UNIVERSIDADE ESTADUAL DO CEARÁ

CENTRO DE CIÊNCIAS E TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

MESTRADO ACADÊMICO EM CIÊNCIA DA COMPUTAÇÃO

CARLOS HENRIQUE ORIÁ OLIVEIRA QUEVEDO

SYDVELM - PROTEGENDO VANETS CONTRA ATAQUES SYBIL USANDO EXTREME LEARNING MACHINE

FORTALEZA – CEARÁ

2019

CARLOS HENRIQUE ORIÁ OLIVEIRA QUEVEDO

SYDVELM - PROTEGENDO VANETS CONTRA ATAQUES SYBIL USANDO EXTREME LEARNING MACHINE

Dissertação apresentada ao Curso de Mestrado Acadêmico em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências e Tecnologia da Universidade Estadual do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Ciência da Computação

Orientador: Prof. Emérito Dr. Joaquim Celestino Júnior

Co-Orientador: Prof. Dr. Gustavo Augusto Lima de Campos

FORTALEZA – CEARÁ

2019

À minha Esposa, meu amor, você significa segurança e certeza de que não estou sozinho nessa caminhada. À minha filha e filho, os frutos de nosso amor, as jóias mais preciosas do meu existir. Mãe e Irmã, seus cuidados me deram a esperança para seguir em frente. Pai, (in memorium), sua honestidade sempre foi exemplo. À minha família, por acreditar em mim.

AGRADECIMENTOS

Agradeço a Deus pelos dons que me deu e que tornaram possível finalizar mais essa tão importante etapa da minha vida e por tudo que conquistei até o momento.

À minha família, que sempre me apoiou e esteve ao meu lado nos momentos alegres, tristes, nervosos, ansiosos e esperançosos, e também pela paciência durante essa minha trajetória.

Ao meu orientador Prof. Emérito Dr. Joaquim Celestino Junior e ao meu Co-orientador Prof. Dr. Gustavo Augusto Lima de Campos, a estes sou bastante grato pela orientação, paciência e pelos incentivos dados ao meu trabalho.

A todos os professores, especialmente ao Prof. Dr. Thelmo de Araújo, pois graças as suas dicas valiosas de Mineração de Dados, que me auxiliaram no preparo dos dados dos experimentos, assim como também a todos os funcionários do MACC (D. Cláudia, Allison, Ana Maria, D. Neuma) pelo conhecimento e experiência que dividiram comigo e que me guiaram durante o meu desenvolvimento no mestrado.

Aos meus amigos Renata Rosa Russo Costa Ribeiro e Antônio Carlos Costa Ribeiro, pelos momentos de descontração, dicas e conselhos valiosos.

Aos colegas de sala e de laboratório do MACC (Diego Allyson, Bruno, Kilvia, Matheus, Jefferson, Jéssica, Levy, Rodrigo, Humberto, dentre outros), que nos momentos de descontração, apoio e coleguismo me ajudaram e que de alguma forma contribuíram com meu curso e com este trabalho.

À FUNCAP pelo auxílio financeiro prestado.

Enfim, a todos os que de alguma forma contribuíram para a realização deste trabalho.

"Eu tentei 99 vezes e falhei, mas na centésima tentativa eu consegui, nunca desista de seus objetivos mesmo que esses pareçam impossíveis, a próxima tentativa pode ser a vitoriosa."

(EINSTEIN, Albert, 1955)

RESUMO

Segurança e Privacidade sempre foram uma das principais preocupações em quaisquer tipos de Redes de Computadores. Desde que foram desenvolvidas, as Redes veiculares as mesmas apresentam em sua arquitetura, nós móveis velozes, auto-organizáveis, rede distribuída e uma topologia com frequentes mudanças. No meio acadêmico e industrial, os aspectos de segurança e integridade dos dados, bem como privacidade das informações dos usuário finais, são um dos maiores interesses em sua crescente área de pesquisa e desenvolvimento. Qualquer tipo de ameaça ou ataque de qualquer tipo em algum dos seus componentes, pode paralisar a comunicação inteira da rede veicular caso ela não tenha mecanismos de segurança incorporados em sua arquitetura. A associação dos conhecidos mecanismos de segurança de redes de computadores aplicados à redes veiculares, como a Inteligência Artificial, visa melhorar os serviços de segurança e intercâmbio de mensagens habilitando uma melhor avaliação das ameaças e ataques trazendo um aumento do discernimento e reconhecimento dessas questões, uma habilidade que os Sistemas de Machine Learning adicionam ao tratamento desses problemas que impactam diretamente na busca por um Sistema de Transporte Inteligente. Este projeto tem como objetivo a pesquisa e o estudo da aplicação de técnicas de Extreme Machine Learning, especialização de Machine Learning, para identificação de ataques Sybil em cenários VANETs, bem como sua avaliação de desempenho.

Palavras-chave: VANETs, ITS, Redes, Segurança, V2V, V2I, Wireless, Autenticação, Criptografia, Machine Learning, Extreme Machine Learning, Vulnerabilidade, Ameaças, Privacidade, Ataques, Integridade, Sybil.

ABSTRACT

Security and Privacy have always been major concerns in any kind of Computer Network. Since they were first developed, the Vehicular Networks (Vehicular Ad hoc Networks – VANETs), a field of specialization of MANETs (Mobile Ad Hoc Networks), they have been strategic ways to achieve the Intelligent Transportation System (ITS), which comprehend the integration and communication between vehicles, sensors and fixed road-side components (routers, gateways and services). The Vehicular Networks present in their architecture fast mobile nodes, self-organized, distributed network and frequently changing topology. In the Academic Community and Industrial Environment, Security Aspects and Data Integrity, also end-user's privacy informations are one of the greatest interests in their growing area of research and development. Any type of attack or threat of any kind in any of their components can easily paralyse the entire communication network in case the same does not have embedded Security Mechanisms in its architecture. The association of the well-known Security mechanisms of the Computer Networks applied to Vehicular Networks, with the support of Artificial Intelligence techniques, aims to improve Security Services and the interchange of Messages to enable a better evaluation of threats and attacks bringing increased power of judgement and acknowledgement of such issues, an ability that Artificial Intelligent Systems add to the treatment of such problems, which impacts directly in the search for Intelligent Transportation Systems. This project has a primal objective: the research and study of the application of Extreme Machine Learning, an specialization of Machine Learning, for Sybil Attacks Identification in VANETs scenarios, as well as its performance evaluation.

Keywords: VANETs, ITS, Networks, Security, V2V, V2I, Wireless, Authentication, Criptography, Machine Learning, Extreme Machine Learning, Vulnerability, Threats, Privacy, Atacks, Integrity, Sybil.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Com a evolução da tecnologia, as Redes Veiculares estão se tornando cada vez mais populares. Os avanços nos requisitos de navegação segura, os investimentos dos fabricantes de veículos e das Autoridades de Transporte público trazem para as VANETs grande potencial em termos de aplicações diversas associadas à Segurança no Trânsito, eficiência do tráfego e entretenimento.

De acordo com (GU *et al.*, 2016), para a maioria das aplicações veiculares, dirigir com segurança é de extrema importância, por isso se torna necessário implantar mecanismos apropriados que forneçam isso. As Redes Veiculares são vulneráveis a vários tipos de ataques em comparação às redes convencionais, devido sua natureza dinâmica e independência de infraestrutura.

As ameaças e vulnerabilidades nas Redes Veiculares são inúmeras. Elas podem ser alvo desde ataques de negação de serviços (DoS), até mesmo os veículos sofrerem hackeamento ou perda do controle de carros autônomos para os fins mais obscuros possíveis.

Dentre esses perigos, um dos mais severos é o Ataque Sybil, onde um ou mais nós (veículos) maliciosos forjam um grande número de identidades falsas, com o intuito de interromper o correto funcionamento da rede ou das aplicações e comunicações da VANET.

Evitar os ataques Sybil nas VANETs, se torna uma difícil e árdua tarefa, já que a maioria das técnicas usuais implementadas e padronizadas, permitem, ou uma grande margem de erro, ou um uso excessivo e demorado de recursos computacionais, o que pode vir a ser inviável nesses casos.

Douceur (2002) foi o primeiro a descrever e definir o Ataque Sybil. Essa ameaça em Segurança de Computadores é um ataque onde o Sistema de Reputação é subvertido através da falsificação de identidades em Redes peer-to-peer. Foi assim nomeado posteriormente ao Sujeito do Livro Sybil, um estudo de caso de uma mulher diagnosticada com problema de Transtorno de Identidade Dissociativa.

Esse ataque consiste no envio de múltiplas mensagens a partir de um nó da rede com múltiplas identidades, enquanto para um nó normal é permitido somente uma identidade. O crescimento de ataques Sybil nas VANETs demonstra a importância de sua detecção. Esse desafio ocorre já que nas redes veiculares não há uma autoridade centralizada logicamente, assim os Ataques Sybil são sempre possíveis de ocorrer (podem permanecer não-detectados); exceto sob a extrema e não-realística hipótese de uma paridade de recursos e coordenação entre as

entidades da rede envolvidas.

Um agressor Sybil pode criar uma ilusão e isso tem o potencial de injetar informação falsa nas redes através de inúmeras identidades virtuais fabricadas. Ele pode até lançar posteriormente ataques DoS prejudicando as operações normais de protocolos de difusão de dados. Por exemplo, na aplicação de sistemas de alerta de desaceleração, se um veículo reduz sua velocidade significativamente, o mesmo irá disseminar um alerta para os veículos seguintes. Os Destinatários dessas mensagens de alerta irão retransmitir as mesmas para os veículos mais atrás. Entretanto, esse processo de encaminhamento de mensagens pode sofrer interferência por um grande número de veículos maliciosos Sybil. Dessa forma, o adversário malicioso pode criar um empilhamento massivo numa auto-estrada, potencialmente gerando uma grande perda de vidas (KAFIL; FATHY; LIGHVAN, 2012).

Mecanismos de Segurança nas Redes Veiculares têm sido amplamente pesquisados nos últimos anos com o objetivo de se chegar em um fornecimento de um Sistema de Transporte Inteligente (ITS) seguro e confiável. Para garantir Segurança e Proteção nas VANETs, vários dos mecanismos propostos nos últimos anos foram padronizados e são recomendados pelo IEEE e ETSI. Por regra, os serviços de segurança propostos se baseiam em três mecanismos principais:

- Algoritmos de Encriptação;
- Infraestrutura de Chave Pública (Public Key Infras-tructure - PKI);
- Pseudônimos.

Esses serviços protegem nos ambientes da comunicação veicular:

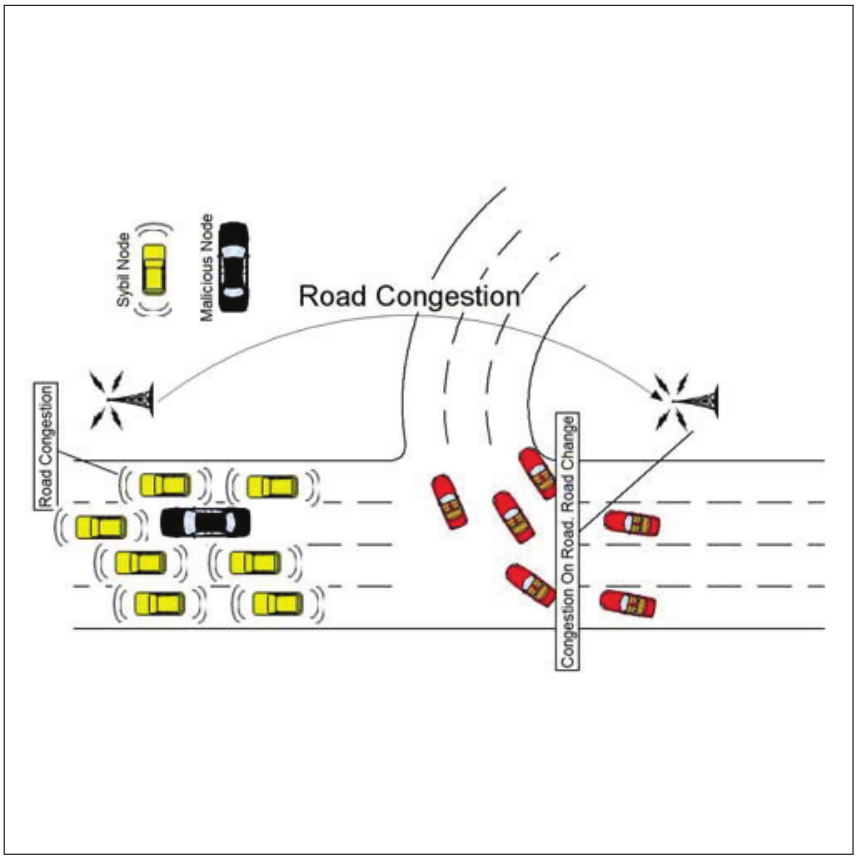
- A privacidade das estações ITS;
- A autenticidade das mensagens;
- A integridade das mensagens.

Todavia, as Redes Veiculares ainda assim são vulneráveis à ataques. As VANETs herdaram todas as falhas de segurança associadas aos outros tipos de redes móveis, e são assim sujeitas às muitas ameaças a segurança e privacidade. Qualquer comportamento malicioso de usuários tal como alteração do conteúdo das mensagens ou ataques replay das mesmas, pode vir a ser fatal para os outros usuários. Essas questões na Segurança em VANET se tornam mais desafiadoras devido às características únicas da rede veicular, tais como a alta mobilidade dos nós e a larga escala da rede.

Além disso, também é preciso fornecer proteção à privacidade, onde as informações relativas aos usuários tais como os seus nomes, a placa dos carros, sua velocidade, seu posicionamento, rotas e relacionamentos precisam ser protegidas. Em aspectos de Segurança, as

autoridades devem ser capazes de obter as identidades dos emissores de mensagens envolvidos em casos de investigação de cenas de crimes/acidentes de carro, e que também podem ser usadas para encontrar testemunhas desses eventos. É de extrema importância o desenvolvimento de um pacote de mecanismos de segurança bem elaborados e cuidadosamente projetados para a obtenção e da preservação de privacidade numa VANET.

Figura 1 – Cenário de Ataque Sybil nas redes veiculares



Fonte – (RABIEH *et al.*, 2015)

Nesse trabalho, será desenvolvido um método para classificar e identificar o ataque Sybil baseado no Aprendizado de Máquina (Machine Learning) Supervisionado, usando de características ocasionais dos nós veiculares, auxiliando na proteção das Redes Veiculares. O método utilizará o mecanismo básico de Extreme Learning Machine (ELM), uma especialização de Redes Neurais Artificiais (RNA), do grupo de algoritmos de Machine Learning chamados de biológicos.

1.1 MOTIVAÇÃO

Nas Redes Veiculares assim como nas Manets, não há autoridade central para monitorar. A natureza altamente dinâmica e aberta da rede, não permite fixar rotas seguras pré-computadas para o envio das mensagens de um nó para outro. Esse tipo de rede não pode ser vinculada por qualquer política de segurança definida pelo dono da informação.

Questões de Segurança têm sido muito estudadas nas pesquisas em VANETs (QU *et al.*, 2015). Nas VANETs tradicionais, uma Infra-estrutura de Chave Pública é comumente adotada pelo padrão IEEE 1609.2 (TIWARI, 2015). Nelas uma Lista de Revogação de Certificado (CRL) é emitida pela Autoridade de Certificação (CA) periodicamente. Não há um mecanismo padrão proposto para CRL. A Chave Pública pode apenas assegurar requisitos fundamentais de Segurança em VANETs, que são a Autenticação e Integridade de messagem.

Os mecanismos criptográficos, para confidencialidade da comunicação e autenticação dos nós, não ajudam contra ataques de abandono ou atraso de pacotes ou ataque da pressa (Rush Attack). O compartilhamento da informação deve ser através dos nós autorizados nas rotas confiáveis, para que a entrega da informação seja assegurada.

Existe um grande número de tipos de ataques em VANETs (AZEES; VIJAYAKUMAR; DEBORAH, 2016) (BARIAH *et al.*, 2015). As Aplicações de Segurança são muito importantes já que as mesmas estão relacionadas diretamente aos condutores e passageiros dos veículos e suas vidas. O objetivo dos ataques é criar problemas para os motoristas, e tornar os serviços da rede indisponíveis.

Temos como exemplos de ameaças, os Ataques Sybil e os Ataque de Negação de Serviço (Denial of Service - DoS) dentre muitos existentes. Os agressores estão em constante evolução e modificando seus padrões de ataque continuamente.

A motivação desse trabalho surgiu da necessidade de se oferecer uma ferramenta que contribua para tornar o ambiente das Redes Veiculares mais seguro e confiável, onde o crescente aumento de possíveis ataques de pessoas mal intencionadas, o trânsito cada vez mais precário e a falta de segurança na comunicação entre os veículos e/ou entre os veículos e infraestrutura, possam vir a ser mitigados evitando acidentes, fatalidades e perdas econômicas.

1.1.1 Objetivo Geral

Desenvolver um Sistema de Detecção de Ataques Sybil, em Redes Veiculares baseado na Técnica de Extreme Learning Machine (ELM), capaz de receber, preparar e analisar os

dados de mobilidade de uma rede veicular, com o objetivo de detectar esses ataques de forma ágil e precisa, nas redes monitoradas, reduzindo assim a taxa de prejuízos causadospor nós maliciosos nas VANETs e aos usuários da mesma como um todo.

1.1.2 Objetivos Específicos

- Desenvolver e implementar cenários que simulem ataques Sybil nas redes veiculares(VANETs)
- Pré-processar os dados obtidos das simulações e analisá-los usando a técnica de Extreme Learning Machine.
- Buscar obter um bom desempenho computacional (baixo tempo de treinamento) e bom nível de taxa de acertos.
- Aplicar algorítmo de classificação nos veículos identificando suspeitos de comportamento malicioso (Sybil) ou normais, buscando otimizar a segurança da rede.
- Propor uma solução que possa processar grande quantidade de informações recebidas continuamente sem que os resultados dos treinamentos prévios seja perdido.

1.1.3 Metodologia

As simulações foram realizadas utilizando o Simulation of Urban Mobility (Sumo versão 1.2.0), escolhido para gerar os dado de mobilidade dos veículos em cenário urbano.

O algoritmo de detecção utilizado e a preparação dos dados foram desenvolvidos utilizando a linguagem de programação Python3.

A aplicação do Método de Aprendizado Extremo, para o treinamento e a classificação dos dados foi implementado no KERAS/Tensorflow.

Para cada experimento, foi gerado um arquivo de simulação. Neles, os dados da mobilidade dos veículos deram origem a matrizes representando os deslocamentos dos nós veiculares num tempo estimado.

Devido aos nós agressores Sybil (nós virtuais), apresentarem movimentos erráticos no tempo, e não poderem usar o posicionamento dos nós veiculares verdadeiros, avaliou-se o método proposto, comparando-se as similaridades entre os padrões de movimentação dos veículos.

1.1.4 Organização do Trabalho

Este trabalho foi organizado da seguinte forma:

No presente capítulo, foi apresentada uma introdução sobre as Redes Veiculares, aspectos de Segurança, características e vulnerabilidades bem como os conceitos do tipo de ataque Sybil. Também foi apresentado a motivação para o desenvolvimento desse trabalho e os objetivos que busco atingir.

O Capítulo 2 apresenta a Fundamentação teórica sobre redes veiculares, sua segurança, e Técnicas de Aprendizado de Máquina; abordando a definição, características, arquiteturas e aplicações das Redes de Veículos, os desafios à sua segurança, bem como aplicação de Sistemas de Detecção de Intrusos para defesa das VANETs.

O Capítulo 3, por sua vez, aborda alguns trabalhos relacionados aos estudos e definições da aplicação de técnicas de Aprendizado de Máquina voltadas à Sistemas de Detecção de Ataques Sybil, que conduziram a proposta deste trabalho

O Capítulo 4 trata da proposta desta pesquisa, especificando a aplicação da Técnica de Aprendizado de Máquina Extremo implementada para identificação dos ataques à Rede Veicular.

O Capítulo 5 mostra os resultados obtidos nas simulações efetuadas do algoritmo desenvolvido, cenários utilizados e as considerações sobre os experimentos efetuados.

