Caracterização e Classificação do Tráfego da *Darknet* com Modelos Baseados em Árvores de Decisão

Mateus Coutinho Marim[1], **Paulo Vitor Barbosa Ramos**[1], **Roberto Massi de Oliveira**[1], **Alex B. Vieira**[1] **e Edelberto Franco Silva**[1]

[1]Programa de Pós-Graduação em Ciência da Computação (PPGCC)
Universidade Federal de Juiz de Fora (UFJF)
Juiz de Fora – MG – Brasil

{mateus.marim,paulo.barbosa,rmassi,alex.borges,edelberto}@ice.ufjf.br

Abstract. *Darknet is a set of networks and technologies, having as fundamental principles anonymity and security. In many cases, they are associated with illicit activities, opening space for malware traffic and attacks to legitimate services. To prevent Darknet misuse is necessary to classify and characterize its existing traffic. In this paper, we characterize and classify the real Darknet traffic available from the* `CIC-Darknet2020` *dataset. Therefore, we performed the feature extraction and grouped the possible subnets with an n-gram approach. Furthermore, we evaluated the relevance of the best features selected by the Recursive Feature Elimination method for the problem. Our results indicate that simple models, like Decision Trees and Random Forests, reach an accuracy above 99% on traffic classification, representing a gain up to 13% in comparison with the state-of-the-art.*

Resumo. *Darknet é um conjunto de redes e tecnologias, tendo como princípios fundamentais o anonimato e a segurança. Em muitos casos, elas são associadas à atividades ilícitas, abrindo espaço para o tráfego de malwares e a ataques a serviços legítimos. Para prevenção do mau uso de uma Darknet, se faz necessária a caracterização e classificação do tráfego nela existente. Neste trabalho, nós caracterizamos e classificamos o tráfego real de uma Darknet disponível pela base* `CIC-Darknet2020`*. Para tanto, realizamos a extração de atributos, e agrupamos possíveis sub-redes com uma abordagem de* n*-gramas. Além disso, avaliamos a relevância dos melhores atributos selecionados pelo método Recursive Feature Elimination para o problema. Nossos resultados indicam que modelos simples, como Decision Trees e Random Forests, alcançam uma acurácia acima de* 99% *na classificação do tráfego, representando um ganho de até 13% em comparação com o estado da arte.*

1. Introdução

A *Internet* apresenta um diversificado nicho segregado pelo nível de disponibilidade e anonimato dos serviços. Aplicações de redes sociais, plataformas de hospedagem e tantas outras ferramentas amplamente divulgadas, são serviços que representam uma ínfima parcela da real rede mundial de computadores. Denominada *Surface Web*, essa reduzida parte é aquela amplamente disponibilizada pelos indexadores de busca e comumente utilizada pelos usuários que procuram serviços e aplicações ditas comuns. A *Deep Web*,

conjunto geralmente "encriptado" e almejado por aqueles que buscam serviços peculiares, disponibiliza os resultados não indexados pelas ferramentas de busca convencionais. Esse conjunto possui seus princípios fundados na contínua alteração de hospedagem e estabelecimento de conexões em pares seguros, *peer-to-peer* (P2P), tendo a *Darknet* como o subconjunto que expande esses princípios e restringe mais a conexão dos pares. [Mirea et al. 2019]

A *Darknet* demonstra o mais alto nível de técnicas de segurança a fim de preservar o anonimato de grupos e prestadores de serviços, preservando a identidade dos sujeitos envolvidos nas relações. Há exemplos de venda de produtos no mercado negro, negociações de serviços e a troca de informações. Embora essas sejam algumas das atividades que estão no lado da ilegalidade, a *Darknet* demonstra uma heterogênea rede estabelecida em princípios fundados na privacidade em meio ao compartilhamento de conteúdos digitais. Tendo essas características, seu objetivo consiste em realizar uma comunicação segura entre os pares, preservando a confidencialidade e integralidade de suas interações, guardando a natureza anônima do compartilhamento. Com isso, a Darknet torna-se um repositório seguro para que qualquer indivíduo possa estabelecer atividades independente de sua natureza, garantindo a vantagem da dificultosa rastreabilidade.

