenção de resultados ainda melhores. Uma das principais limitações que podemos citar é a característica do projeto e da pesquisa. O fato de trabalharmos com tecnologias que envolvem diversos dispositivos e veículos de difícil manuseio fez com que a etapa de testes e validação do projeto dependesse de algumas variáveis, como: disponibilidade de trens livres para instalação, cordialidade de maquinistas para testes e instalações locais, fiação, computadores e demais equipamentos para rede de sensores sem fios, liberação da empresa para realização dos experimentos, dentre outros.

Outra limitação que foi crucial no projeto foi a liberação dos resultados das variáveis já existentes no trem, tais como: informações das bolsas de ar, velocidade do trem, abertura e fechamento das portas, corrente e voltagem de operação do trem, etc. Se fosse possível a utilização destes dados o modelo criado pelos algoritmos de mineração seria muito mais preciso. Tal melhora se daria devido ao fato de que, com a realização de correlações com esses valores, seria possível criar um modelo que serve de suporte para que no futuro as informações pudessem ser deduzidas apenas com os dados dos acelerômetros e giroscópios. Utilizando a metodologia aplicada neste projeto e devido a importância de criação de um modelo desse tipo ser tão grande, não desistimos de tentá-lo. Por isso, como trabalhos futuros tentaremos novas parcerias para que essa correlação possa ser realizada.

Além disso, este trabalho direciona diversos tópicos para trabalhos futuros, sejam de desenvolvimento ou de pesquisa. Dentre eles podemos citar: uso dos acelerômetros para detectar irregularidades nos trilhos, uso de redes neurais para descobrir padrões e perfis de maquinistas dos trens, modelos de gerência de horários para otimização do uso, modelos de análise de eficiência energética, sistemas de interação com usuário através de totens ou sistemas móveis, etc.

Um fato curioso foi perceber que os resultados encontrados também demonstraram ser expansíveis para outros campos de pesquisa. Atualmente, o uso de acelerômetros em ônibus pela nossa equipe de pesquisa tem elencado bons resultados e expandido o escopo de aplicação das hipóteses iniciais deste trabalho. Atualmente, um sistema está sendo desenvolvido para testes com essa finalidade.

Referências Bibliográficas

Aboelela, E., Edberg, W., Papakonstantinou, C., and Vokkarane, V. (2006). Wireless sensor network based model for secure railway operations. In *IPCCC*. IEEE.

Ali, S. and Smith, K. A. (2006). On learning algorithm selection for classification. *Applied Soft Computing*, 6(2):119 – 138.

André, P. and Varum, H. (2013). *Accelerometers Principles, Structure and Applications*. Nova Publisher.

Anjos, E. G., dos Santos, S. G., de Araújo, I. R. S., Araújo, R. C. C., and Belo, F. A. (2017). Using accelerometers to improve real time railway monitoring systems based on wsn. In *Computational Science and Its Applications – ICCSA 2017*, pages 761–769, Cham. Springer International Publishing.

Araujo, R. C. C. (2009). *Sistema Telemétrico Dinâmico Móvel Aplicado ao Trem Unidade Elétrica do Metrô do Recife*. PhD thesis, Programa de Pós-graduação em Engenharia Mecânica da Universidade Federal da Paraíba, Joao Pessoa, PB, Brazil.

Araújo, A. L. V. d. (2009). Aplicação de regras de associação para auxílio na gestão de vendas de uma empresa varejista utilizando a ferramenta weka. Monografia (Bacharel em Engenharia de Computação), Escola Politécnica de Pernambuco, Brazil.

Araújo, R., Belo, F. A., Santos, J. L. S., Lima, J., Holanda, S., and Lima-Filho, A. (2010). Utilização do sistema railbee para estimativa em tempo real do número de passageiros e análise do desempenho do tue. *Revista Ferroviária*, 3:60–63.

Ataei, S., Mohammadzadeh, S., and Miri, A. (2016). Dynamic forces at square and inclined rail joints: Field experiments. *Journal of Transportation Engineering*, 142(9).

Atanasovski, V. and Gavrilovska, L. (2011). *Application and Multidisciplinary Aspects of Wireless Sensor Networks: Concepts, Integration, and Case Studies*, chapter Vehicular Sensor Networks: General Aspects and Implementation Issues, pages 213–241. Springer London, London.