No Capítulo 6 apresenta-se a conclusão deste trabalho.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 REDES VEICULARES

O interesse e progresso no campo das Redes Veiculares (VANETs) tem tido um crescimento acelerado ao longo dos últimos anos. As VANETs compreendem comunicações veículo-a-veículo (V2V) e veículo-a-infraestrutura (V2I), utilizando para isso tecnologias de rede local wireless. O conjunto de aplicações voltadas para as redes veiculares (Ex: aviso de colisão e informações de tráfego para os motoristas e autoridades de trânsito), os recursos (espectro de frequência licenciado, fonte de energia recarregável), e ambiente (Ex: padrões de fluxo de tráfego veicular, questões de privacidade) tornam as VANETs uma área única das comunicações wireless. Nessa seção apresentamos as redes veiculares, componentes e estrutura.

2.1.1 Definição

A indústria automobilística e os avanços da tecnologia sem fio contribuiram para o desenvolvimento de redes veiculares (VANETs) que são um subtipo de Mobile Ad hoc NETworks (MANETs) onde os nós são veículos ou unidades de acostamento (Road Side Units - RSUs).

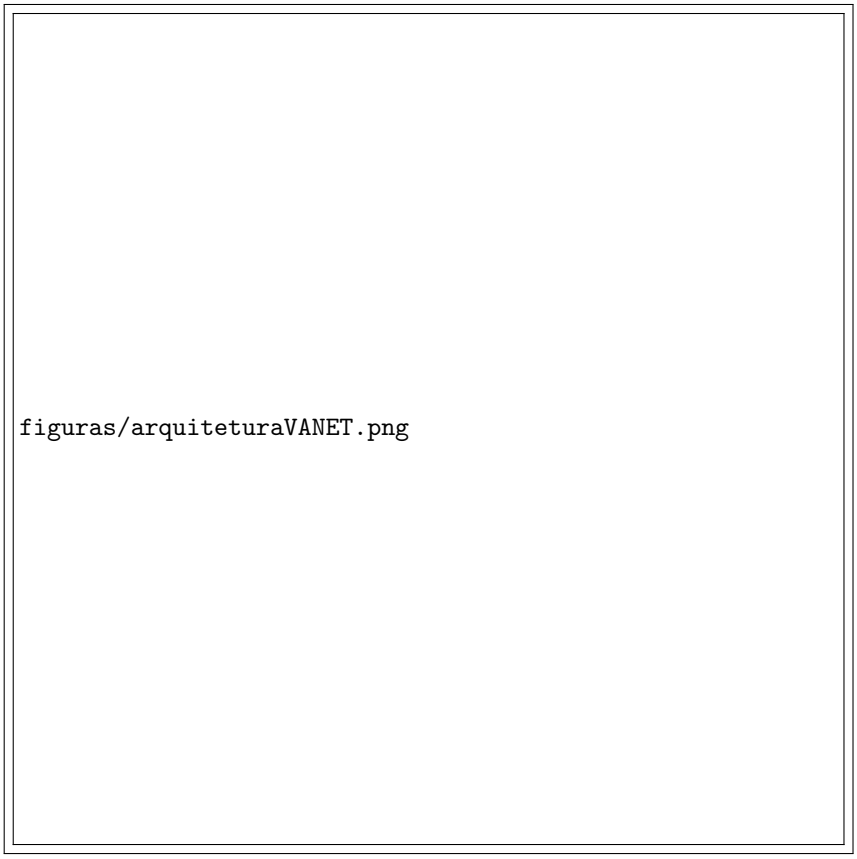
As VANETs são um tipo especial de redes projetadas para melhorar a experiência dos usuários em dirigir veículos fornecendo para esses usuários diferentes serviços tais como online video streaming, acesso internet, direção autônoma (carros autônomos), etc. Nelas, os nós de rede que são os veículos, são equipados com sensores que agrupados formam a Unidade de Bordo ou On-board Unit (OBU), para sensoriamento das atividades em sua vizinhança. Essas unidades funcionam de forma distribuída e cooperativa para facilitar o intercâmbio de informações cruciais tais como a velocidade do veículo, localização, direção e outras informações relativas às condições do tráfego nas estradas, com outros nós para uma melhor dirigibilidade nas estradas.

2.1.2 Características

As VANETs apresentam as seguintes características de acordo com (TANUJA *et al.*, 2015).:

- Topologia Dinâmica: A elevada velocidade dos veículos nas VANETs leva à uma mudança constante na topologia e a disponibilidade de múltiplos caminhos faz com que as redes veiculares tenham uma topologia em constante mudanças.

Figura 2 – Modelo das redes veiculares



Fonte – (SINGH; NANDI; NANDI, 2019)

- Padrões de Movimentação Previsíveis: Apesar da movimentação veloz do veículo parecer difícil de prever a localização e a sua direção, no ambiente da VANET, a maioria dos veículos se movimentam em estradas e rodovias pré-definidas. Isso permite o uso de padrões de movimentação previsíveis no projeto de rede.
- Interação com sensores à bordo: Os sensores são os meios de comunicação na VANET. Os Sensores podem comunicar ao centro de dados principal a leitura dos dados associados a velocidade e direção do veículo. Portanto para formação de link e nos protocolos de roteamento, os sensores podem ser usados.
- Inexistência de falta de armazenagem e capacidade de energia: Os nós nas VANETs dispõem de geração de energia e sua armazenagem comparada as Redes de Sensores.

2.1.2.1 Desafios

Em (TANUJA *et al.*, 2015) as redes veiculares apresentam vários desafios para sua implantação tais como rede desconectada com frequência, dificuldade de atrasos, segurança.

- Frequente desconexão da rede: A natureza veloz dos veículos descreve a topologia dinâmica na qual por sua vez requer uma infraestrutura (RSU) para ser conectada (V2V, V2I). A ausência de tal infraestrutura resulta em desconexões frequentes.
- Restrições de Atraso rigorosas: Em determinada situação, a entrega de mensagens de emergência em tempo hábil é extremamente essencial comparado à somente as altas taxas de dados.
- Modelagem de mobilidade: Como uma das características da VANET é a topologia dinâmica, para desenvolver seu ambiente eficazmente e eficientemente, faz-se necessário um modelo de mobilidade perfeito.
- Segurança e Autenticação: A maior questão na comunicação V2V é manter a segurança do conteúdo da mensagem. Para utilizar a informação tão cedo quanto possível, o conteúdo da mensagem recebida tem de ser verificado dentro de um curtíssimo tempo. A Autenticação é assegurada com garantia de que a comunicação é autêntica em suas entidades. Somente quando mensagens difundidas são geradas por remetentes legais, os veículos devem reagir aos eventos informados.
- Integridade e Confidencialidade: A Integridade está relacionada com a estabilidade das mensagens. É um desafio real assegurar o ordenamento das mensagens, por exemplo, as mensagens são recebidas como enviadas, sem qualquer modificação, inserção, e reordenamento. A Confidencialidade é também um grande desafio, dar a garantia de que a privacidade dos motoristas contra observadores não autorizados está segura.
- Disponibilidade e Escalabilidade: De fato, mesmo um potente canal de comunicação pode em qualquer caso, enfrentar alguns ataques, como por exemplo, Negação de Serviço (Dos), o qual pode cortar a rede. Em seguida, a Disponibilidade é também outra grande questão em VANET. O termo Escalabilidae implica que apesar do fato de que o volume de atividade expanda, não deveria haver nenhuma degradação de desempenho ou mesmo apagão da rede, sem modificação dos componentes de sistema e protocolos.

2.1.3 Arquitetura

Conforme (TANUJA *et al.*, 2015) os componentes básicos principais de uma VANET são a Unidade Certificadora (Authority Unit -AU), unidade de bordo (On Board Unit - OBU), e as Estações de Base (Base Stations: Road Side Unit - RSU, torres de celular - 3G/4G/LTE/5G). As estações base são unidades de comunicação localizadas ao largos das vias e estradas em acostamentos; enquanto as OBUs são unidades de comunicação montadas em veículos. A estação base pode agir como host de aplicação que fornece serviços a OBU. A OBU é um dispositivo que usa esses serviços através da Unidade Certificadora (Authority Unit - AU). A aplicação pode residir em uma estação base ou em uma OBU. O dispositivo que hospeda a aplicação é chamado de provedor de serviço e o dispositivo que usa a aplicação é chamado de usuário. Cada veículo é equipado com uma OBU e um conjunto de sensores para coletar a informação e então a enviar como uma mensagem para outros veículos ou unidades de beira de estrada através do meio de comunicação wireless. A estação base também pode conectar-se à Internet ou a um outro servidor que permita OBUs de múltiplos veículos de conectarem à Internet.

A arquitetura VANET consiste de Autoridade Central confiável (Trusted Authority - TA) que possui sob sua administração múltiplas estações base que por sua vez gerenciam inúmeros veículos se movimentando numa estrada. Cada veículo é montado com uma OBU que possui a habilidade de comunicação. Considerando que o alcance de transmissão de qualquer veículo é mais do que a largura total da estrada, a posição de um veículo não tem efeito na comunicação. Tanto a OBU quanto a estação base são equipadas com chave privada / chave pública e uma chave compartilhada que são fornecidas pela Authority Unit (AU).

A AU (Authority Unit), entidade certificadora, distribui chaves e certificados temporários às OBU/estações base. Cada veículo possui um dispositivo à prova de adulteração para armazenar diferentes chaves. Um veículo receberá um certificado temporário no momento de autenticação pela AU através da estação base correspondente para a comunicação inter-veículo. O módulo da AU executa a encriptação de mensagens baseada na sua necessidade. Se a comunicação é entre V2V e V2I, existe a necessidade de realizar encriptação. Cada veículo deve estar conectado a qualquer uma das estações base mais próximas através de sua OBU. A AU é muito requisitada no caso de um veículo de emergência, uma vez que ele precisa comunicar-se com outros veículos de uma forma bastante ágil, e o tempo e velocidade da comunicação desempenham um papel importante nesses casos emergenciais. Em tais casos, a comunicação V2I não deve implementar métodos de criptografia para uma comunicação ágil e rápida. Como uma

unidade de beira de estrada pode ser conectada com qualquer número de veículos, em sua área de cobertura, então por consequência,a capacidade de armazenamento das unidades de beira de estrada também deve ser considerado como um requisito importante.

2.1.3.1 Comunicação Veículo a Veículo

De acordo com (TANUJA *et al.*, 2015) a comunicação V2V (Vehicle to Vehicle) é projetada para permitir carros conversarem uns com os outros. O sistema usa uma região do espectro de frequência de largura de banda de 5.9GHz usando o protocolo wireless padrão 802.11p para vários serviços de segurança pública. Essa tecnologia é para segurança e o intercâmbio de dados wireless é dinâmico entre veículos na proximidade que oferecem a oportunidade para para significantes melhorias de segurança. Os dados intercambiados podem incluir a posição do veículo, velocidade, ângulo de direção, condiçoes de freio, condição de sinal de dobrar, número de pessoas no veículo, etc.

As comunicações V2V habilitam um veículo detectar ameaças e perigos com a conscientização da posição de outros veículos e da ameaça ou perigo que eles possam representar. As mensagens podem advir do uso de tecnologias não-veiculares tais como comunicação GPS. A tecnologia de comunicação V2V usa Comunicações de curto alcance dedicadas (Dedicated Short Range Communications - DSRC).

Vantagens da Comunicação V2V :

- Permite comunicação de curto e médio alcance.
- Não precisa de qualquer infraestrutura de beira de estrada.
- Menor custo.
- Suporta entrega de mensagens curtas.
- Minimiza a latência no enlace de comunicação.
- É rápida e confiável e fornece segurança em tempo real.
- Protege veículos de perigos na estrada em potencial e melhora a segurança.

Desvantagens da Comunicação V2V:

- Frequente fragmentação da topologia devido à alta mobilidade.
- Problemas na comunicação de longo alcance.
- Uso de protocolo tradicional é desafiador.
- Problema na difusão de mensagens em alto tráfego e contra forças ambientais.

2.1.3.2 Aplicações de V2V

As Aplicações são classificadas em 3 categorias:

1. Aplicações de Seguraça na estrada
    - Evitando colisões;
    - Evitando obstáculos fixos ou móveis;
    - Distribuindo informações de tempo/clima.
2. Aplicações de assistência à condução veicular
    - Auxiliar na travessia do veículo;
    - Prevenção da ocorrência de rotas retas ou rígidas.
3. Aplicações de Comforto
    - Acesso móvel à Internet;
    - Mensagem Eletrônica;
    - Bate-papo Inter-veicular;
    - Jogos em Rede, etc.

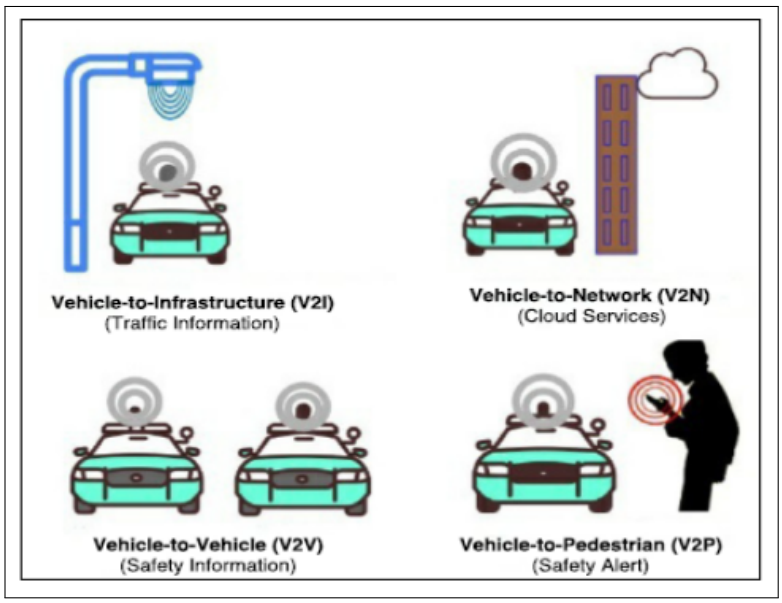
2.1.3.3 Comunicação Veículo para Tudo

Conforme (SINGH; NANDI; NANDI, 2019) o sistema de comunicação veicular inclui veículos e outras entidades de comunicação à sua volta tais como as Estações Base (Base Stations) ou Roadside Units (RSUs), Nuvens (Clouds), Grade (Grid) e Redes de Névoa (Fog Networks), a Internet, dispositivos transportados por um indivíduo e um pedestre, etc. O objetivo da Comunicação Veicular é buscar garantir a segurança na estrada, evitar acidentes de estrada, reduzir o consumo de combustível e emissões de carbono, poupar tempo e ofertar um novo nível de conforto ao dirigir. Para atingir esses objetivos, informação é trocada entre veículos e outras entidades de comunicação.

Esse tipo de Comunicação é referenciada como Comunicação Veículo-para-Tudo (Vehicle-to-everything - V2X). Ela incorpora comunicações como.:

- Vehicle-to-Vehicle (V2V),
- Vehicle-to-Infrastructure (V2I),
- Vehicle-to-Network (V2N),
- Vehicle-to-Pedestrian (V2P),
- Vehicle-to-Device (V2D)
- etc.

Figura 3 – Cenário V2X



Fonte – (SINGH; NANDI; NANDI, 2019)

Desta forma, V2X possibilita aos veículos de comunicarem com seus respectivos ambientes. Como mostrado na Figura.3, V2I, V2N, V2V, e V2P auxilia otimizar fluxo de tráfego e congestionamento, acesso a Nuvem e serviços de Internet avisam aos veículos em suas proximidades por perigos e acidentes e alerta aos pedestres por possíveis colisões, respectivamente.

2.1.3.4 Comunicação V2I

O Sistema de Comunicação V2I deve conter.:

a. Unidade de Bordo ou Equipamento de Bordo (Vehicle On - Board Unit ou Equipment - OBU/OBE)

b. Estações Base ou Unidade de Beira de Estrada ou Equipamento de Beira de Estrada (Base Station ou Road Side Unit ou Equipment - RSU/RSE)

c. Canal de Comunicação Seguro.

As OBUs são o lado veicular do Sistema V2I. As OBUs são compostas logicamente de um Rádio Transmissor (tipicamente DSRC), um sistema GPS, um processador de aplicações e interfaces de rede de comunicações para os sistemas veiculares e interface veicular Homem Máquina (Human Machine Interface - HMI). As OBUs fornecem comunicação entre ambos, veículos e unidades de beira de estrada, bem como, entre veículos e outros veículos vizinhos nas proximidades. As OBUs transmitem mensagens de status regularmente para outras OBUs para darem suporte às aplicações de segurança entre veículos. Em intervalos de tempo, as

OBUs coletam dados para darem suporte às aplicações públicas. As OBUs acomodarão o armazenamento de muitos instântaneos de dados, dependendo de sua capacidade de memória e capacidade de comunicação. Depois de algum período de tempo, o dado mais antigo é sobrescrito. As OBUs também agrupam dados veiculares juntamente com dados de GPS como séries de instântaneos para transmissão para as unidades de beira de estrada. Essas unidades estão localizadas em cruzamentos e outros locais estratégicos e provêem a interface de comunicação aos veículos dentro de seus alcances.

Uma estação base ou unidade de beira de estrada é composta também de um transmissor de rádio (DSRC, WAVE, etc), um processador de aplicação, e de interface para rede de comunicações V2I. Ela também possui uma unidade de GPS acoplada. Através de uma interface adicional, ela pode dar suporte aplicações de segurança da infraestrutura local. A unidade de beira de estrada é conectada a rede de comunicações V2I. Usando sua interface para a rede de comunicações V2I, ela pode enviar dados privados para e a partir dos OEM´s. A unidade de beira de estrada administra a priorização de mensagens para e a partir dos veículos. A OBU também tem uma série de prioridades dentro de suas aplicações e prioridade também deve ser algo definido dentro da unidade de beira de estrada para assegurar que a largura de banda disponível não seja excedida. Aplicações de segurança local e veículo a veículo têm de ter a prioridade mais elevada, as menssagens associadas com aplicações de várias redes públicas e de redes devem ter a prioridade mais baixa.

2.1.3.5 Comunicação Híbrida

A estação base ou unidade de beira de estrada pode conectar-se às redes infraestruturadas ou à Internet, permitindo a OBU acessar a rede infraestruturada. Nesse caso é possível que a OBU possa conectar-se em qualquer hospedeiro baseado na Internet. A OBU também pode comunicar com outros hosts por aplicações não relacionadas à Segurança, usando a comunicação de redes rádio celular (GSM, GPRS, UMTS, HSDPA, WiMax, 4G e 5G). Ela combina ambas as comunicações Veículo-a-Veículo (V2V) e Veículo-a-Infrastrutura (V2I). Neste cenário,um veículo pode comunicar com a com a infraestrutura de beira de estrada quer com modalidade de salto único (single hop) ou salto múltiplo (multi-hop), dependendo da distância, por exemplo, se ela pode ou não acessar diretamente a estaçao base ou unidade de beira de estrada. Ela habilita a conexão de longa distância à Internet ou à veículos que estão muito distantes.

2.1.3.6 Comunicação Infraestrutura para Infraestrutura (I2I/RSU to RSU):

A estação base ou unidade de beira de estrada conecta à Internet e produz a informação necessária para o usuário, quando o veículo está conectado a uma estação base ou unidade de beira de estrada circundante. O veículo com aplicação VANET contém o Sistema de Posicionamento o qual identifica a unidade de beira de estrada próxima. Uma vez que o veículo identifica qual a unidade mais próxima, ele envia o pacote de mensagem Hello para obter confirmação. Toda informação será fornecida para o usuário, quando ele estiver registrado na estação base ou numa unidade de beira de estrada (no caso do envio de informação de advertência entre 2 veículos não há necessidade de registro).

A autenticação é feita através de algoritmo de assinatura digital. Isso fornece uma chave única para todos os usuários que tenham se registrado em uma estação base ou unidade de beira de estrada. Estações Base ou unidades de beira de estrada são colocadas em cruzamentos ou no final de uma estrada. Quando o veículo sai de um alcance de uma estação base ou unidade de beira de estrada em particular para uma outra, ocorre um esquema de transição chamado de handover. A informação de flexão será transferida da entidade de comunicação antiga para a nova . O serviço fornecido pela entidade d ecomunicação é chamado de Serviço VANET Orientado.

2.1.4 Aplicações em VANET

A Rede Veicular apresenta uma grande gama de aplicações variando desde Sinalização de Tráfego até aplicações de Entretenimento (TANUJA *et al.*, 2015).