Classificar e categorizar o tipo de aplicação e origem nessas situações de criptografia é um dos objetivos do estudo do problema de *Traffic Classification*. O objetivo de determinar certas classes, usando a análise do tráfego de dados –verificando padrões de duração de conexão, informações sobre origem e destino desses dados, portas conectadas e o tipo de aplicação relacionado ao fluxo analisado– tem utilidade, por exemplo, na detecção de intrusão, gerenciamento de *Quality of Service* em escalabilidade, prevenção à disseminação de *malwares* ou ataques de serviço [Parchekani et al. 2020]. Os métodos que podem ser descobertos na literatura variam, podendo estar vinculados à análise de portas e *payload* ou inserindo inferências estatísticas para classificar o registro em análise [Medeiros et al. 2019].

Mesmo que haja poucos esforços para detectar e caracterizar o tráfego da *Darknet* utilizando aprendizado profundo (*Deep Learning*), tais como *Deep Image Learning* [Gurdip Kaur 2020], há notáveis esforços para identificar o tráfego do mesmo conjunto utilizando-se de técnicas tradicionais de aprendizado de máquinas. Isso demonstra que o estado da arte está em desenvolvimento para sistemas complexos, mas consolidando abordagens mais simples, aprimorando metodologias já existentes para alcançar eficiência dos modelos.

Desta forma, o presente trabalho aborda o problema de classificação de tráfego da *Darknet* utilizando uma base de dados representativa –a *CIC-Darknet2020*– [Gurdip Kaur 2020]. Almejando o melhor desempenho dos modelos abordados de classificação da origem e categorização da aplicação, é feita uma análise dos atributos existentes para a criação de novos campos, expandindo a gama de parâmetros de entrada existentes no *dataset*. Nesse trabalho, também foi realizada uma seleção de atributos com o *Recursive Feature Elimination* para analisar sua relevância em suas respectivas tarefas. Com o subconjunto de características gerado, são ranqueados os mais importantes, a fim de verificar se os novos aparecem entre os mais relevantes. Usando 30 atributos, a validação resultou em uma acurácia de 99,89% para a tarefa de classificação de origem do tráfego. Para a tarefa de categorização da aplicação, com 50 atributos, a acurácia

resultante foi de 98,62%, indicando que é possível remover significativa parte dessas características sem a perda do desempenho geral.

O trabalho está dividido da seguinte forma: a Seção 2 traz alguns trabalhos relacionados, fundamentando a base estudada e o tema abordado. Na Seção 3 é feita uma breve descrição do *dataset* e, logo em seguida, na Seção 4, falamos sobre a correção de rótulos, utilização de modelos de n-gramas para geração de novos atributos e demais manipulações feitas no *dataset*. Na Seção 5 demonstramos os resultados obtidos pela implementação dos modelos de classificação da origem e categorização da aplicação, analisando as relevâncias dos atributos. Por último, realizamos as considerações finais na Seção 6, discutindo interpretações do trabalho, sobre a base de dados e a análise da seleção de atributos feita, sendo deixados sugestões para trabalhos futuros. O código fonte está disponibilizado no *Github*[1].

2. Trabalhos Relacionados

Antes de demonstrar diferentes soluções para o problema de *Traffic Classification*, é necessário entender a definição do problema, que consiste em usar os dados de tráfego entre remetente e destinatário para classificar e categorizar a aplicação usada. Um dos principais desafios é realizar essa tarefa usando dados encriptados, abordagem feita em duas bases disponibilizadas pela *University of New Brunswick*, a *ISCXVPN2016* [Draper-Gil et al. 2016] e *ISCXTor2016* [Lashkari et al. 2017], que, respectivamente, fornecem o tráfego em redes usando VPN (*Virtual Private Network*) e Tor[2].

Recentemente, em um trabalho publicado por [Gurdip Kaur 2020], houve a disponibilização de uma base de dados que é a união das outras duas supracitadas, a chamada *CIC-Darknet2020*. Tal trabalho realiza a classificação das aplicações provenientes da *Sufarce Web* e da *Darknet*, respectivamente, sendo definidas como origem benigna e *Darknet*. Nesse caso, o tráfego *Darknet* era encriptado pelo uso da VPN e Tor. Além da publicação da base de dados, o autor apresentou acurácia de 92% para identificação da origem do tráfego e 86% para a categorização do mesmo em seu modelo de classificação usando redes neurais profundas, utilizando uma técnica chamada de *Deep Image Learning*. Algo importante a se observar é que, apesar da base de dados prover a informação dos IPs de origem e destino do tráfego, não é possível detectar a origem real dele, já que o tráfego que passa pelo Tor faz um caminho por redes intermediárias a fim de esconder a localização real do usuário. Quanto ao tráfego proveniente de VPNs, a detecção da origem não é possível por serem IPs falsos ou *bogons*.