Bassetti, M., F. Braghin, G. C., and Castelli-Dezza, F. (2013). *Triaxial Multi-range MEMS Accelerometer Nodes for Railways Applications*, chapter Special Topics in Structural Dynamics, pages 463–484. Springer New York, New York.

Berkhin, P. (2006). *A Survey of Clustering Data Mining Techniques*, pages 25–71. Springer Berlin Heidelberg, Berlin, Heidelberg.

Bertran, E. and Delgado-Penin, J. A. (2004). On the use of gps receivers in railway environments. *IEEE Transactions on Vehicular Technology*, 53(5):1452–1460.

Bin, Z. and Wensheng, X. (2015). An improved algorithm for high speed train's maintenance data mining based on mapreduce. In *2015 International Conference on Cloud Computing and Big Data (CCBD)*, pages 59–66.

Browne, K., Chen, C.-C., and Volakis, J. L. (2007). A novel radiator for a 2.4 ghz wireless unit to monitor rail stress and strain from a train mounted receiver. In *2007 IEEE Antennas and Propagation Society International Symposium*, pages 2618–2621.

BS IEC 60747-14-4 (2011). Iec standard. Standard, British Standards Institution.

Budgen, D., Turner, M., Brereton, P., and Kitchenham, B. (2008). Using Mapping Studies in Software Engineering. In *Proceedings of PPIG 2008*, pages 195–204. Lancaster University.

Buss, D. (2011). Utilização de técnicas de inteligência de negócios para descoberta em bases de dados acadêmicas. Monografia (Bacharel em Ciência da Computação), UFPEL (Universidade Federal de Pelotas), Brazil.

Ceapa, I., Smith, C., and Capra, L. (2012). Avoiding the crowds: Understanding tube station congestion patterns from trip data. In *Proceedings of the ACM SIGKDD International Workshop on Urban Computing*, UrbComp '12, pages 134–141, New York, NY, USA. ACM.

Cornelius Junior, R. (2015). Uso da mineração de dados na identificação de alunos com perfil de evasão do ensino superior. Monografia (Bacharel em Ciência da Computação), Universidade de Santa Cruz do Sul, Brazil.

Crnjin, A. (2011). *Introduction: Bird's-Eye View of Wireless Sensor Networks*, pages 1–9. Springer London, London.

Culler, D. E. and Hong, W. (2004). Wireless sensor networks - introduction. *Commun. ACM*, 47(6):30–33.

Cuomo, F., Luna, S. D., Todorova, P., and Suihko, T. (2008). Topology formation in ieee 802.15.4: Cluster-tree characterization. In *Pervasive Computing and Communications, 2008. PerCom 2008. Sixth Annual IEEE International Conference on*, pages 276–281.

de Araújo, I. R. S. (2016). Detecção de padrões no monitoramento de trens urbanos através de técnicas de mineração de dados. Monografia (Bacharel em Ciência da Computação), CI (Centro de Informática), UFPB (Universidade Federal da Paraíba), João Pessoa, Brazil.

De Vasconscelos, Lívia Maria Rocha; De Carvalho, C. L. (2004). Aplicação de regras de associação para mineração de dados na web. Technical report, Instituto de Informática da Universidade Federal de Goiás.

Delprete, C. and Rosso, C. (2009). An easy instrument and a methodology for the monitoring and the diagnosis of a rail. *Mechanical Systems and Signal Processing*, 23:940–956.

dos Santos, J. L. A., de Araújo, R. C. C., Filho, A. C. L., Belo, F. A., and de Lima, J. A. G. (2011). Telemetric system for monitoring and automation of railroad networks. *Transportation Planning and Technology*, 34(6):593–603.

Eriksson, J., Girod, L., Hull, B., Newton, R., Madden, S., and Balakrishnan, H. (2008). The pothole patrol: Using a mobile sensor network for road surface monitoring. In *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services*, MobiSys '08, pages 29–39, New York, NY, USA. ACM.

Evandro Costa, Ryan Baker, L. A., Magalhães, J., and Marinho, T. (2013). *Mineração de Dados Educacionais: Conceitos, Técnicas, Ferramentas e Aplicações*, pages 1–29. Comissão Especial de Informática na Educação.

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996a). Knowledge discovery and data mining: Towards a unifying framework. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, pages 82–88. AAAI Press.