- Sinalização de Tráfego: Com o auxílio das tecnologias das VANETs, a comunicação a partir do Semáforo pode ser criada. Aplicações de Segurança são aquelas nas quais veículos lentos ou estáticos difundirão mensagens de alerta para sua vizinhança. Notificação sobre congestionamentos na estrada que pode ser usada para rotear e para planejamento da jornada. As Redes Veiculares têm se mostrado particularmente úteis para Gerenciamento de Tráfego.
- Visão Aprimorada: Na visão aprimorada, os motoristas obtém uma visão razoável dos veículos e checagem em ambientes enevoados, e podem podem obter reconhecimento da presença de veículos cobertos por blocos, estruturas, e por diferentes veículos.
- Informações: Dados relativos ao clima podem ser redefinidos/solicitados por uma aplicação via Comunicação DSRC (Dedicated Short Range Communica-

tion). Em pós-acidentes, uma notícia mostraria mensagens de alerta sobre suas circunstâncias para pelotões de veículos para repassarem os dados à vigilância de trânsito para cuidados. Servidores de Parques de Estacionamento fornecendo a disponibilidade de vagas para possiveis clientes. Para a conveniência do veículo, mapas de zonas urbanas ou rurais fornecem dados que permitem fugir de condições de congestionamentos e condiçoes de infortúnio, inclusive mostrando formas de evitar circunstâncias que levem a perdas de tempo.

- Segurança: Aplicações de Segurança incorporam alertas de impacto imediato, reconhecimento de obstáculos a frente e como evitá-los, difusão de mensagens de crises, evitar acidentes na rodovia ou de trens, colaborador de virar a direita/esquerda, cautela crescente no trajeto, ajuda no sinal de parada obrigatória e condições de acidentes na rua, reforço na escolha em cruzamentos, direção assistida (Ex: Alerta de impacto, fusão de caminhos, etc.)
- Entretenimento: Entretenimento vem sob Aplicações não relacionadas à Segurança. Inúmeras aplicações focam o entretenimento dos passageiros que gastam um longo período no trânsito. Tais como fornecendo acesso Internet access, rádio FM, comércio de jogos móveis, multimedia, videostreaming. A comunicação dessas aplicações é sob demanda somente respostas baseadas em requisições.

2.2 SEGURANÇA EM VANETS

Em (MISHRA; SINGH; KUMAR, 2016), a transmissão insegura de informação vital pela comunicação na VANET pode levar a uma catástrofe. Dessa forma, a informação precisa ser exata, eficiente e confiável. Toda e qualquer tarefa no domínio da Rede Veicular tem o objetivo de prover segurança nas estradas eficientemente através do frequente compartilhamento de informação entre os nós da rede. Qualquer tipo de ataque bem sucedido pode levar a sérios acidentes, com perdas de vidas ou perdas econômicas.

Segurança é necessária na Rede Veicular (Vehicular Ad-hoc network) pelas seguintes razões:

- Informações Sensíveis são difundidas na VANET, o que, por sua vez, atrai vários atacantes.
- Facilidade de ataque devido ao modelo de infraestrutura aberta.
- Altíssimas chances de ameaças à privacidade.

- Conexões intrusas são muito facilitadas devido a frequente mudança topológica.

2.2.1 Requisitos de Segurança

A Rede Veicular (MISHRA; SINGH; KUMAR, 2016), foca na melhoria crescente do transporte seguro, prevenção de colisões, eficiência de tráfego e fornecimento de entretenimento. Então alguns pré-requisitos devem ser assegurados pelo Sistema de Segurança implantado.:

- Autenticação: Fornece uma certeza de que a informação/messagem é gerada por um usuário legítimo. Na VANET os nós respondem de acordo com a informação recebida a partir da outra extremidade da comunicação, então é muitíssimo necessário que a informação propagada no sistema seja verdadeira e gerada por um usuário legítimo.
- Confiabilidade: Os dados recebidos na comunicação devem ser corretos e concretos. Uma verificação periódica do sistema é feita para eliminar informações incorretas.
- Integridade: A informação recebida não deve ser alterada por um usuário não autorizado, Tal alteração pode danificar o sistema e pode ocasionar sérias casualidades catastróficas.
- Anonimicidade: A maioria dos usuários estão conduzindo veículos nas Redes veiculares. Então medidas de segurança devem assegurar a privacidade de todos os nós genuínos.
- Disponibilidade: Esse sistema manuseia dados urgentes, então os dados devem estar disponíveis para todos os usuários autorizados de forma fácil e eficientemente.
- Manuseio de Atraso: Informação de Segurança é sensível ao tempo, dessa forma atrasos devem ser evitados e tratados.
- Confidencialidade: Dados Sensíveis não devem ser acessados por usuários não autorizados.

2.2.1.1 Desafios para Segurança em VANETs

Conforme (MISHRA; SINGH; KUMAR, 2016) a VANET possui um conjunto de várias características que fornecem a base de independência no campo de sua atuação. Contudo, essas características algumas vezes criam obstáculos na implantação da Rede Veicular. Tais

desafios são classificados como desafios técnicos (cobrindo o gerenciamento da dinamicidade da rede, gerenciamento de atraso, congestionamento e análise de colisão, impacto atmosférico e desafios de Segurança) e também desafios sociais e econômicos(cobrindo o impacto de custo e aceitação social da VANET).

A Rede Veicular fornece medidas de segurança e de análise de tráfego, de forma que a informação comunicada deve permanecer segura e a rede deve ser robusta. Esses desafios são essenciais para se atingir uma Rede Veicular segura e eficiente.

Os maiores desafios em segurança a serem conquistados pelo sistema de segurança na VANET são:

- Consistência de Dados: Qualquer alteração maliciosa em informação crítica vital pode levar à acidentes. Para evitar atividade maliciosa de nós autenticados e não-autenticados, que causem inconsistências nos dados, alguns mecanismos precisam ser projetados. Checagem cruzada da informação recebida de vários nós é feita para evitar tais atividades.
- Alta Mobilidade: Uma VANET é uma rede extremamente móvel, então ela precisa de algoritmos menos complexos para segurança apesar de ser capaz de alto processamento e de elevada potência de armazenagem.
- Tolerância à Erro: A ação de receber e responder em uma VANET é muito rápida, de forma que qualquer engano nos protocolos ou algoritmo pode danificar o sistema severamente. Assim os protocolos precisam ser projetados tendo isso em consideração.
- Controle de Atraso: A informação compartilhada nas redes veiculares é sensível ao tempo. Para atingir restrição de tempo real, criptografia e outro algoritmo usado na segurança deve ser rápido e eficiente.
- Gerenciamento de Chave: Todos os algoritmos usados em Segurança na VANET são dependentes de chave. Assim, a criação, manutenção e distribuição de chaves têm de ser gerenciadas especialmente.

2.2.2 Sistema de Detecção de Intrusos

Conforme (LEE *et al.*, 2017), devido ao rápido e crescente desenvolvimento das redes de computadores em todas as áreas, elas se estabeleceram nos negócios, entretenimento, media social, indústria, educação, saúde, transportes e governos.

O conhecimento diverso e expertise complexa requerida para Cyber-Segurança, faz com que muitas vezes o pessoal técnico pode não ser capacitado para gerenciar essas redes. Como resultado, as Redes são suscetíveis de ataques, o que levou ao surgimento dos Sistemas de Detecção de Intrusos (Intrusion Detection Systems - IDS).

Os Sistemas IDS são usados para proteger as Redes de Computadores de ataques e filtrar ou detectar comportamentos maliciosos lançados por atacantes (DAS; NENE, 2017).

Um dos métodos mais comuns de detecção de intrusos é baseado em Assinatura usando algoritmos de Data Mining (Ex: Modelos white list ou black list. São usados métodos de análise e estatísticos para criar as assinaturas para os sistemas.

Contudo, os modelos são criados a partir de dados abrangentes e detalhados, coletados através de redes protegidas. Quando dados desconhecidos são enviados à esses sistemas, eles são mais provavelmente rotulados como categorias incorretas.

O resultado disso, se o sistema carece de mecanismos discriminantes, serão muitos falsos positivos ou falsos negativos como resultados. Para Sistemas baseados em Aprendizado de Máquina (Machine Learning - ML), seus modelos podem aprender a partir de dados de treinamento e ajustar os resultados de classificação de saída.

O aprendizado não é baseado apenas em um único tipo de dados, como os modelos white ou black list, mas é baseado também em todos os tipos de dados. Através de computação e comparação, dados para um sistema baseado em ML são classificados como a categoria mais provável. Muitas técnicas de ML têm sido usadas para construir IDS's (Ex: Algoritmos genético e fuzzy, clustering e K-nearest neighbor (k-NN), etc.

2.2.3 Aprendizado de Máquina - Machine Learning

Segundo (DAS; NENE, 2017) o Aprendizado de Máquina é uma das técnicas de Inteligência Artificial (Artificial Intelligence - AI), que têm sido amplamente utilizadas no campo da engenharia pelas últimas duas décadas. As preocupações principais em Aprendizado de Máquina são a eficiência e acurácia dos sistemas computacionais.

A principal razão do porquê Machine Learning se tornou tão popular é por que essa técnica somente requeria programação implícita em Ciência da Computação, que desse aos computadores a habilidade de aprender.

O objetivo principal dos métodos de Machine Learning é aprender a partir de dados existentes e treinar um modelo preditivo com relação a prever os resultados e tendências futuras.

Existem muitas técnicas de Machine Learning que podem ser usadas para uma aplicação de classificação tais como as Máquina de Vetor de Suporte (Support Vector Machine - SVM), Máquina de Aprendizado Extremo (Extreme Learning Machine - ELM), Rede Neural Artificial (Artificial Neural Network) e outras. Entretanto, os algoritmos tradicionais de Machine Learning são na maioria baseados no Gradiente de Descida, o qual, se depara com vários problemas como taxa de aprendizado imprópria, mínimo local, sobreajuste e subajuste. O Backpropagation (BP) e a característica iterativa desses métodos, os tornam inapropriados para aplicações de ponta com dados em larga escala por causa dos parâmetros nos algoritmos de Machine Learning precisarem ser ajustados lentamente.

Anteriormente, A SVM era o método de Machine Learning predominante, que era mais extensivamente utilizado. Havia sido provado que ele era eficiente em aplicação de classificação devido à alta acurácia de classificação. Contudo, o ELM então emergiu como um dos métodos que é capaz de superar os métodos de aprendizado convencionais tais como o SVM e a Rede Neural.A implementação de Hardware da ELM é a tendência atual de pesquisa no campo de Machine Learning. O interesse na implementação de Hardware de ELM aumenta devido a várias características de ELM que superam os métodos tradicionais de aprendizado. A ELM surge atualmente como um dos melhores métodos de aprendizado para Classificação bem como sua acurácia de classificação e velocidade de aprendizado.

2.2.3.1 Aprendizado de Máquina - Características

É a capacidade de Máquinas tomarem decisões inteligentes, através de um processo no qual um conjunto de parâmetros limites é treinado para classificar um comportamento desconhecido, importando-se com a acurácia das Decisões.

A tomada de Decisão via identificação de padrões complexos nos dados.

Tipos de Algoritmos.:

1. Aprendizado Supervisionado.:
    - Quando os dados incluídos para treinar o algoritmo incluem a solução desejada (a resposta certa), ou rótulo (label).
2. Aprendizado Não-Supervisionado.
    - Não existe rótulo (label). O algoritmo aprende sem uma resposta certa ou através da redução de dimensão (dados) ou de agrupamento (Clustering).
3. Aprendizado Semi-Supervisionado.

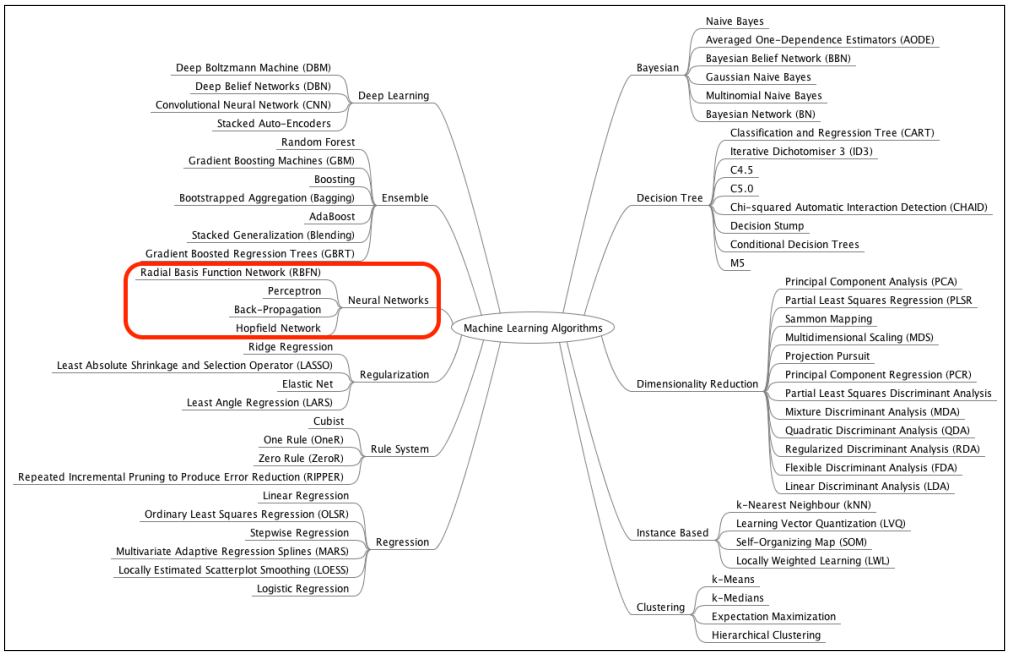
- Apresenta alguns dados com rótulo e outros sem o mesmo.

4. Aprendizado por Reforço.
   - Interação com um ambiente dinâmico com feedbacks em termos de premiações e punições.

   Tipos de Algoritmos Supervisionados.:

   - Classificação.: A variável a ser predita é qualitativa.
   - Regressão.: A variável a ser predita é quantitativa.
   - Predição.: Interesse em performance Preditiva.
   - Interpretação (Inferência).: Interesse em entender a relação entre as variáveis. Exemplos.:
     1. Árvores de Decisão
     2. Support Vector Machines (SVM)
     3. K-Nearest Neighbours (KNN)
     4. Redes Neurais Artificiais (RNA)

Figura 4 – Algoritmos de Machine Learning



Fonte – https://medium.com/@andressasivolella/afinal-o-que-machine-learning-e-redes-neurais-fazem-7c89e1885064

Problema.:

- *Como as Técnicas de Machine Learning irão contribuir na Segurança em Redes Veiculares?*
  - Representando um método direcionado à dados efetivo, tornando a Segurança mais robusta em lidar com tipos de dados heterogêneos onde nenhuma suposição explícita é feita na distribuição dos dados.
  - Ajudando o sistema a ficar mais bem informado para poder tomar decisões dirigidas aos dados mais eficazmente.
  - Aliviando desafios de comunicação.

2.2.4 Redes Neurais Artificiais - RNA

As Redes Neurais Artificiais (RNAs) são modelos computacionais inspirados no sistema nervoso central, ou seja, capazes de realizar o aprendizado de máquina (machine learning) e reconhecimento de padrões.

O tipo mais simples de rede neural artificial foi proposto em 1958 por Frank Rosenblatt, conhecido como perceptron. A palavra em latim para o verbo compreender é “percipio“, e sua forma supina é “perceptum”, ou seja, a rede deve ser capaz de compreender o mundo exterior.

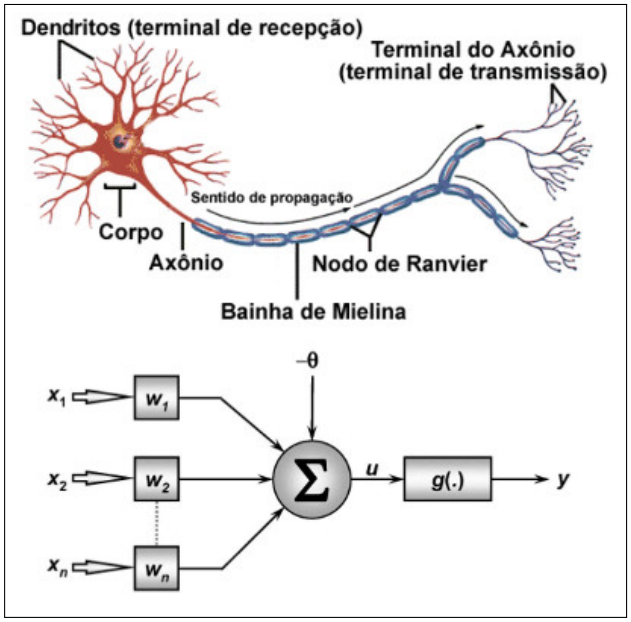
Esse algoritmo de aprendizagem supervisionada considera um período de treinamento (com valores de entrada e saída) para definir se uma nova entrada pertence a alguma classe específica ou não.

Um neurônio recebe um impulso através dos dendritos, processa o sinal e dispara um segundo impulso, que produz uma substância neurotransmissora que flui do corpo celular para o axônio, e então para outro neurônio.

Analogia para criar um “neurônio artificial”:

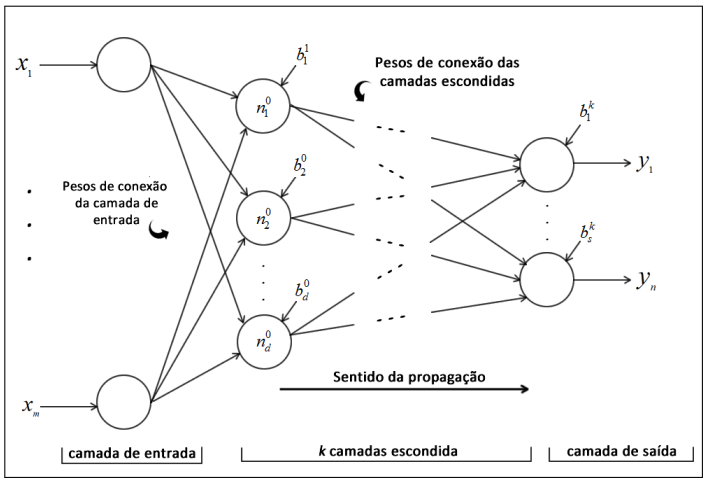
1. Os sinais de entrada $\{x_1, x_2, x_n\}$ são ponderados/multiplicados por $\{w_1, w_2, w_n\}$
2. A função agregadora recebe todos os sinais e realiza a soma dos produtos dos sinais
3. Ao resultado, é somado o limiar de ativação (também chamado de bias ou parâmetro polarizador), soma essa conhecida como potencial de ativação u; o bias é uma constante que serve para aumentar ou diminuir a entrada líquida u, de forma a transladar a função de ativação no eixo de u
4. Obs.: o bias pode ser tratado como “mais um peso”, o que na prática envolve acrescentar

Figura 5 – Esquema Neurônio Natural x Artificial



Fonte – https://www.monolitonimbus.com.br/perceptron-redes-neurais/

Figura 6 – Rede Neural Artificial



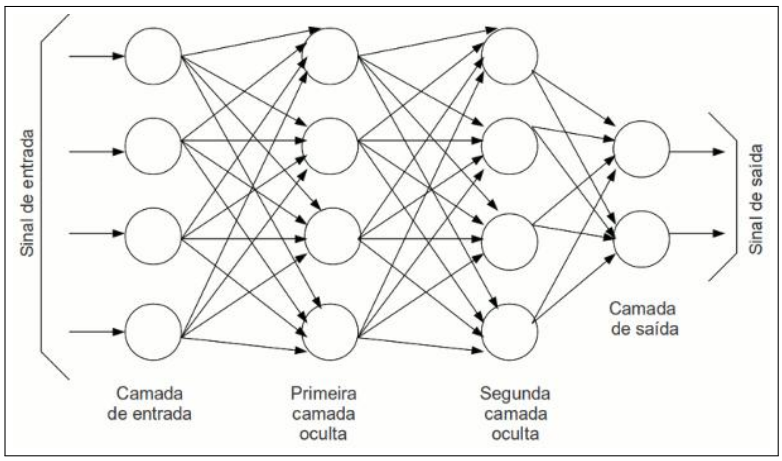
uma nova entrada do tipo $x_{k0} = 1$ com um peso associado w_{k0}.

5. A função de ativação $\{g\}$ é aplicada sobre o potencial de ativação u para deixar o sinal passar ou não.

Todas as saídas da rede são trocadas no início de intervalos discretos chamados de época. No início de cada época, a soma das entradas de cada neurônio é somado e aplicada a função de ativação. Essa função de ativação pode ser uma função bipolar (somente dois valores de saída), uma reta (função linear) ou até uma função gaussiana, hiperbólica, etc.

No caso de uma função bipolar, pode-se considerar a saída igual a “1” se o valor de u (somatório dos produtos) for maior ou igual a "0"e “-1” no caso de $u < 0$.

Figura 7 – Rede Neural Artificial - Unidirecional

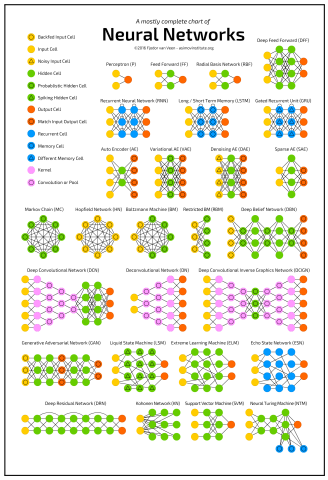


O processo de treinamento tem como objetivo calibrar os pesos de modo iterativo, partindo de valores aleatórios (geralmente entre 0 e 1).

A taxa de aprendizagem $\{\eta\}$ (também um valor entre 0 e 1) diz o quão rápido a rede chega ao seu processo de classificação: um valor muito pequeno causa demora a convergir, enquanto que um valor muito alto pode levar para valores fora do ajuste e nunca convergir.

Assim, o vetor contendo os pesos de um passo de iteração será o resultado da soma de si mesmo no passo anterior com o produto das seguintes parcelas: a taxa de aprendizagem, a amostra de aprendizagem desse passo e a diferença entre o valor desejado (saída "certa") para esse passo e o valor de saída produzido pela rede. Essa diferença é chamada de erro de saída: se for diferente de zero, é aplicada a correção.

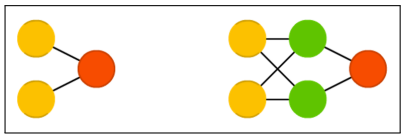
Figura 8 – Tipos de Redes Neurais Reduzido



Fonte – https://www.asimovinstitute.org/neural-network-zoo/

2.2.4.1 Perceptrons (P) e Feedforward Neural Networks - FF ou FFNN

Figura 9 – Perceptrons



Fonte – https://www.asimovinstitute.org/neural-network-zoo/

O Perceptron é um classificador linear, ou seja, os problemas solucionados por ele devem ser linearmente separáveis.

Na estrutura unidirecional (feedforward), todas as saídas dos neurônios de uma camada são conectadas com todos os neurônios da camada posterior obedecendo à direção entrada => saída, sem conexões entre neurônios de uma mesma camada.

2.2.4.2 Redes Neurais Artificiais Feedforward

Existem muitos tipos de Redes Neurais, todavia, as Redes Neurais do tipo feed-forward devem ser uma das mais populares redes neurais.

Uma rede neural feedforward consiste de uma camada de entrada recebendo o estímulo de ambientes externos, uma ou múltiplas camadas escondidas, e uma cama de saída enviando a saída da rede aos ambientes externos.

Os três métodos geralmente usados para treinar essas redes são.:

1. Baseado no Gradiente descendente
2. Método baseado em Otimização Padrão
3. Baseado no Quadrado-Mínimo (Ex.: Aprendizado de Rede pela Função de Base Radial (FBR/RBF))

Tradicionalmente, todos os parâmetros de uma rede unidirecional têm que ser ajustados;

Métodos baseados em gradiente descendente têm sido usados em vários algoritmos de treinamento (p. ex., algoritmo BackPropagation);

Tais métodos consomem grande tempo de treinamento devido ao ajuste iterativo dos parâmetros.

2.2.5 Desvantagens do Backpropagation

Quando a taxa de treinamento $\{\eta\}$ é muito pequena, o algoritmo de treinamento converge muito lentamente. Caso contrário, quando η é muito grande, o algoritmo se torna instável e a rede diverge;

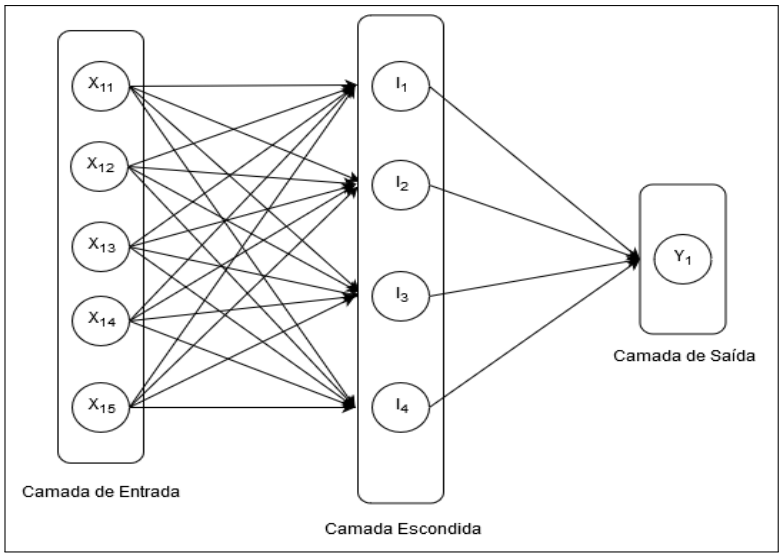
É indesejável que o algoritmo pare em um mínimo local distante do mínimo global;

Redes Neurais podem ser super-treinadas com o algoritmo BP de maneira que a generalização fique prejudicada (overfitting);

Aprendizado baseado em gradiente descendente pode consumir demasiado tempo de treinamento em muitas aplicações.

2.2.6 Máquina de Aprendizado Extremo (Extreme Learning Machine - ELM)

Figura 10 – ELM



Fonte – Elaborado pelo autor.

O nome Extreme Learning Machine - ELM foi dado à tais modelos pelo seu inventor principal Guang-Bin Huang (HUANG; ZHU; SIEW, 2004). De acordo com seus criadores, esses modelos são capazes de produzir um bom desempenho generalizado e aprender milhares de vezes mais rápido do que redes treinadas usando backpropagation. Proporcionam uma melhor performance à uma velocidade de aprendizado muito maior com menos intervenção humana. Na literatura, esses modelos demonstram que também podem superar as Redes Neurais Artificiais (RNA) tradicionais e as Support Vector Machines - SVM (desde que as SVM fornecem soluções sub-ótimas em ambas aplicações de classificação e de regressão).

As Extreme learning machines são redes neurais feedforward também chamadas de Single Hidden Layer Feedforward Neural Networks (SLFN), com conexões aleatórias (não são recorrentes, não usam backpropagation), para classificação, regressão, clustering, aproximação esparsa, compressão e aprendizado de característica; que podem conter uma única ou múltiplas camadas de nós escondidos, onde os parâmetros dos nós escondidos (não somente os pesos conectando as entradas aos nós escondidos) não precisam ser calibrados de forma obrigatória como nas redes neurais clássicas (HUANG; ZHU; SIEW, 2006).

Para evitar calibrar os parâmetros, as ELM iniciam com pesos randômicos e treinam os pesos em uma única etapa de acordo com o ajuste dos Quadrados Mínimos (taxa de erro mínima dentre todas as funções). Isso resulta em uma rede muito menos expressiva mas ela é também muito mais rápida do que o uso de backpropagation (HUANG; WANG; LAN, 2011).

Os nós escondidos podem ser assinalados aleatoriamente e nunca atualizados (por exemplo eles são projeções randômicas mas com transformações não-lineares), ou podem ser herdados de seus ancestrais sem serem modificados. Na maioria dos casos, os pesos de saída dos nós escondidos são geralmente aprendidos em uma única etapa, a qual equivale ao aprendizado de um modelo linear.

2.2.6.1 Máquina de Aprendizado Extremo - Características.:

Esse algoritmo contorna as desvantagens citadas anteriormente. Foi desenvolvido para redes com apenas duas camadas: a camada de entrada e a camada escondida. Os pesos de entrada e os bias da camada escondida são escolhidos aleatoriamente. Os pesos da camada de saída são determinados analiticamente (por exemplo, não há ciclos iterativos para ajuste de parâmetros).

2.2.6.2 ELM: Teoria e Modelagem matemática.:

$$\sum_{i=1}^{\tilde{N}} \beta_i g_i(\mathbf{x}_j) = \sum_{i=1}^{\tilde{N}} \beta_i g(\mathbf{w}_i.\mathbf{x}_j + b_i) = \mathbf{o}_j, j = 1,...,N$$

$$\Downarrow$$

$$\sum_{i=1}^{\tilde{N}} \beta_i g(\mathbf{w}_i.\mathbf{x}_j + b_i) = \mathbf{t}_j, j = 1,...,N$$

- (x_j, t_j): N padrões de entrada;
- w_i : vetor peso do neurônio i da camada escondida;
- b_i : bias do neurônio i da camada escondida;
- $\widetilde{N}$: número de neurônios da camada escondida.
- β_i : vetor peso entre o neurônio escondido i e a camada de saída.

Em forma matricial:

$$\mathbf{H}\beta = \mathbf{T}$$

$$\mathbf{H}(\mathbf{w}_1,...,\mathbf{w}_{\tilde{N}}, b_1,...,b_{\tilde{N}}, \mathbf{x}_1,...,\mathbf{x}_{\tilde{N}}) = \begin{bmatrix} g(\mathbf{w}_1.\mathbf{x}_1 + b_1) & \cdots & g(\mathbf{w}_{\tilde{N}}.\mathbf{x}_1 + b_{\tilde{N}}) \\ \vdots & \cdots & \vdots \\ g(\mathbf{w}_1.\mathbf{x}_N + b_1) & \cdots & g(\mathbf{w}_{\tilde{N}}.\mathbf{x}_N + b_{\tilde{N}}) \end{bmatrix}$$

Figura 11 – Matriz ELM - Pesos e Biases

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_{\tilde{N}}^T \end{bmatrix}_{\tilde{N}xm}, \mathbf{T} = \begin{bmatrix} \mathbf{t}_1^T \\ \vdots \\ \mathbf{t}_{\tilde{N}}^T \end{bmatrix}_{Nxm}$$

Teorema.: Dada uma SLFN (Single Hidden Layer Feedforward Neural Networks)com N neurônios na camada escondida e função de ativação $g : R \to R$ infinitamente diferenciável em qualquer intervalo, para N exemplos de treinamento distintos $(x_i, t_i), x_i \in R^n, t_i \in R^m$ para quaisquer w_i e b_i aleatoriamente selecionados dentro de quaisquer intervalos R^n e R, respectivamente, por qualquer função de distribuição de probabilidade, então com probabilidade 1, a matriz de saída da camada escondida H da SLFN é inversível e $\|H\beta - T\| = 0$.

Se o número de neurônios $\widetilde{N}$ da camada escondida é igual ao número N de exemplos de treinamento, $N = \widetilde{N}$, então a matriz H é quadrada e inversível quando o vetor de pesos w_i e os bias b_i são aleatoriamente escolhidos e, assim, as a SLFNs podem aprender estes exemplos de treinamento com erro zero.

Entretanto, na maioria dos casos o número de neurônios da camada escondida é muito menor do que o número de exemplos distintos de treinamento, $\widetilde{N} << N$, e a matriz H não é quadrada;

Solução de mínimos quadrados com a menor norma: $\beta = H\dagger T$;

$H\dagger$: matriz inversa generalizada de Moore-Penrose da matriz H (pseudo inversa).

2.2.6.3 ELM: Algoritmo

INÍCIO

- Passo 1: Selecionar aleatoriamente valores para os pesos w_i e os bias b_i , $i = 1, ..., N$;
- Passo 2: Calcular a matriz de saída H da camada escondida.
- Passo 3: Calcular os pesos de saída $\beta = H\dagger T$.

FIM

2.2.6.4 ELM: Algoritmo - Características

Menor erro de treinamento: A solução $\beta = H\dagger T$ é uma das soluções de mínimos quadrados de um sistema linear geral $H\beta = T$, o que significa que o menor erro de treinamento pode ser encontrado por esta solução.

Menor norma dos pesos: Além disso, a solução $\beta = H\dagger T$ tem a menor norma entre todas as soluções de mínimos quadrados de $H\beta = T$.

A solução de menor norma é única.

2.2.6.5 ELM: Algoritmo - Matriz PseudoInversa

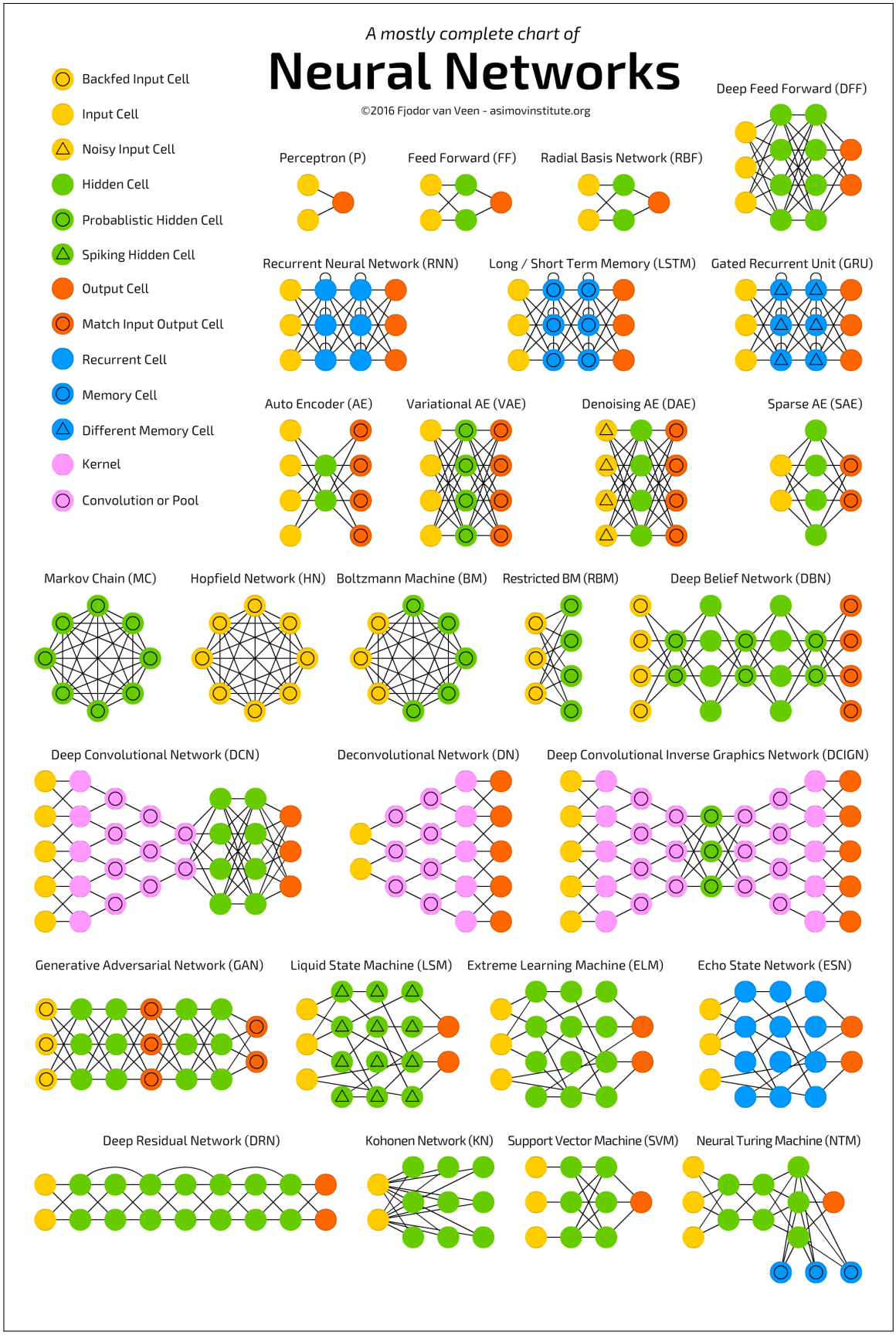
$H\dagger$ satisfaz as seguintes propriedades:

1. $HH\dagger H = H$
2. $H\dagger HH\dagger = H\dagger$
3. $(HH\dagger)^T = HH\dagger$

4. $(H\dagger H)^T = H\dagger H$

Pode ser calculada eficientemente pelo método da Decomposição por Valores Singulares (Singular Value Decomposition - SVD).

Figura 12 – Tipos de Redes Neurais Expandida



Fonte – https://www.asimovinstitute.org/neural-network-zoo/

3 TRABALHOS RELACIONADOS

Em (GU *et al.*, 2016), um método de detecção eficiente de ataque Sybil baseado no padrão de mobilidade do veículo em cenário urbano, é apresentado. Esse método foi desenvolvido para detectar o modo errático do padrão de movimentação dos nós Sybil, o qual inclui o seu tempo de vida e sua movimentação incomum. Nesse método, os padrões de movimentação dos veículos, é representado principalmente usando os Autovalores de sua Matriz de novimentação. Então utiliza-se a Distância de Mahalanobis, que é baseada nas correlações entre variáveis com as quais padrões distintos podem ser identificados e analisados (É uma estatística útil para determinar a similaridade entre uma amostra desconhecida e uma conhecida), para medir a semelhança dos padrões de condução dos veículos. Os resultados em simulações mostraram que o método obteve alta taxa de detecção, com uma baixa taxa de erro. O método proposto levou em consideração apenas os tipos de ataques do modelo racional e simulou apenas condições de tráfego de limitada densidade de veículos.

Em (GU *et al.*, 2017a), o mesmo modelo e método utilizado em (GU *et al.*, 2016) de medida de semelhança do padrão de movimentação veicular em cenário de tráfego próximo de sua capacidade máxima, foi apresentado. Esse método foi proposto para diferenciar a movimentação de nós benignos de nós Sybil, detectando a variação entre seus padrões de movimento. Essa variação de movimentação é refletida na Matriz de deslocamento dos mesmos. Esses padrões são representados principalmente usando os Autovalores de suas matrizes no método de detecção proposto. O método de Classificação kNN é utilizado para classificar os veículos, diferenciando os nós virtuais dos nós benignos. Contudo, a principal desvantagem desse método é sua alta complexidade de tempo de execução. Dois métodos diferentes são apresentados nesse trabalho para abordar essa questão da complexidade. Ainda assim as simulações demonstraram um bom desempenho no controle de erros e alta taxa de detecção.

Em (GU *et al.*, 2017b), o modelo e método abordados em (GU *et al.*, 2016), voltam a ser utilizados, em condições similares à (GU *et al.*, 2017a). Então uma abordagem para detecção do ataque Sybil usando métodos SVM (Support Vector Machine) para classificar os veículos e distingui-los em nós virtuais e nós benignos.

As simulações demonstraram que os nós benignos apresentam padrões de movimentação semelhantes, enquantos os nós virtuais Sybil demonstraram serem inconstantes e erráticos. No método abordado foi usado três Funções de Kernel (Polinomial, Função de Base Radial Gausiana e Perceptron de Multicamada) para classificar os nós veiculares e em todas obtiveram

boas taxas de detecção com baixas taxas de erro.

Em (HAMED; KESHAVARZ-HADDAD; HAGHIGHI, 2018), uma técnica robusta para detecção de ataque Sybil para Redes Veiculares é proposta. A idéia de encontrar algum tipo de relacionamento entre os nós Sybil, é a noção fundamental desse esquema proposto. Os resultados demonstraram que o método apresentado possui um desempenho apropriado em termos de detecção de ataques do tipo Sybil. Foi medido um limite apropriado para a configuração da simulação. E embora, achando o melhor valor limite para diferentes cenários e densidades de veículos, especialmente nos casos onde as RSU´s possuem intersecção de faixas de comunicação, isso se torna um desafio para o esquema. Uma outra área de desafio, poderia ser a implantação de RSUs, pela minimização do número das mesmas em favor de maximização do desempenho da detecção.

De acordo com (REDDY *et al.*, 2017), os ataques Sybil são detectados satisfatoriamente pela técnica proposta. Além disso, pela utilização da Função de Hash e operação XOR (Ou Exclusivo), esta técnica também checa o tempo de existência da identidade do veículo. Um método de criptografia de assinatura digital é usado para estabelecer confiança entre as entidades participantes no processo de comunicação. A privacidade e a segurança é o que mais importa nas VANET´s. Por essa razão esquemas como esse de comunicação eficiente ainda precisam ser revisados e extendidos.