Fayyad, U. M., Piatetsky-Shapiro, G., and Smyth, P. (1996b). Advances in knowledge discovery and data mining. chapter From Data Mining to Knowledge Discovery: An Overview, pages 1–34. American Association for Artificial Intelligence, Menlo Park, CA, USA.

Garcia, A. C. (2012). Mineração de dados aplicada a sistemas de recomendação. Monografia (Bacharel em Ciência da Computação), Universidade Santa Cruz do Sul, Brazil.

Geethapriya, S. and Jawahar, A. (2013). Performance evaluation of hybrid topology control in wsn. In *Communications and Signal Processing (ICCSP), 2013 International Conference on*, pages 9–13.

Gil-Castineira, F., Gonzalez-Castano, F. J., Duro, R. J., and Lopez-Pena, F. (2008). Urban pollution monitoring through opportunistic mobile sensor networks based on public transport. In *Computational Intelligence for Measurement Systems and Applications, 2008. CIMSA 2008. 2008 IEEE International Conference on*, pages 70–74. IEEE.

Goodloe, A. and Pike, L. (2010). Monitoring distributed real-time systems: A survey and future directions. Technical Report NASA/CR-2010-216724, NASA Langley Research Center. Available at `http://ntrs.nasa.gov/search.jsp?R=278742&id=3&as=false&or=false&qs=Ns%3DArchiveName%257c0%26N%3D4294643047`.

Heirich, O., Lehner, A., Robertson, P., and Strang, T. (2011). Measurement and analysis of train motion and railway track characteristics with inertial sensors. In *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1995–2000.

Hu, Z. X., Gallacher, B. J., Burdess, J. S., Fell, C. P., and Townsend, K. (2011). Precision mode matching of mems gyroscope by feedback control. In *2011 IEEE SENSORS Proceedings*, pages 16–19.

IEEE 1293 (2008). Specification format guide and test procedure for linear single-axis, nongyroscopic accelerometers. Standard, Institute of Electrical and Electronics Engineers (IEEE).

Jain, A. K. (2010). Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651 – 666. Award winning papers from the 19th International Conference on Pattern Recognition (ICPR).

Kitchenham, B. and Brereton, P. (2013). A systematic review of systematic review process research in software engineering. *Information and Software Technology*, 55(12):2049 – 2075.

Kitchenham, B. and Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering. Technical Report EBSE 2007-001, Keele University and Durham University Joint Report.

Kitchenham, B. A. (2012). Systematic review in software engineering: Where we are and where we should be going. In *Proceedings of the 2Nd International Workshop on Evidential Assessment of Software Technologies*, EAST '12, pages 1–2, New York, NY, USA. ACM.

Kohavi, R. (1995). The power of decision tables. In Lavrac, N. and Wrobel, S., editors, *Machine Learning: ECML-95*, pages 174–189, Berlin, Heidelberg. Springer Berlin Heidelberg.

Kopetz, H. (1997). *Real-Time Systems: Design Principles for Distributed Embedded Applications*. Kluwer Academic Publishers, Norwell, MA, USA, 1st edition.

Kouroussis, G., Kinet, D., Moeyaert, V., Dupuy, J., and Caucheteur, C. (2016). Development of structural railway monitoring solutions using fbg sensors. *Proceedings of the 23rd International Congress on Sound and Vibration*.

Lai, Y.-J., Kuo, W.-H., Chiu, W.-T., and Wei, H.-Y. (2012). Accelerometer-assisted 802.11 rate adaptation on mobile wifi access. *EURASIP Journal on Wireless Communications and Networking*, 2012(1):1–18.

Lewis, F. L. (2005). *Wireless Sensor Networks*, pages 11–46. John Wiley and Sons, Inc.

Librelotto, Solange Rubert; Mozzaquatro, P. M. (2014). Análise dos Algoritmos de Mineração J48 e Apriori Aplicados na Detecção de Indicadores da Qualidade de Vida e Saúde. *Revista Interdisciplinar de Ensino, Pesquisa e Extensão*, 1.

Lidén, T. (2015). Railway infrastructure maintenance - a survey of planning problems and conducted research. *Transportation Research Procedia*, 10:574 – 583. 18th Euro Working Group on Transportation, {EWGT} 2015, 14-16 July 2015, Delft, The Netherlands.