Em (SHARMA *et al.*, 2016), o principal objetivo neste trabalho é apresentar uma nova técnica para proteger a rede veicular do ataque Sybil. É proposto o uso de pseudo-certificados de vida curta no sentido de fornecer privacidade e anonimato para os veículos, bem como reduzir o risco do ataque Sybil na rede veicular. O custo computacional será minimizado pela redução do número de operações de intersecção e da requisição do número de anúncios difundidos para detectar um nó Sybil, comparado com outros métodos de listas de vizinhança. Neste esquema proposto, cada veículo anuncia somente sua lista de nós suspeitos de serem Sybil. Isso reduzirá a sobrecarga na rede. O esquema introduz e descreve um mecanismo robusto de deteção de ataque Sybil. O uso do método de identidade temporária irá reduzir o risco do ataque, pelo fornecimento de privacidade e anonimato ao usuário. Em uma VANET, o anonimato significa manter em segredo a identidade permanente dos veículos em comunicação. Se ocorrer do atacante adquirir a identidade temporária de um veículo, isso não será útil depois da expiração da mesma. A detecção do nó Sybil é feita por todos os veículos neste método (tanto nó honesto quanto Sybil). Assim, o resultado pode mudar facilmente pela alteração da informação da lista da vizinhança. Neste método o uso da lista e identidade temporária se complementam.

(EZIAMA *et al.*, 2018) fornece uma operação veicular para segurança tanto quanto uma comunicação DSRC mais ecológica e eficiente. A natureza dinâmica da topologia da rede veicular permite muitos desafios de segurança para uma comunicação efetiva V2V e V2I. Nos modelos existentes carecem de flexibilidade e funcionalidade suficiente para detectarem os comportamentos dinâmicos de nós maliciosos nos sistemas de comunicação veicular altamente volátil. No método proposto deste trabalho, um modelo de confiabilidade com respeito ao Machine/Deep Learning (ML/DL) é proposto devido a lacuna existente nos modelos de segurança. O modelo orientado por dados oferece solução de desafios de segurança em redes dinâmicas, modelando confiabilidade como um processo de classificação e extraindo aspectos relevantes usando modelos híbridos de classificação como Bayesian Neural Network que combina deep learning como modelagem probabilística para identificação de nós honestos e desonestos na rede. As principais contribuições desse trabalho são:

- Fornece uma comparação de diferentes algoritmos de Machine Learning/Deep Learning (ML/DL) com diferentes técnicas de otimização.
- Fornece um modelo que estima a credibilidade inicial da informação enviada pelos nós veiculares na ausência de experiência prévia acerca desses nós.
- Formula avaliação de confiabilidade via o fornecimento atributos confiáveis sonoros e intuitivos na comunicação V2V com as correlações das mensagens de segurança básica (BSMs).
- Fornece um modelo com informação prévia/especializada em termos de regularização implícita. Isso aprimora a definição de parâmetros em Neural Network, prevenção de over-fitting e o aumento de modelo de desempenho no caso de dados escassos em cenários de rede veicular menos densamente populados.

Em (GROVER *et al.*, 2011), é apresentado um Método de Machine Learning para classificar múltiplos tipos de mau comportamento em Vanet, ataque Sybil inclusive, usando características concretas e comportamentais para cada nó veicular que envia pacotes de segurança. Um framework de Segurança foi projetado para diferenciar um nó malicioso de um nó legítimo. Foram implementados vários tipos de mau comportamento em Vanet através da manipulação da informação presente em pacotes nos pacotes propagados. Esses maus comportamentos são classificados através de características variadas tais como variação da velocidade do nó, força do sinal recebido (RSS), número de pacotes entregues, número de pacotes descartados, etc. Dois tipos de Acurácia de Classificação são medidas.: Binária e Multi-Classe. Na classificação binária,todos os tipos de maus comportamentos são vistos como estando em uma única classe

de mau comportamento; enquanto que na classificação Multi-classe é capaz de categorizar maus comportamentos em classes de maus comportamentos particulares. Esse método proposto foi avaliado usando o WEKA; sendo considerado eficiente na classificação. Os resultados dos expeimentos desse trabalho mostraram que os classificadores Random Forest e o J-48 tiveram melhor desempenho comparado com outros tipos de classificadores, demonstrando assim resultados promissores em identificar nós legítimos e maliciosos. Todavia, nos cenários VANET realistas, o método proposto é inadequado para detectar ataques temporais devido à indisponibilidade de informação de transmissão/recepção de pacote por nó individual.

Tabela 1 – Comparação entre os Trabalhos Relacionados

Método	Referência	Vantagem	Desvantagem
Machine Learning Deep Learning Modelo de Confiabilidade Rede Neural Bayesiana	Eziama et al. (2018) (EZIAMA *et al.*, 2018)	Comparação de diferentes técnicas de classificação com diferentes técnicas de otimização. Avaliação de confiabilidade. Prevenção de over-fitting. Aumento de performance em caso de dados esparsos em cenários de rede veicular menos densamente populados. Modelo eficiente com baixa latência na captura de informação.	Não foram fornecidos experimentos de simulação. Não realizaram análises posteriores nem estimação do comportamento dos nós baseada na informação fornecida com o modelo de Rede Neural Bayesiana proposta.
Comparação entre Lista de Vizinhança de cada RSU´s IOAC (Infrastructure Observation based Affinity Computation).	Hamed et al. (2018) (HAMED; KESHAVARZ-HADDAD; HAGHIGHI, 2018)	Resultados mostram desempenho apropriado em termos de detecção de ataque Sybil.	Complexidade de Controlede ID´s vistas na Intersecção de Faixas de Comunicação. Nenhum procedimento foi introduzido para corrigir erros de detecção, ID´s legítimas como Sybil IDs.
Machine Learning SVM	Gu et al.(2017b) (GU *et al.*, 2017b)	Todas as três funções de Kernel usadas atingiram alta taxa de detecção com baixa taxa de erro.	Somente ataques racionais (que lançam ataques em condições de capacidade máxima) foram avaliados. Alto custo computacional. Elevado tempo de treinamento.
Machine Learning K-NN	Gu et al. (2017a) (GU *et al.*, 2017a)	Bom desempenho no controle de erros. Alta taxa de detecção.	Alta complexidade de Tempo de Execução.
Criptografia de Assinatura Digital Confiabilidade Chave Pública / Privada	Reddy et al. (2017) (REDDY *et al.*, 2017)	Detecção de Ataques Sybil bem sucedidos.	Não foram fornecidos experimentos de simulação.
Classificador de Distância Mínima Mahalanobis (Distância Estatística)	Gu et al. (2016) (GU *et al.*, 2016)	Projetou um formato de Dados (Matriz Padrão de Movimentação Veicular). Definiu variação do padrão de movimento veicular. Medição de Semelhança do movimento veicular. Detecção da inexatidão do padrão de movimento Sybil.	O método proposto levou em consideração apenas os tipos de ataques do modelo racional. Simulou apenas condições de tráfego de limitada densidade de veículos.
Técnica de Geração Dinâmica de Certificados (Criptografia) e Listas de Vizinhança	Sharma et al. (2016) (SHARMA *et al.*, 2016)	Identidades temporárias geradas por técnicas de criptografia. Fornecem Anonimidade/Privacidade aos veículos. Reduz o risco de ataque Sybil. Minimizou custo computacional através da redução do número de operações de intersecção e broadcast requeridos para detecção de nó Sybil. Esquema proposto é eficiente em todos os ambientes de VANET.	Não foram fornecidos experimentos de simulação. Não realizaram análises posteriores. Não fizeram a estimação do comportamento dos nós baseada na informação fornecida.
Machine Learning Classificação Binária e Multiclasse WEKA	Grover et al. (2011) (GROVER *et al.*, 2011)	Os resultados são validados por métricas de avaliação computadas por vários algoritmos classificadores.: Naive Bayes, IBK, AdaBoost1 (com J-48 como classificador base), J-48 Random Forest (RF).	Método proposto não se adequa para identificar ataques temporais devido a indisponibilidade de informação dos pacotes transmitidos/recebidos por nós individualmente.

Fonte – Elaborado pelo autor

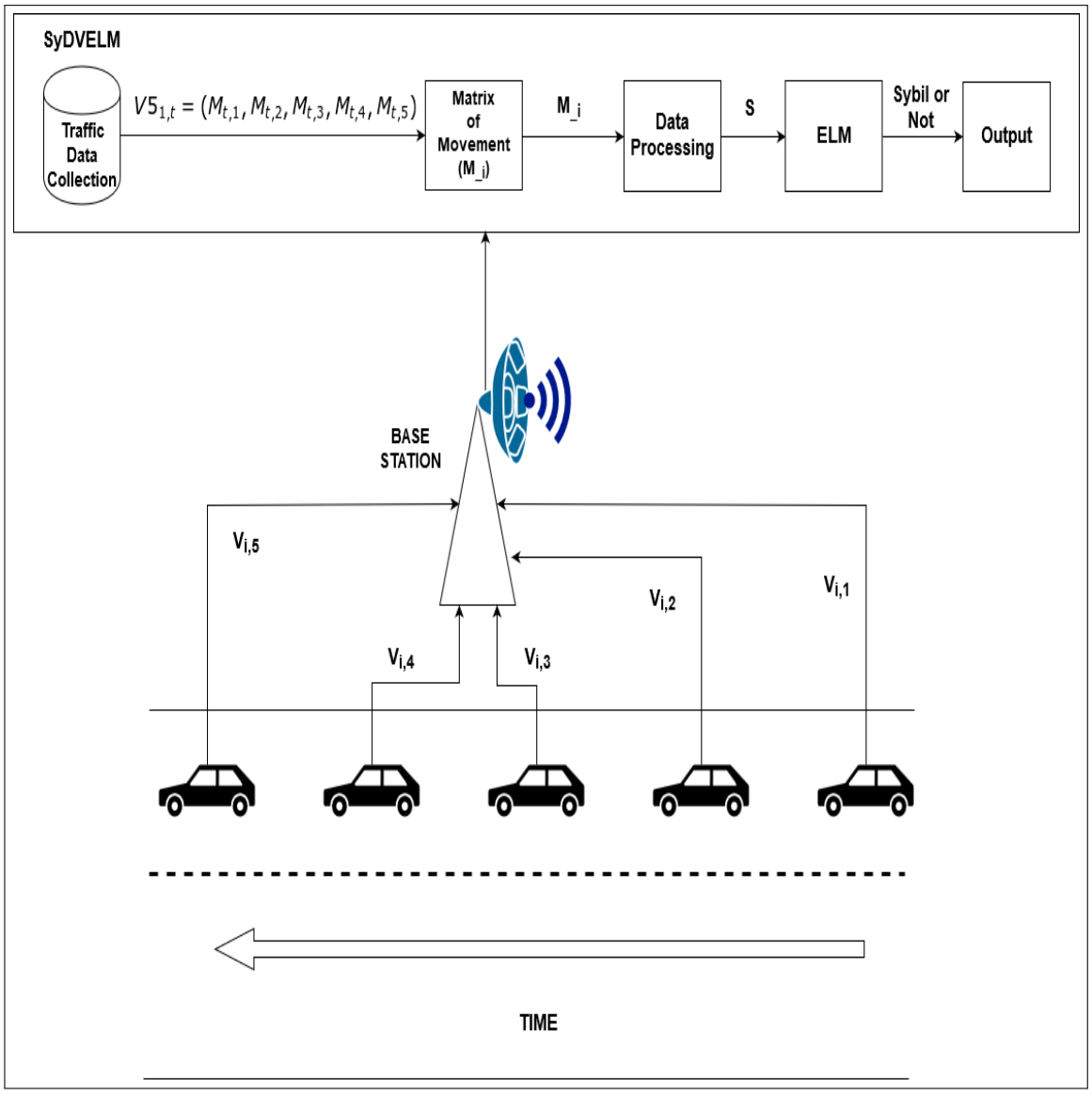
4 PROPOSTA

O Sistema SyDVELM - Sybil Detection in VANETs by Extreme Learning Machine, é um mecanismo de Detecção de Ataque Sybil em VANETs dinamicamente usando técnica de Extreme Learning Machine (ELM) em Classificação Supervisionada.

O método de detecção deve ser implementado em estações base na borda da rede (RSU's, Torres LTE´s - 4G, Torres 5G) pois elas fazem o trabalho de intermediar o mecanismo de identificação/comunicação de veículos, pois os nós veiculares têm que enviar periodicamente seus dados de deslocamento na rede para essas estações, através de mensagens.

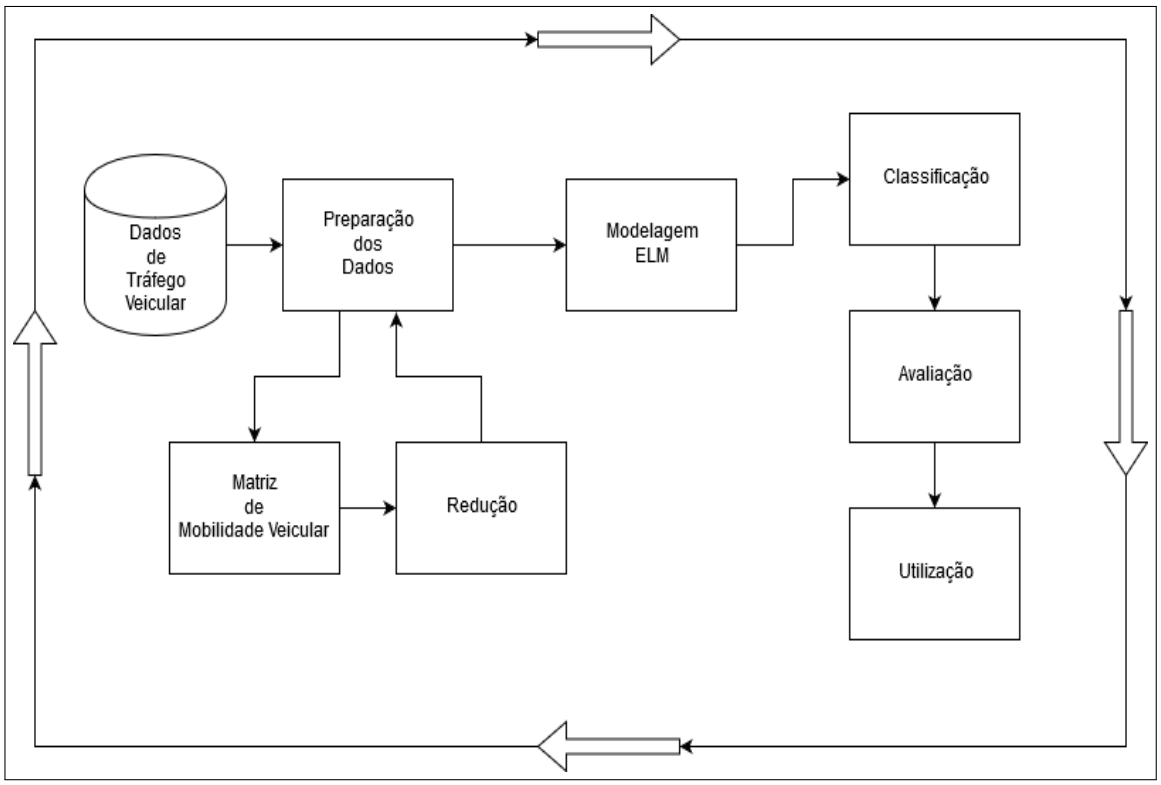
O método é baseado no trabalho de (GU *et al.*, 2016).

Figura 13 – Mecanismo do SyDVELM



Fonte – Elaborado pelo autor.

Figura 14 – Fases de Operação do SyDVELM



Fonte – Elaborado pelo autor.

Este trabalho apresenta as características a seguir:

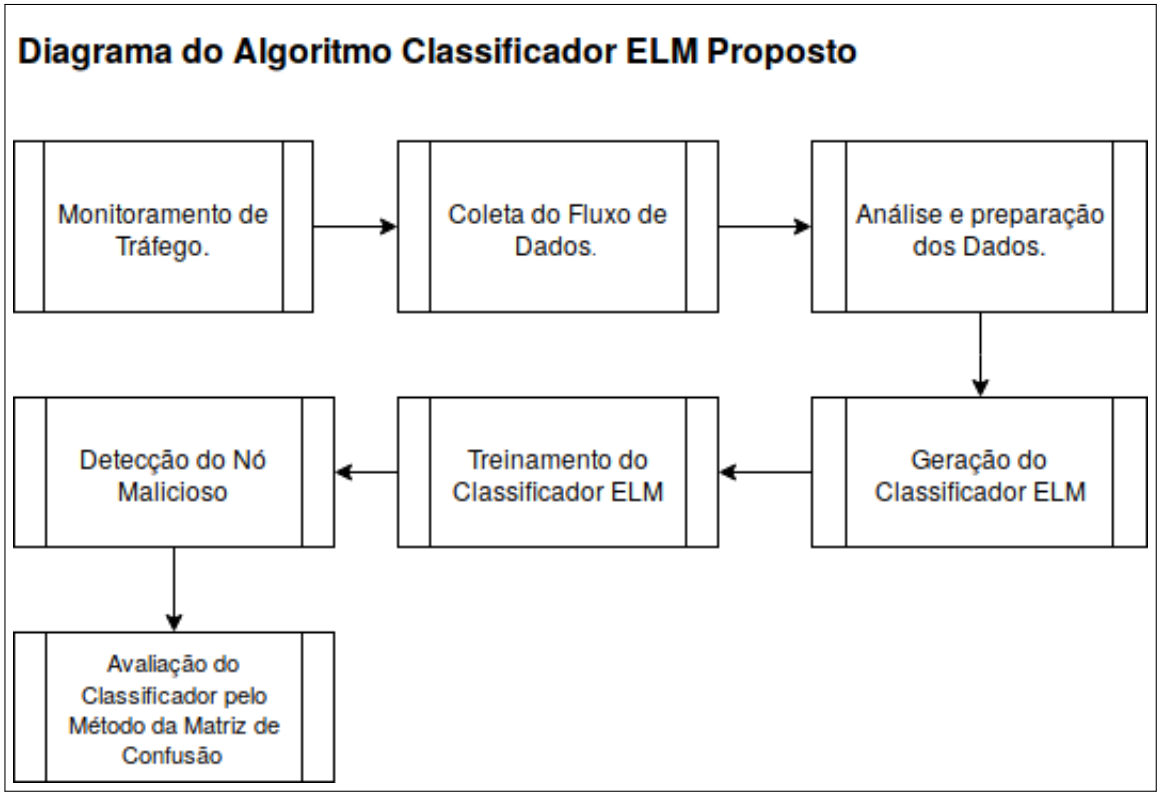
Tomando-se em consideração as Informações de beacon dos veículos enviadas para as unidades de beira de estrada, os dados coletados pelas mesmas serão organizados em uma matriz, que descreverá o padrão de movimentação dos nós veiculares na Vanet durante um determinado intervalo de tempo.

Utilizando uma classificação em ELM, é proposto um método mais eficiente que os baseados em técnicas de Machine Learning usuais(k-NN, SVM, etc.), para detecção de ataques Sybil, comparando os padrões de movimento dos veículos.

A semelhança dos padrões de movimentação dos nós reais, oferece a condição para propor um Classificador ELM mais otimizado em termos de runtime e recursos computacionais.

Nas seções seguintes o Sistema SyDVELM será detalhado.

Figura 15 – Metodologia de Detecção de Ataque Sybil usando técnica ELM



Fonte – Elaborado pelo autor

4.1 SYDVELM

4.1.1 Gerando o Conjunto dos Dados

4.1.1.1 Matriz de Movimentação (MM)

- Descrição do Padrão de Condução do veículo (GU *et al.*, 2017b):
  - Coleta e preparação dos dados que representam o padrão da movimentação dos veículos em uma VANET, num período de tempo específico.
  - O padrão de condução de um veículo em um certo tempo t é descrito pelo uso de um Vetor de 5 elementos.
    - Quando i (Identidade do Veículo), t (Instante do tempo), a Atributos, podemos representar o esse vetor a seguir.
    - $\vec{V_i} = (x_{ta_1}, x_{ta_2}, x_{ta_3}, x_{ta_4}, x_{ta_5})$

Para um dado instante $t_i = 1$.:

$\vec{V_1} = (x_{11}, x_{12}, x_{13}, x_{14}, x_{15})$

1. x_{11} representa horário do veículo no tempo t_1;

2. x_{12} representa localização do veículo no tempo t_1;
3. x_{13} representa velocidade do veículo no tempo time t_1;
4. x_{14} representa aceleração do veículo no tempo t_1;
5. x_{15} representa variação da aceleração do veículo no tempo t_1;

– Os vetores de cada veículo são estruturados em uma matriz que representa o seu deslocamento na rede num intervalo de tempo específico.

– Os vetores de cada veículo em um dado intervalo de tempo (t_1, t_n) são estruturados em uma matriz que irá representar sua movimentação na rede dentro desse intervalo de tempo.

– Assim cada veículo dentro do intervalo de tempo (t_1, t_n) pode ser representado como.:

- Matriz de Movimentação (MM): A = $\begin{vmatrix} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} \\ x_{21} & x_{221} & x_{23} & x_{24} & x_{25} \\ x_{31} & x_{32} & x_{33} & x_{34} & x_{35} \\ : & : & : & : & : \\ . & . & . & . & . \\ x_{n1} & x_{n2} & x_{n3} & x_{n4} & x_{n5} \end{vmatrix}$

- Para cada veículo na VANET é produzida uma Matriz de Movimentação individual.
- Para cada veículo também é calculado a variação da aceleração entre dois instantes consecutivos.
- Após gerar todas as matrizes individuais de cada cenário, o passo seguinte é reduzir a dimensionalidade dos dados através do cálculo dos autovalores de cada matriz (GU *et al.*, 2016).

  – Para isso calculamos para cada veículo a matriz Transposta de sua Matriz de Movimentação.

  – Multiplicamos a Transposta pela própria matriz de forma a gerarmos uma matriz quadrada (5x5). ($M = A^T * A$)

  – Aplicamos o cálculo dos Autovalores na Matriz Quadrada obtida para cada veículo.

  – O cálculo direto dos Autovalores na matriz quadrada gerada, produz Vetores onde surgem números complexos cuja parte imaginária dos mesmos leva a cálculos imprecisos e ineficientes.

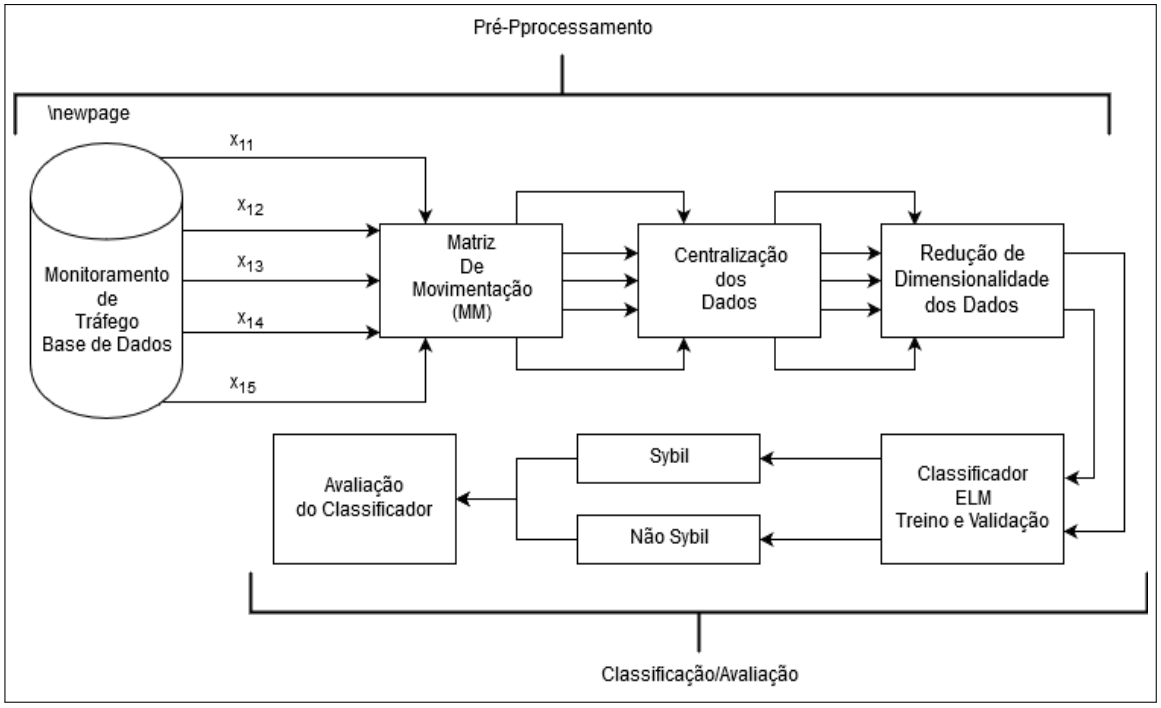
4.1.1.2 Centralização dos Dados

- Para minimizar e contornar a geração de números complexos, aplicou-se o método de centralização dos dados das matrizes de movimentação veicular. Com essa técnica, os Autovalores resultantes são números reais.
- A Centralização dos dados é obtida ao calcularmos a média simples de cada coluna de cada matriz, e em seguida subtraímos essa média de cada elemento de cada coluna, substituindo esse elemento pelo resultado da subtração.
  - $MediaColuna = \frac{\sum_{i=1}^{m} x_{i1}}{m}$, $x_{i1} = (x_{i1} - MediaColuna)$
- Essa operação fez com que os valores das matrizes gerassem Autovalores em números reais, o que possibilitou a Redução da Dimensionalidade dos dados, que eram matrizes (uma para cada veículo), para uma linha de dados (Vetor) para cada veículo.

4.1.1.3 Análise dos Autovalores

- Ao calcular os autovalores para representarem as matrizes dos veículos, foi necessário avaliar se dois autovalores como proposto em (GU *et al.*, 2016), seriam suficientes, ou se seria necessário considerar uma maior quantidade de autovalores.
- Para isso calculamos a energia de dois e de três autovalores através das fórmulas.:
  - Onde E=Energia,AV=Autovalor
    - $E_1 = \frac{(AV_1 + AV_2)}{\sum AutoValores}$
    - $E_2 = \frac{(AV_1 + AV_2 + AV_3)}{\sum AutoValores}$
- Verificamos que a energia dos dois primeiros autovalores já era mais de 90% e desta forma dois autovalores já eram suficientes para representarem cada matriz.

4.1.1.4 Metodologia



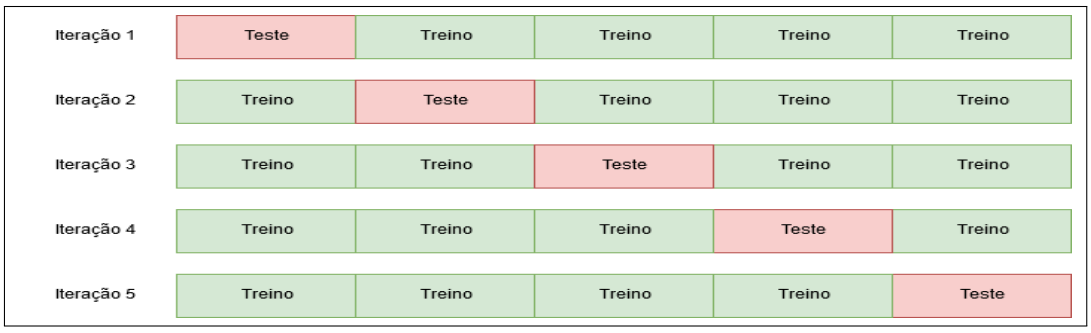
Fonte – Elaborado pelo autor

- Sistema passo a passo
    1. Etapa 1 - Monitoramento de Tráfego
    2. Etapa 2 - Coleta do Fluxo de Dados
    3. Etapa 3 - Análise e Preparação dos Dados
    4. Etapa 4 - Geração do Classificador ELM
    5. Etapa 5 - Treinamento do Classificador ELM
    6. Etapa 6 - Detecção do Nó Malicioso
    7. Etapa 7 - Avaliação do Classificador pelos métodos de Matriz de Confusão, Acurácia e Perda
    8. Etapa 8 - Validação do Classificador pelo métodos de Validação Cruzada

- Etapa 1 - Monitoramento de Tráfego
  - Nessa etapa aplicações de sensoriamento urbano (Sistemas de Gerenciamento de Tráfego - TMS) extraem dados de sensores veiculares e da infraestrutura viária e de comunicação.
  - Os TMS coletam dados do tráfego a partir de fontes heterogêneas, utilizando vários tipos de algoritmos para sumarizar, agregar e fundir esses dados visando a geração de informação útil, e utilizam essa informação para conceber aplicações e serviços com o objetivo de detectar, controlar e reduzir congestionamentos.
- Etapa 2 - Coleta do Fluxo de Dados
  - Nessa etapa são obtidos os dados, que representam os deslocamentos veiculares pela rede viária.
- Etapa 3 - Análise e Preparação dos Dados
  - Esta fase é dedicada ao pré-processamento dos dados, incluindo tarefas de redução, transformação e tudo que for necessário para obter o conjunto de dados a ser utilizado na análise.
- Etapa 4 - Geração do Classificador ELM
  - Nessa etapa a geração dos nós virtuais Sybil bem como o algoritmo de detecção foram implementados. É nesta fase que se estabelece o modelo de solução do problema.
- Etapa 5 - Treino e Validação do Classificador ELM
  - Após termos nossa rede neural montada, ela foi treinada em nosso modelo. Esta é a fase onde são aplicados os algoritmos, as técnicas de aprendizagem e predição mais adequadas ao problema.
- Etapa 6 - Detecção do Nó Malicioso
  - Nessa etapa, validamos os resultados do nosso Modelo com outra massa de dados (dados de teste) não conhecidos previamente pelo modelo, para identificar novos nós agressores Sybil, e assim nos certificarmos de que o algoritmo não ficou viciado e que não acabou aprendendo a prever apenas dados parecidos com aqueles que usou para ser treinado. Só então poderemos colocar nosso algoritmo em produção com confiança de que ele realmente consegue prever dados reais.
- Etapa 7 - Avaliação do Classificador pelos métodos de Matriz de Confusão, Acurácia e Perda
  - Nesta fase os resultados são validados, comparados e interpretados, permitindo verificar se o modelo proposto conseguiu alcançar os objetivos.

- Etapa 8 - Validação do Classificador ELM - Validação Cruzada
  - É uma técnica que visa entender como seu modelo generaliza, ou seja, como ele se comporta quando vai prever um dado que nunca viu. É uma das melhores técnicas para saber se o seu modelo generaliza bem.
  - Usamos uma parcela dos dados para validar nosso algoritmo, isto é, verificamos se ele retorna previsões corretas de Ataques Sybil.
  - A validação cruzada (Cross-Validation - CV) é usada para estimar o erro de teste associado a um modelo para avaliar seu desempenho ou para selecionar o nível apropriado de flexibilidade. A avaliação do desempenho de um modelo é geralmente definida como avaliação de modelo e a seleção de modelo é usada para selecionar o nível de flexibilidade. Esta terminologia é amplamente utilizada no campo da ciência de dados.
  - Na validação cruzada, ou "cross validation", um dos métodos mais utilizados é o K-Fold. Nele são selecionados diferentes conjuntos de dados ("kfolders") que são seleções diferentes do conjunto total de dados. Por exemplo, são selecionados $\frac{1}{10}$ dos dados para validação e o restante para treinamento; depois, outro conjunto de $\frac{1}{10}$ para validação (pode ser uma outra amostra randômica ou um conjunto de dados diferentes) e assim por diante. O resultado (e o erro) é a média dos resultados (e erros) de todas as validações (sessões de treinamento) contendo N-K amostras, sempre variando os conjuntos de amostras: $K = \frac{N}{10}$.
  - A Validação Cruzada é usada para criar diferentes conjuntos de treino e teste, treinar o modelo e ter certeza de que ele está performando bem. Nesse caso, ao invés de usarmos apenas um conjunto de teste para validar nosso modelo, utilizaremos K outros a partir dos mesmos dados.

Figura 16 – Exemplo da Validação Cruzada K-fold



Fonte- Elaborado pelo autor.

4.1.1.5 Métricas de Avaliação

É preciso verificar a Precisão do nosso algoritmo, que indica o quão próximo as classes previstas se aproximam da verdade.

Podemos fazer o mesmo com a perda, que indica o quão distante as classes previstas são da verdade. Ela pode ser considerada o inverso da precisão. A Perda também é chamada Função de Custo. Para saber se meu modelo previu bem a Classe que queremos (Sybil), essas e outras questões podemos entender com as matrizes de confusão, que são tabelas que mostram as frequências de classificação para cada classe do modelo, baseando-se na frequência de acertos e erros da classificação.

Em decorrência da Matriz de Confusão obtemos a acurácia, que diz quanto o meu modelo acertou das previsões possíveis. É a razão entre o somatório das previsões corretas (verdadeiros positivos com verdadeiros negativos) sobre o somatório das previsões. No caso da Perda, a Entropia Cruzada Binária é a função de custo usada em problemas que envolvem decisões binárias do tipo Sim/Não (0/1). Ela mede quão distante do valor verdadeiro (o qual pode ser ambos 0 ou 1) a predição está para cada classe e então calcula a média desses erros de classificação para obter a perda (custo) final. Ela é usada comumente junto com a Função de Ativação Sigmoide.

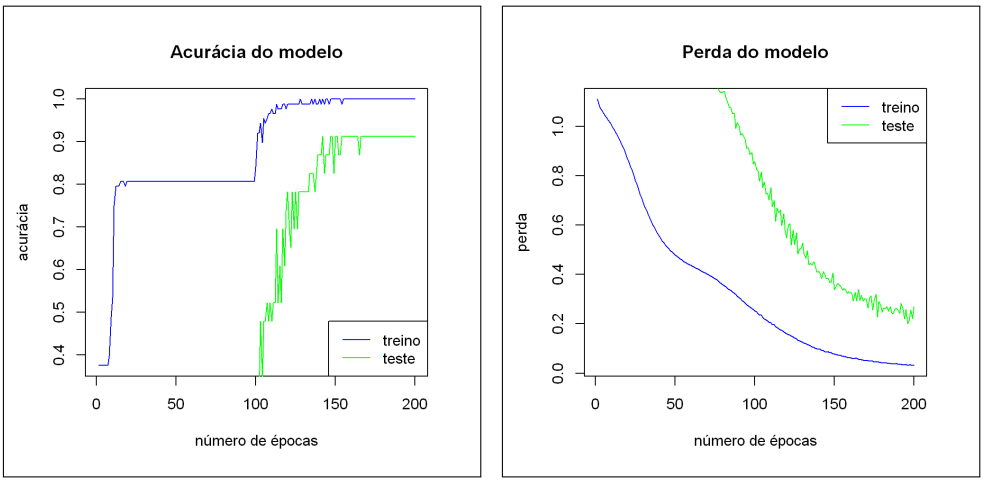
Figura 17 – Exemplo da Matriz de Confusão

1. Verdadeiro Positivo - True positive (TP) é o número de veículos maliciosos identificados corretamente como sendo maliciosos
2. Falso Positivo - False positive (FP) é o número de nós legítimos incorretamente identificados como veículos maliciosos.
3. Falso Negativo - False negative (FN) é o número de nós maliciosos identificados incorretamente como legítimos.
4. Verdadeiro Negativo - True negative (TN) é o número de nós legítimos corretamente classificados como veículos legítimos.

- Acurácia e Perda
  1. Acurácia = $\frac{TP+TN}{TP+FP+TN+FN} = \frac{predies\ corretas}{todas\ as\ predies}$

  2. Perda = Loss (L) = $L(y,\hat{y}) = \frac{1}{N}\sum_{i=0}^{N}(y * log(\hat{y}_i) + (1-y) * log(1-\hat{y}_i))$ onde $\hat{y}$ é o valor previsto esperado e o y é o valor observado.

Figura 18 – Exemplo de Acurácia e Perda



4.2 PREPARANDO OS DADOS

4.2.1 O Ciclo de Treinamento

Podemos visualizar o ciclo de treino de uma rede neural seguindo os passos abaixo:

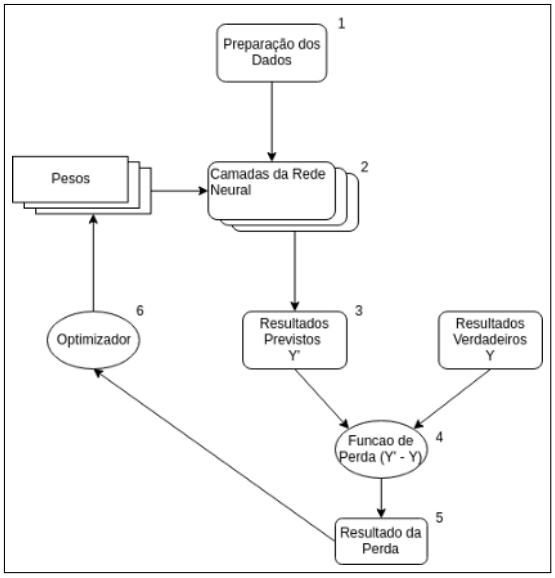
4.2.2 Seleção de características

Nosso primeiro passo será carregar os dados e prepará-los para o consumo. Necessitamos selecionar quais características são relevantes para um algoritmo de aprendizado e excluir as características que não são.

4.2.3 Centralização

A seguir, necessitamos preparar os dados para consumo. Na maioria dos casos, a inserção de dados sem pré-processamento resultará em resultados imprecisos. No caso de redes neurais é importante que as características estejam em um range numérico padronizado. Para centralizar os dados calculamos a média e subtraímos de cada um dos itens essa média. Isto

Figura 19 – Exemplo de Treinamento de uma Rede Neural



Fonte – https://medium.com/ensina-ai/introdução-a-classificadores-binários-usando-keras-3dac9e2a3c6d

possibilita que o algoritmo venha a convergir mais rápido ao resultado.

4.2.4 Separação entre dados de treino e validação

Os dados devem ser separados em duas partes, treino e validação. Dados de treinos serão inseridos no algoritmo de classificação, e dados de teste serão usados para validar a performance de nosso algoritmo.

Nosso próximo passo, será montar nossa rede neural de classificação.

4.2.4.1 A Rede Sequencial

O modelo sequencial permite inserir camadas em série, onde o output da primeira camada serve como input da segunda, e assim por diante.

Nossa camada de entrada receberá uma matriz de dimensão 2, que são as colunas de features selecionadas em nosso dataset. A camada terá 4 neurônios na camada escondida (hidden). O número de neurônios afeta como a rede neural interpreta nossos dados. Mais neurônios permitem mais liberdade em aprender dados mais complexos, porém torna a rede mais cara computacionalmente para ser treinada.

Também devemos dizer qual função de ativação será computada. No nosso caso usaremos a função Sigmoide, que é dada pela fórmula na figura logo a seguir.

model.add(Dense(4, activation='sigmoid', input_dim=2))

Figura 20 – Fórmula da Função Sigmoide

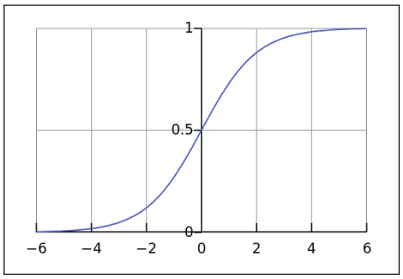
$$s(x) = \frac{1}{1 + e^{-x}}$$

Fontehttps://medium.com/ensina-ai/introdução-a-classificadores-binários-usando-keras-3dac9e2a3c6d

Para classificadores binários, estamos interessados na probabilidade de nosso dado pertencer a uma classe ou outra. Para isso adotaremos a função de ativação da camada de saída a Sigmoide. A função de ativação decide se um neurônio da camada é disparado ou não.

Ou graficamente.:

Figura 21 – Exemplo do Gráfico da Função Sigmoide



Fontehttps://medium.com/ensina-ai/introdução-a-classificadores-binários-usando-keras-3dac9e2a3c6d

Para finalizar, necessitamos de uma camada de saída. A camada de saída deve ter como dimensão de output o número de classes que queremos reconhecer. No nosso caso de classificação binária, temos somente 1 dimensão de saída, true ou false para Sybil.

4.2.5 Compilação

A compilacao serve para validar e finalizar a estrutura de nossa rede neural. Ela recebe 3 parametros:

- Otimizador - Optimizer: Função que define como os pesos da rede neural são atualizados. Escolhemos o ADAM (A METHOD FOR STOCHASTIC OPTIMIZATION), método para otimização estocástica (KINGMA; BA, 2014). Propomos

a adoção desse método que requer apenas gradientes de primeira ordem, com um mínimo de requisição de memória. Esse método computa taxas de aprendizado adaptativas individuais para diferentes parâmetros a partir de estimativas do primeiro e segundo momentos dos gradientes; e o nome Adam é derivado da Estimação de momento adaptativo (adaptive moment estimation). Algumas das vantagens do Adam são que as magnitudes das atualizações de parâmetros são invariantes para redimensionamento do gradiente, suas dimensões são delimitadas aproximadamente pela dimensão do hiper-parâmetro, e não requer um objetivo estacionário, funcionando com gradientes esparsos, e naturalmente executa uma forma de têmpera de delimitação.