Lipan, F. and Groza, A. (2010). Mining traffic patterns from public transportation gps data. In *Proceedings of the 2010 IEEE 6th International Conference on Intelligent Computer Communication and Processing*, pages 123–126.

MIL-A-27261 (2011). Accelerometer aircraft. Standard, Military and Government Specs and Standars - Naval Publuications and Form Center - NPFC.

Mirabadi, A., Mort, N., and Schmid, F. (1999). Design of fault tolerant train navigation systems. In *American Control Conference, 1999. Proceedings of the 1999*, volume 1, pages 104–108 vol.1.

Nakamura, E. F., Loureiro, A. A. F., and Frery, A. C. (2007). Information fusion for wireless sensor networks: Methods, models, and classifications. *ACM Comput. Surv.*, 39(3).

Nejikovsky, B. and Keller, E. (2000). Wireless communications based system to monitor performance of rail vehicles. In *Railroad Conference, 2000. Proceedings of the 2000 ASME/IEEE Joint*, pages 111–124, Newark, NJ, USA.

NK Das, CK Das, R. M. and Bhowmik, J. C. (2009). Satellite based train monitoring system. In *Journal of Electrical Engineering*, volume 36, pages 35 – 38. J. Elec. Engg., Instn. Engrs., Bangladesh.

Pachiarotti, J. F. B. (2012). Aplicação de técnicas de mineração de dados no aprimoramento do atendimento médico em um cenário de plano de saúde. Monografia (Bacharel em Ciência da Computação), Universidade de Vila Velha, Brazil.

Passaro, V. M. N., Cuccovillo, A., Vaiani, L., De Carlo, M., and Campanella, C. E. (2017). Gyroscope technology and applications: A review in the industrial perspective. *Sensors*, 17(10).

Penumuchu, C. V. (2007). *Simple Real-time Operating System: A Kernel Inside View for a Beginner*. Trafford Publishing.

Peters, A., Chung, K. Y., and Chu, S. (2001). High-precision gravity measurements using atom interferometry. *Metrologia*, 38(1):25.

Petersen, K., Feldt, R., Mujtaba, S., and Mattsson, M. (2008). Systematic mapping studies in software engineering. In *Proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering*, EASE'08, pages 68–77, Swinton, UK, UK. British Computer Society.

q. Ma, Z., f. Gao, Z., and Yan, Y. (2006). Wireless monitoring system of train speed based on rcm3000. In *2006 International Conference on Machine Learning and Cybernetics*, pages 661–664.

Rezende, S. (2003). *Sistemas inteligentes: fundamentos e aplicações*. Manole.

Shah, R. C., Wan, C.-y., Lu, H., and Nachman, L. (2014). Classifying the mode of transportation on mobile phones using gis information. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '14, pages 225–229, New York, NY, USA. ACM.

Sharma, V. K. and Vaidya, Y. M. (2007). Radio frequency identification based rail wagon monitoring system. In *2007 IEEE International Conference on Mechatronics*, pages 1–6.

Stockx, T., Hecht, B., and Schöning, J. (2014). Subwayps: Towards smartphone positioning in underground public transportation systems. In *Proceedings of the 22Nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, SIGSPATIAL '14, pages 93–102, New York, NY, USA. ACM.

Tariq Abuhamdia, Saied Taheri, A. M. and Davis, D. (2014). Rail defect detection using data from tri-axial accelerometers. In *ASME/IEEE Joint Rail Conference*, pages 93–102, New York, NY, USA. THE TRANSPORTATION DIVISION, ASME.

Waharte, S., Boutaba, R., Iraqi, Y., and Ishibashi, B. (2006). Routing protocols in wireless mesh networks: challenges and design considerations. *Multimedia Tools and Applications*, 29(3):285–303.

Weston, P. F., Roberts, C., Goodman, C. J., and Ling, C. S. (2006). Condition monitoring of railway track using in-service trains. In *Railway Condition Monitoring, 2006. The Institution of Engineering and Technology International Conference on*, pages 26–31.

Witten, I., Frank, E., Hall, M., and Pal, C. (2016). *Data Mining: Practical Machine Learning Tools and Techniques*. The Morgan Kaufmann Series in Data Management Systems. Elsevier Science.

Yang, S.-H. (2013). *Wireless Sensor Networks: Principles, Design and Applications*. Springer Publishing Company, Incorporated.