- Perda - loss: Função que calcula a diferença entre os dados de teste e os dados de validação. Para classificadores binários, usaremos a binary_crossentropy.
- Métricas - metrics: métricas que devem ser guardadas para avaliação.

4.2.6 Treino e Validação

Agora que temos nossa rede neural montada, iremos treiná-la em nosso dataset. Necessitamos escolher um numero de épocas de treino para nossa rede. Se escolhermos um numero muito baixo de épocas, nosso algoritmo não convergirá para um mínimo global. Se escolhermos um número muito grande, correremos o risco de nosso algoritmo se acostumar demais aos dados de teste, e terá má performance em dados reais

Usaremos uma parcela dos dados para validar nosso algoritmo, isto é, verificar se ele retorna previsões corretas de nós Sybil.

5 RESULTADOS EXPERIMENTAIS

O uso de ferramentas de simulação de redes veiculares possibilita prever o comportamento e desempenho de vários componentes através dos resultados gerados.

A simulação de condições de tráfego são importantes para prever o comportamento de situações atípicas em fluxos de trânsito veicular.

Os simuladores permitem através de aplicações computacionais gerar cenários reais e também facilitam testes de modificações para análise de viabilidade de mudanças.

Um simulador auxilia os pesquisadores a criar vericações técnicas e métricas sem a necessidade de execução de testes em campo. A manipulação de validações pode ser inviável em determinadas situações tais como aquisição e uso de equipamentos com custo elevado e geralmente não se dispõe de tempo para criar uma estrutura eficiente de testes.

Com isso, os simuladores são aliados a nível ferramental e com a função de criar ambientes o mais próximo possível daqueles encontrados na vida real.

O fluxo de tráfego gerado pelo simulador pode utilizar dados reais de trânsito e de mobilidade, os quais podem ser usados como entrada nos simuladores de redes. Assim, é possível calcular e criar componentes em redes wireless com detalhadas estruturas de todos os nós que enviam e recebem pacotes, canais de tráfego e transmissão de sinais.

Nessa etapa serão realizadas simulações de mobilidade veicular, sob o simulador SUMO - Software de Simulação de ambientes de Mobilidade (Tráfego) de Veículos (Simulation of Urban Mobility), pois o mesmo se apresenta de forma adequada para criar fluxos de tráfego em cenários urbanos ou não, onde seus parâmetros se baseiam em mobilidade veicular em tempo real.

A geração dos nós virtuais Sybil bem como do algoritmo de detecção será implementada na Linguagem Python. Utilizando-se da ferramenta Keras, que é uma biblioteca para rede neural de alto-nível escrita em Python e que executa como frontend em TensorFlow ou Theano.

O importante disso é que você pode substituir uma rede neural por outra utilizando o Keras. Ela foi desenvolvida para facilitar experimentações rápidas, isto é, sem que você tenha que dominar cada um dos backgrounds, de maneira rápida e eficiente.

O Keras (Bilioteca Python de Aprendizado Profundo - Deep Learning Library) é uma biblioteca de Redes Neurais, capaz de rodar com o TensorFlow (uma plataforma fim-a-fim que facilita construir e implementar Modelos de Machine Learning desenvolvida pela Google), e que foi projetada pensando em possibilitar uma fácil e rápida prototipação.

Keras é então uma Biblioteca de redes neurais em Python, capaz de rodar em cima das bibliotecas de tensores TensorFlow, CNTK ou Theano. Ela provê uma estrutura que permite compilar redes neurais combinando camadas de diferentes dimensões e funções de ativação, tornando o ciclo de desenvolvimento de novos modelos de aprendizado de máquina muito mais rápido.

As Redes Neurais Multicamadas são aquelas nas quais os neurônios estão estruturados em duas ou mais camadas (layers) de processamento (já que no mínimo haverá 1 layer de entrada e 1 layer de saída).

Os algoritmos de simulação de ataque Sybil, detecção de nós agressores, classificação e treinamento ELM serão implementados utilizando a linguagem de programação Python. Para cada experimento, será criado um arquivo de simulação.

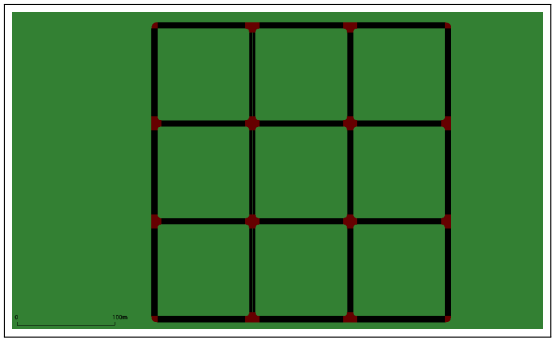
Para analisar o desempenho do algoritmo de detecção baseado em ELM para a identificação de ataque Sybil em VANET, nós simulamos um ambiente de tráfego urbano no SUMO e obtivemos os dados de mobilidade veicular que foram trabalhados via programas em Python e classificados no KERAS. Nessa seção, nós analisamos os resultados produzidos após a classificação.

Nossos resultados experimentais são avaliados usando as métricas de acurácia e perda obtidas através das matrizesde confusão obtidas.

5.1 PARÂMETROS USADOS NA SIMULAÇÃO

Parâmetro	Valor
Simulador de Tráfego	SUMO 1.2.0
Cenário de Simulação	Urbano Manhatan Grid 4X4
Tempo de Simulação	300s
Comprimento das Ruas	1 km
Largura das Ruas	2 Faixas
Velocidade dos Veículos	10 - 70 km/h
Número de Veículos Simulados	300 - 800
Ciclo de Treinamento do Modelo (Época)	300 - 1000
Percentual de Amostras Sybil	1%-5%-10%-15%-20%

Figura 22 – Cenário de Rede Veicular Urbano



Fonte: Elaborado pelo autor.

5.2 EXPERIMENTOS

- Para cada experimento, foi criado um arquivo de simulação.
- Cada ciclo completo de treinamento do Modelo é chamado de época (“epoch“).
- Em nossos experimentos foi necessário escolher um número de épocas de treino para nossa rede. Ao escolhermos um número de 100 épocas, nosso algoritmo não convergiu para um mínimo global. Razão pela qual passamos a variar o número de épocas para 300, 500 e 1000. Pois se escolhermos um número muito grande, correremos o risco de nosso algoritmo se acostumar demais aos dados de teste, e terá má performance em dados reais.
- Durante os experimentos observou-se que de 500 épocas em diante, a convergência para o ótimo global era atingida.
- Realizamos uma análise do Classificador ELM, treinado com as características mais significativas (Autovalores) de suas amostras, consistindo de um universo de 300, 500 e 800 nós legítimos e aplicação de um percentual variando entre 1%, 5%, 10%,15% e 20% de amostras maliciosas (identidades e posicionamentos forjados). Na classificação binária, o comportamento tipo Sybil é uma classe de tipo único.

5.3 RESULTADOS DOS EXPERIMENTOS

Realizamos uma análise do Classificador ELM, treinado com as características mais significativas (Autovalores) de suas amostras, consistindo de um universo de 300, 500 e 800 nós legítimos e aplicação de um percentual variando entre 1%, 5%, 10%,15% e 20% de amostras maliciosas (identidades e posicionamentos forjados). Na classificação binária, o comportamento tipo Sybil é uma classe de tipo único.

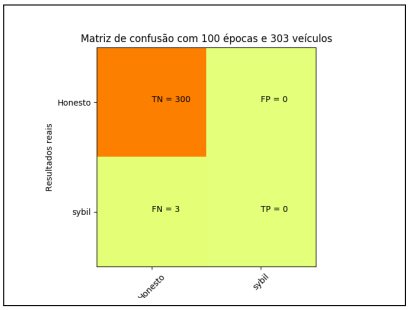
Pode ser observado pelos gráficos que o método de detecção SyDVELM consegue atingir uma alta TPR com uma baixa FPR. Em todos os casos simulados mais de 90% de nós Sybil puderam ser identificados pelo método. Nota-se que quanto maior a quantidade de nós virtuais Sybil, mais rapidamente o modelo de classificação converge para o Ótimo Global de detecção. A Quantidade de Épocas de treinamento usadas também têm correlação direta com essa convergência, pois com 100 épocas em todos os cenários bem como todas as classificações não foram eficientes, e ao se incrementar as Épocas de treinamento do modelo melhora a classificação proporcionalmente.

A Classificação ELM se mostrou mais eficiente pela sua leveza e extrema rapidez de convergência para atingir um resultado ótimo global, onde as propriedades de outros classificadores citados na literatura demonstram resultados não tão otimizados em comparação ao ELM.

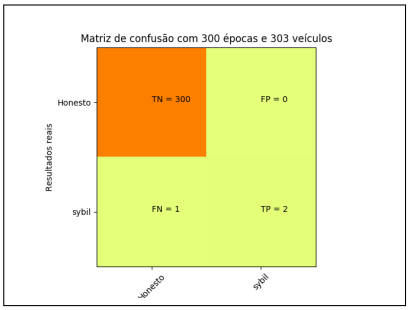
O modelo de Classificação ELM se mostrou estável em todos os cenários a partir de 500 épocas de treinamento, demonstrando a sua versatilidade independente da quantidade de nós veiculares virtuais agressores.

Através dos gráficos de resultados das simulações, fica demonstrado que o método de identificação de ataque Sybil SyDVELM mostra resultados bastante promissores em instâncias legítimas e maliciosas coletadas a partir do processo de simulação e classificação, obtendo baixo consumo de recursos computacionais bem como extrema agilidade em atingir seu objetivo de dar celeridade aos nós legítimos para reagirem em casos de ataques dessa modalidade.

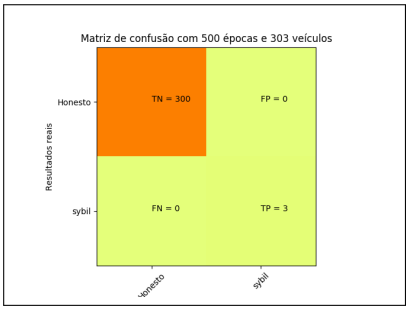
5.4 MATRIZES DE CONFUSÃO OBTIDAS DO MODELO PROPOSTO



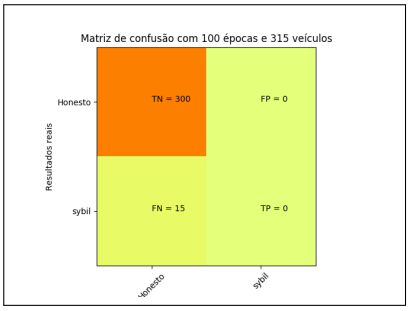
Fonte: Elaborado pelo autor.



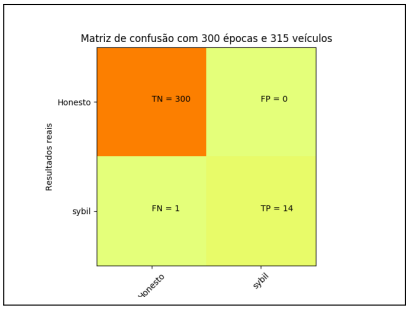
Fonte: Elaborado pelo autor.



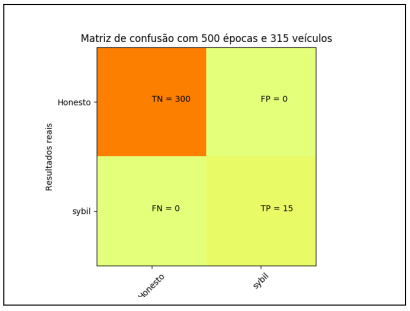
Fonte: Elaborado pelo autor.



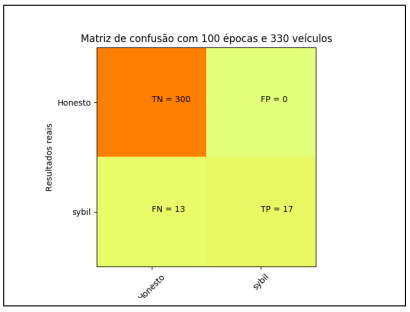
Fonte: Elaborado pelo autor.



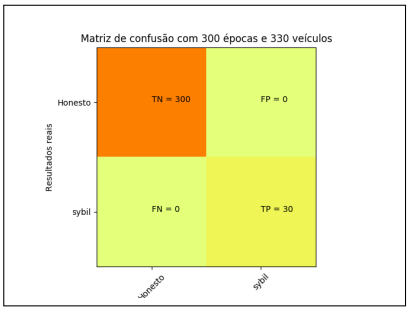
Fonte: Elaborado pelo autor.



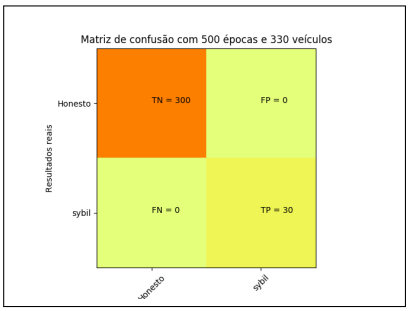
Fonte: Elaborado pelo autor.



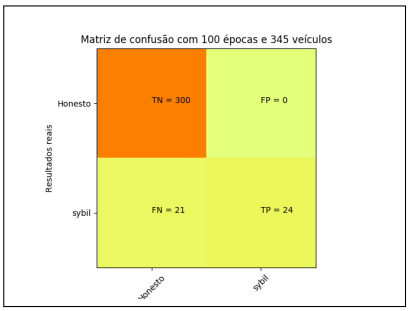
Fonte: Elaborado pelo autor.



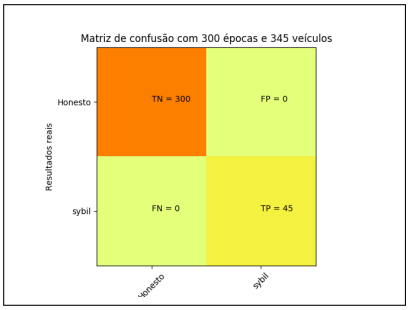
Fonte: Elaborado pelo autor.



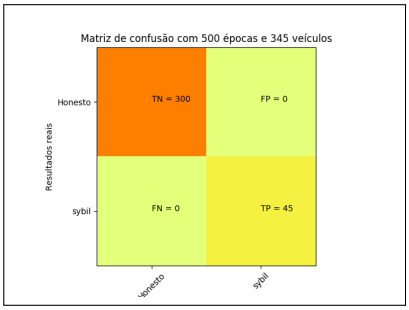
Fonte: Elaborado pelo autor.



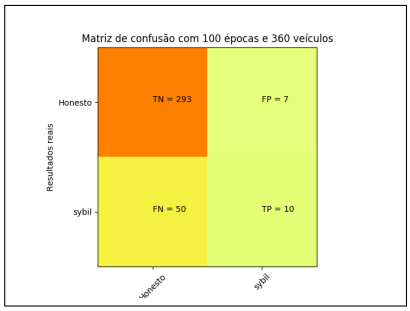
Fonte: Elaborado pelo autor.



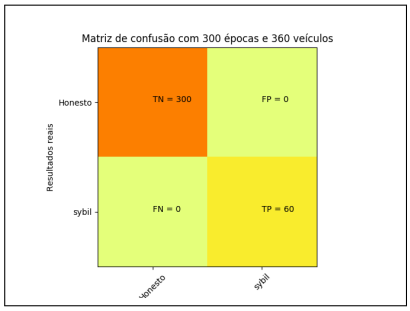
Fonte: Elaborado pelo autor.



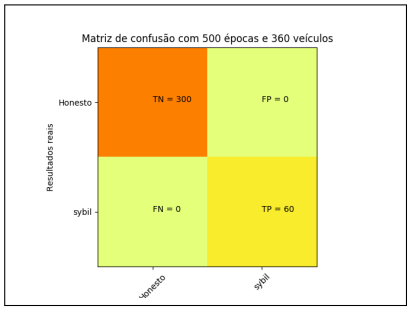
Fonte: Elaborado pelo autor.



Fonte: Elaborado pelo autor.

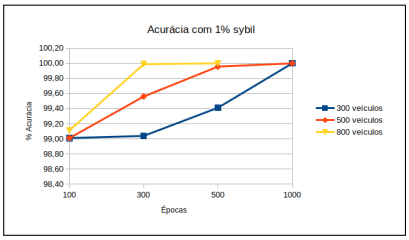


Fonte: Elaborado pelo autor.

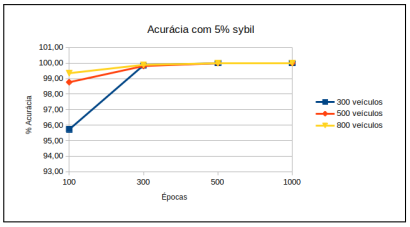


Fonte: Elaborado pelo autor.

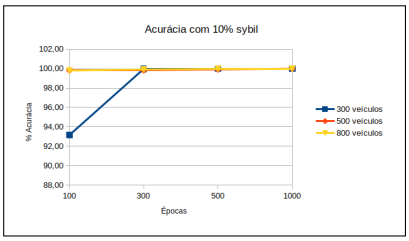
5.5 ACURÁCIA OBTIDA DO MODELO PROPOSTO



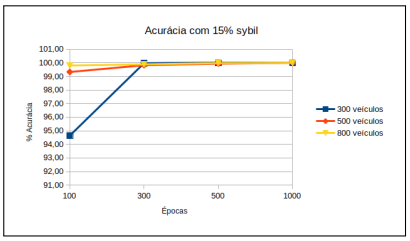
Fonte: Elaborado pelo autor.



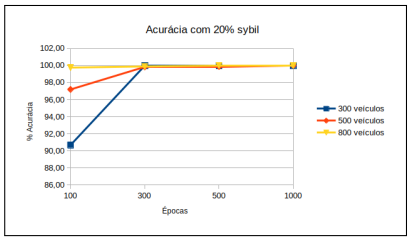
Fonte: Elaborado pelo autor.



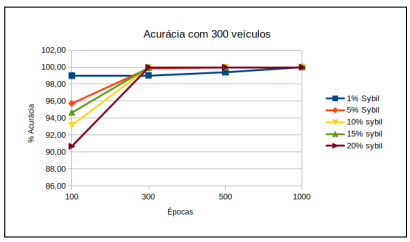
Fonte: Elaborado pelo autor.



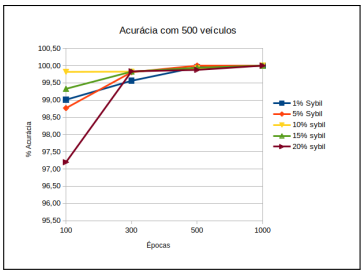
Fonte: Elaborado pelo autor.



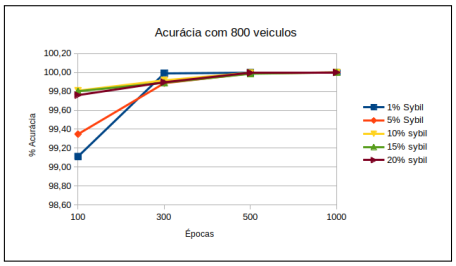
Fonte: Elaborado pelo autor.



Fonte: Elaborado pelo autor.

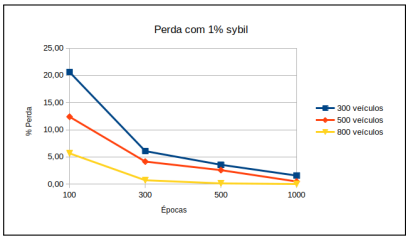


Fonte: Elaborado pelo autor.

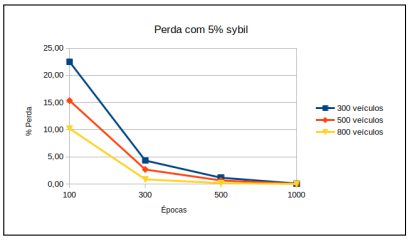


Fonte: Elaborado pelo autor.

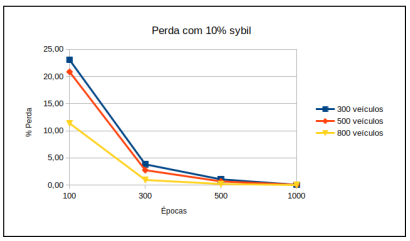
5.6 PERDA OBTIDA DO MODELO PROPOSTO



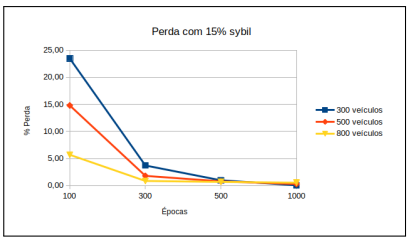
Fonte: Elaborado pelo autor.



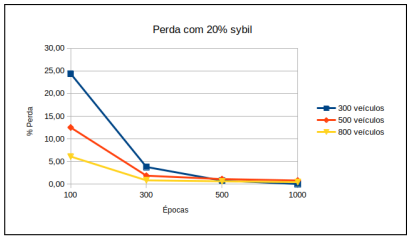
Fonte: Elaborado pelo autor.



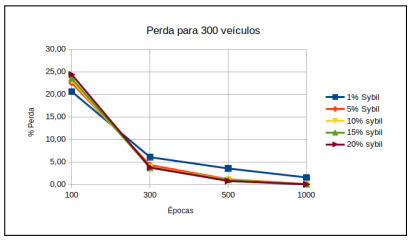
Fonte: Elaborado pelo autor.



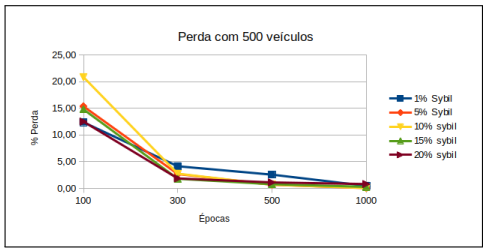
Fonte: Elaborado pelo autor.



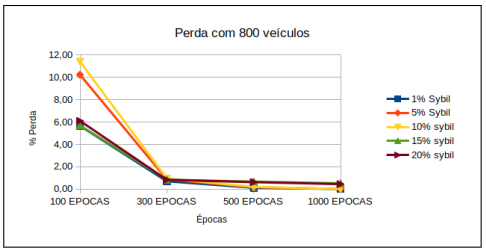
Fonte: Elaborado pelo autor.



Fonte: Elaborado pelo autor.



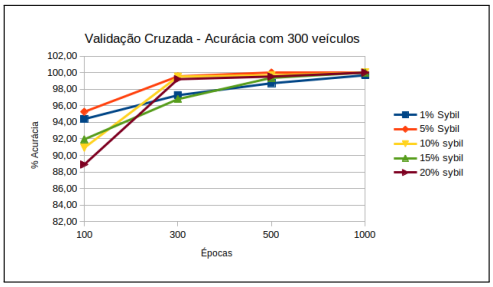
Fonte: Elaborado pelo autor.



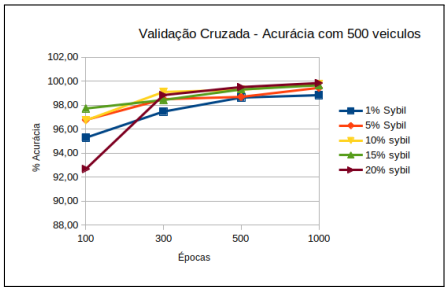
Fonte: Elaborado pelo autor.

5.7 RESULTADOS - VALIDAÇÃO CRUZADA - K-FOLD

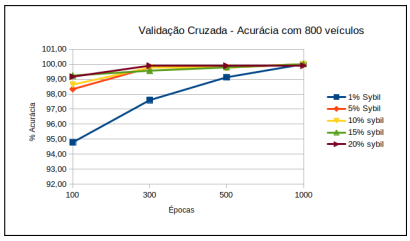
5.7.1 Acurácia - Validação Cruzada - K-Fold



Fonte: Elaborado pelo autor.

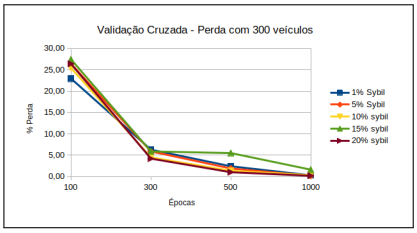


Fonte: Elaborado pelo autor.

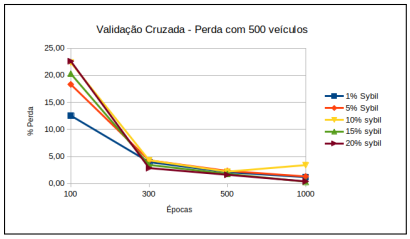


Fonte: Elaborado pelo autor.

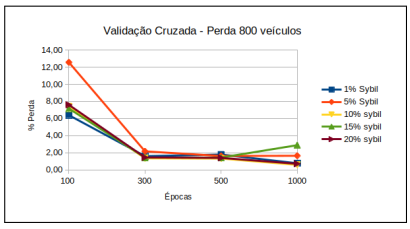
5.7.2 Perda - Validação Cruzada - K-Fold



Fonte: Elaborado pelo autor.

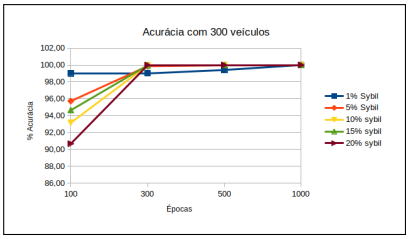


Fonte: Elaborado pelo autor.

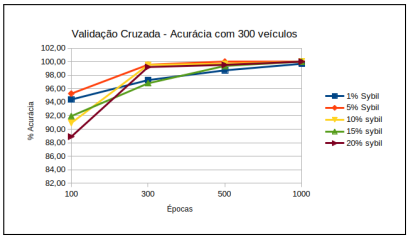


Fonte: Elaborado pelo autor.

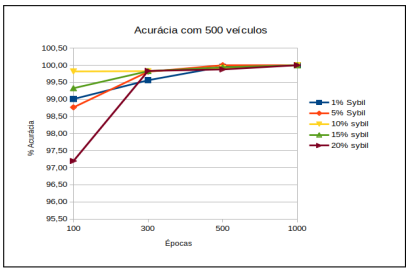
5.7.3 Modelo Proposto x Validação Cruzada - K-Fold - Acurácia



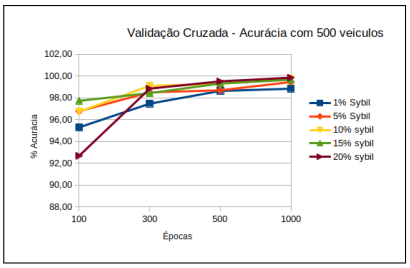
Fonte: Elaborado pelo autor.



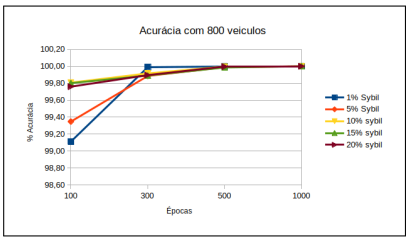
Fonte: Elaborado pelo autor.



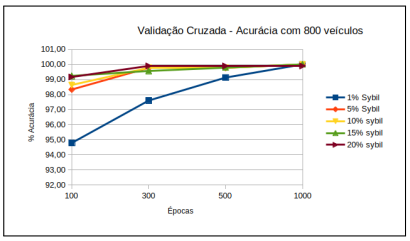
Fonte: Elaborado pelo autor.



Fonte: Elaborado pelo autor.

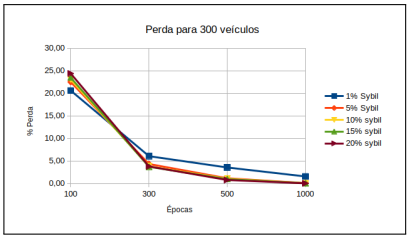


Fonte: Elaborado pelo autor.

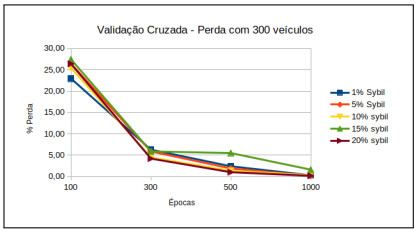


Fonte: Elaborado pelo autor.

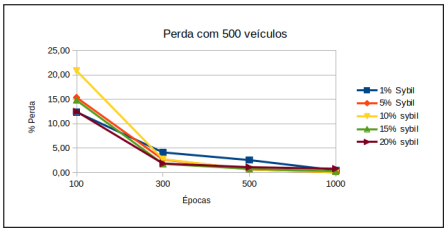
5.7.4 Modelo Proposto x Validação Cruzada - K-Fold - Perda



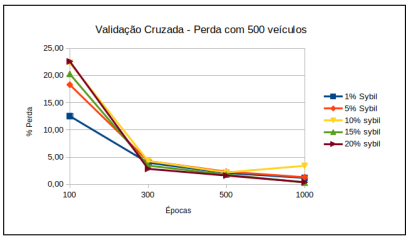
Fonte: Elaborado pelo autor.



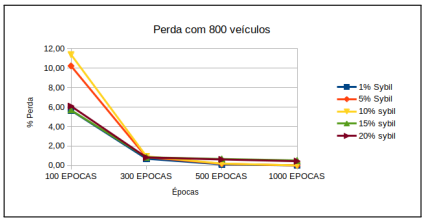
Fonte: Elaborado pelo autor.



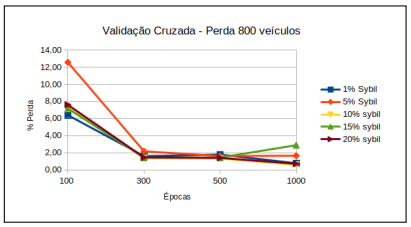
Fonte: Elaborado pelo autor.



Fonte: Elaborado pelo autor.



Fonte: Elaborado pelo autor.



Fonte: Elaborado pelo autor.

5.8 RESULTADOS - ANÁLISE DO TRABALHO PROPOSTO

Este trabalho propôs a utilização da Técnica de Extreme Learning Machine para trazer mais eficiência e acurácia na identificação de Ataques Sybil em VANETs.

No projeto foi introduzido um método de detecção desse tipo de ameaça, baseando-se no padrão de movimentação dos nós veiculares em cenários urbanos, comparando-se a movimentação dos veículos reais em contraste com a inexatidão dos deslocamentos dos nós agressores virtuais.

Nas Simulações dos experimentos os resultados demonstraram que a adoção da Técnica de ELM bem como do tratamento dos dados, obtiveram uma alta taxa de detecção com baixíssimas taxas de erro.

Os principais atributos do presente trabalho são listados na tabela a seguir.

Tabela 2 – Trabalho Proposto

Método	Referência	Vantagem	Desvantagem
ELM Extreme Learning Machine Classificação Binária KERAS	Quevedo et al. (2019)	Alto desempenho de predição, classificação e acurácia. Baixo atraso de detecção. Baixo custo computacional. Reduzido tempo de Treinamento. Rápida convergência para ótimo Global. Função Sigmoide usada na computação do resultado da função não-linear.	O método não foi testado em cenários veiculares esparsos como de áreas rurais.

Fonte – Elaborado pelo autor

6 CONCLUSÃO E TRABALHOS FUTUROS

6.1 CONCLUSÕES

Como demonstrado nessa pesquisa, a detecção de ataques Sybil usando técnica ELM, procurou atingir um tempo computacional pequeno e baixa taxa de erro que fossem condizentes com a dinamicidade peculiar das Redes Veiculares no mundo real, sem ser preciso novos treinamentos a cada detecção e também a capacidade para o tratamento de uma base de dados de tamanho considerável como a obtida das VANETs. A utilização da técnica de ELM mostrou que isso é possível conforme os experimentos mostraram.

Sem dúvida o trabalho de análise e tratamento dos dados foi de suma importância para viabilizar a construção do modelo proposto.

Devemos ressaltar a importância das ferramentas (softwares) utilizadas. Na fase de tratamento dos dados, o diferencial ficou por conta do KERAS (biblioteca de rede neural de código aberto escrita em Python) e da versatilidade da Linguagem Python (linguagem de programação de alto nível, interpretada, de script, imperativa, orientada a objetos, funcional, de tipagem dinâmica e forte) em si mesma que possibilitou a geração das matrizes, dos vetores e cálculos dos Autovalores das mesmas. Tendo tudo isso sido fundamental para a finalização do processo proposto.

Em todo trabalho de pesquisa, os conhecimentos adquiridos são ampliados e novos são adquiridos. São exemplos disso o aprendizados sobre as técnicas de Machine Learning e seus algoritmos.

Os Resultados obtidos foram considerados satisfatórios nas fases do treinamento e testes do processo bem como da validação dos mesmos com o modelo de Classificação de Extreme Learning Machine projetado.

6.1.1 Contribuições

As contribuições do presente trabalho, além dele estar dentro do universo pioneiro do uso da técnica de ELM ao problema de detecção de ataques às VANETs (conforme levantamento efetuado para a realização do mesmo), ao projetar um modelo, baseado em Extreme Learning Machine, aplicado à Detecção de ataques Sybil às Redes Veiculares, isso pode ser descrito da seguinte forma:

1. Representa um instrumento moderno de detecção de ataques do tipo Sybil e, consequente-

mente, de aumento de Segurança nas Estradas;

2. Representa uma ferramenta flexível diante da complexidade das técnicas de Detecção de Ataques atualmente utilizadas, uma vez que as Extreme Learning Machines podem ser treinadas e re-treinadas, a qualquer tempo, aprendendo novas tendências ou padrões de ataques;
3. Representa uma contribuição à sociedade, pois a segurança nas estradas, principalmente das VANETs, tem como conseqüência direta na busca para atingirmos os Sistemas de Transportes Inteligentes.
4. Este trabalho propôs a utilização da Técnica de Extreme Learning Machine para trazer mais eficiência e acurácia na identificação de Ataques Sybil em VANETs.
5. No projeto foi introduzido um método de detecção desse tipo de ameaça, baseando-se no padrão de movimentação dos nós veiculares em cenários urbanos, comparando-se a movimentação dos veículos reais em contraste com a inexatidão dos deslocamentos dos nós agressores virtuais.
6. Nas Simulações dos experimentos os resultados demonstraram que a adoção da Técnica de ELM bem como do tratamento dos dados, obtiveram uma alta taxa de detecção com baixíssimas taxas de erro.

6.2 TRABALHOS FUTUROS

Como parte de trabalhos futuros, outros comportamentos maliciosos de outros tipos de ataques às Redes Veiculares, bem como outros tipos de cenários de mobilidade podem ser implementados.

Também outros ambientes veiculares como Cenários de Highways podem ser desenvolvidos e comparados com o já implementado, produzindo novos resultados para análise.

Outros testes podem acrescentar a proposição de medidas protetivas a serem tomadas quando da detecção de novos ataques às Redes Veiculares aumentando assim a sua Segurança e Confiabilidade, eficiência e robustez.

REFERÊNCIAS

AZEES, M.; VIJAYAKUMAR, P.; DEBORAH, L. J. Comprehensive survey on security services in vehicular ad-hoc networks. **IET Intelligent Transport Systems**, IET, v. 10, n. 6, p. 379–388, 2016.

BARIAH, L.; SHEHADA, D.; SALAHAT, E.; YEUN, C. Y. Recent advances in vanet security: a survey. In: IEEE. **Vehicular Technology Conference (VTC Fall), 2015 IEEE 82nd**. [S.l.], 2015. p. 1–7.

DAS, S.; NENE, M. J. A survey on types of machine learning techniques in intrusion prevention systems. In: IEEE. **2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)**. [S.l.], 2017. p. 2296–2299.

DOUCEUR, J. R. The sybil attack. In: SPRINGER. **International workshop on peer-to-peer systems**. [S.l.], 2002. p. 251–260.

EZIAMA, E.; TEPE, K.; BALADOR, A.; NWIZEGE, K. S.; JAIMES, L. M. Malicious node detection in vehicular ad-hoc network using machine learning and deep learning. In: IEEE. **2018 IEEE Globecom Workshops (GC Wkshps)**. [S.l.], 2018. p. 1–6.

GROVER, J.; PRAJAPATI, N. K.; LAXMI, V.; GAUR, M. S. Machine learning approach for multiple misbehavior detection in vanet. In: SPRINGER. **International Conference on Advances in Computing and Communications**. [S.l.], 2011. p. 644–653.

GU, P.; KHATOUN, R.; BEGRICHE, Y.; SERHROUCHNI, A. Vehicle driving pattern based sybil attack detection. In: IEEE. **High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 2016 IEEE 18th International Conference on**. [S.l.], 2016. p. 1282–1288.

GU, P.; KHATOUN, R.; BEGRICHE, Y.; SERHROUCHNI, A. k-nearest neighbours classification based sybil attack detection in vehicular networks. In: IEEE. **Mobile and Secure Services (MobiSecServ), 2017 Third International Conference on**. [S.l.], 2017. p. 1–6.

GU, P.; KHATOUN, R.; BEGRICHE, Y.; SERHROUCHNI, A. Support vector machine (svm) based sybil attack detection in vehicular networks. In: IEEE. **Wireless Communications and Networking Conference (WCNC), 2017 IEEE**. [S.l.], 2017. p. 1–6.

HAMED, H.; KESHAVARZ-HADDAD, A.; HAGHIGHI, S. G. Sybil attack detection in urban vanets based on rsu support. In: IEEE. **Electrical Engineering (ICEE), Iranian Conference on**. [S.l.], 2018. p. 602–606.

HUANG, G.-B.; WANG, D. H.; LAN, Y. Extreme learning machines: a survey. **International journal of machine learning and cybernetics**, Springer, v. 2, n. 2, p. 107–122, 2011.

HUANG, G.-B.; ZHU, Q.-Y.; SIEW, C.-K. Extreme learning machine: a new learning scheme of feedforward neural networks. In: IEEE. **Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on**. [S.l.], 2004. v. 2, p. 985–990.

HUANG, G.-B.; ZHU, Q.-Y.; SIEW, C.-K. Extreme learning machine: theory and applications. **Neurocomputing**, Elsevier, v. 70, n. 1-3, p. 489–501, 2006.

KAFIL, P.; FATHY, M.; LIGHVAN, M. Z. Modeling sybil attacker behavior in vanets. In: IEEE. **2012 9th International ISC Conference on Information Security and Cryptology**. [S.l.], 2012. p. 162–168.

KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. **arXiv preprint arXiv:1412.6980**, 2014.

LEE, C.-H.; SU, Y.-Y.; LIN, Y.-C.; LEE, S.-J. Machine learning based network intrusion detection. In: IEEE. **Computational Intelligence and Applications (ICCIA), 2017 2nd IEEE International Conference on**. [S.l.], 2017. p. 79–83.

MISHRA, R.; SINGH, A.; KUMAR, R. Vanet security: Issues, challenges and solutions. In: IEEE. **2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)**. [S.l.], 2016. p. 1050–1055.

QU, F.; WU, Z.; WANG, F.-Y.; CHO, W. A security and privacy review of vanets. **IEEE Transactions on Intelligent Transportation Systems**, IEEE, v. 16, n. 6, p. 2985–2996, 2015.

RABIEH, K.; MAHMOUD, M. M.; GUO, T. N.; YOUNIS, M. Cross-layer scheme for detecting large-scale colluding sybil attack in vanets. In: IEEE. **2015 IEEE International Conference on Communications (ICC)**. [S.l.], 2015. p. 7298–7303.

REDDY, D. S.; BAPUJI, V.; GOVARDHAN, A.; SARMA, S. Sybil attack detection technique using session key certificate in vehicular ad hoc networks. In: IEEE. **2017 International Conference on Algorithms, Methodology, Models and Applications in Emerging Technologies (ICAMMAET)**. [S.l.], 2017. p. 1–5.

SHARMA, A. K.; SAROJ, S. K.; CHAUHAN, S. K.; SAINI, S. K. Sybil attack prevention and detection in vehicular ad hoc network. In: IEEE. **2016 International Conference on Computing, Communication and Automation (ICCCA)**. [S.l.], 2016. p. 594–599.

SINGH, P. K.; NANDI, S. K.; NANDI, S. A tutorial survey on vehicular communication state of the art, and future research directions. **Vehicular Communications**, Elsevier, p. 100164, 2019.

TANUJA, K.; SUSHMA, T.; BHARATHI, M.; ARUN, K. A survey on vanet technologies. **International journal of computer applications**, Citeseer, v. 121, n. 18, 2015.

TIWARI, N. On the security of pairing-free certificateless digital signature schemes using ecc. **ICT Express**, Elsevier, v. 1, n. 2, p. 94–95, 2015.