[Draper-Gil et al. 2016] aborda a classificação da comunicação via VPN, usando redes neurais e a base de dados *ISCXVPN2016* para a classificação do tráfego em dois estágios. O primeiro, usando *Multi-Layer Perceptron* como função de ativação para o segundo estágio, uma *Recorrent Neural Network* para identificar as seis classes empregadas pelo *dataset*. Estabelecendo dois cenários de categorização da aplicação, os autores definem modelos de *K-Nearest Neighbors* (KNN) e C4.5 *Decicion Tree* para a tarefa. Eles apresentam um resultado com acurácia acima de 80%, tendo destaque para o C4.5 que teve resultados de medidas de precisão melhores.

[1] `https://github.com/mateus558/Darknet-traffic-classification`
[2] `https://www.torproject.org/`

Em [Lotfollahi et al. 2020], utilizando a mesma base de registro, foi apresentado o desenvolvimento do *framework Deep Packet* para a solução do problema. O *framework* compreende em dois métodos de rede de aprendizado profundo, uma rede neural convolucional e um autoencoder, ambos para a tarefa de classificação e caracterização. Tendo os resultados de acurácia e precisão acima de 90%, o trabalho teve um importante papel em fundamentar outros métodos além daquele que foi empregado para a solução do problema. A classificação utilizando portas, não sugerido pela baixa porcentagem de classificação, a inspeção de *payload* e o uso da abordagem estatística, demonstrando acurácias em torno de 91% para a classificação de protocolos HTTP, POP3 e SMTP e 87% para FTP, IMAP, SSH e TELNET, são sugestões e alternativas de abordagens vistas em [Crotti et al. 2007].

Os trabalhos de [Lotfollahi et al. 2020] e [Draper-Gil et al. 2016], embora tenham contribuído para a categorização da aplicação pelo tráfego, não estão focados nos dados provenientes da *Darknet*. [Gurdip Kaur 2020] aborda a categorização com *deep learning* em duas camadas, a primeira relacionada a classificação da origem e a segunda em relação ao tráfego proveniente da *Darknet*, verificando os atributos do *dataset* mais importantes para a classificação. Mesmo com uma acurácia de 86% para o problema, os autores não realizam qualquer tipo de comparação com modelos de classificação mais simples.

Diferentemente dos trabalhos supracitados, nossa proposta utiliza os modelos *Decision Tree* e *Random Forest* com o objetivo de realizar a comparação entre resultados dos modelos de classificação da origem do tráfego e a categorização da aplicação dos dados provenientes da *Darknet*, além de verificar a influência da criação de novos atributos. Por meio dessa comparação, os modelos utilizados, embora simples, possibilitaram resultados próximos à 100% de acurácia geral, ao contrário dos modelos de *Deep Learning* encontrados no estado da arte. Dessa forma, a base de dados disponibilizada por [Gurdip Kaur 2020] é a escolhida para alcançar os objetivos propostos, expandindo-a com novas características, inserindo informação dos endereços IP de origem e destino, dividindo-os em *n-grams* [Wressnegger et al. 2013], e manipulando os registros originais. Essas alterações e comparações possibilitam a evolução tanto da base quanto do tema em destaque.

3. Conjunto de dados

O conjunto de dados (*dataset*) utilizado para o desenvolvimento de modelos de classificação, *CIC-Darknet2020*, além de incluir registros do tráfego da *Surface Web*, adiciona os dados disponibilizados em ISCXTor2016 e ISCXVPN2016 [Gurdip Kaur 2020]. [Gurdip Kaur 2020] propõe uma base mais completa e disponibiliza dados originários de dois diferentes conjuntos, da *Surface Web*, sendo o tráfego regular de dados, e da *Darknet*, representando os dados anônimos gerados pelo uso de criptografia em múltiplas camadas e tunelamento. É possível observar, pela inspeção dos *timestamps*, que os registros foram obtidos nos períodos de 01/04/2015 até 09/07/2015 e de 23/02/2016 até 25/02/2016.

A base de dados analisada expande o problema de *Traffic Classification*, estabelecendo duas tarefas de classificação. A primeira é a detecção do tráfego da rede como regular (*Benign*) ou como advindo da *Darknet*, tendo a distribuição dos rótulos da detecção descrita na Figura 1. A outra tarefa é referente a caracterização da aplicação do tráfego em oito categorias, cuja distribuição nos dados está representada na Figura 2, sendo elas: *browsing*, *email*, *chat*, *audio-streaming*, *video-streaming*, *File-Transfer*, VOIP e P2P.

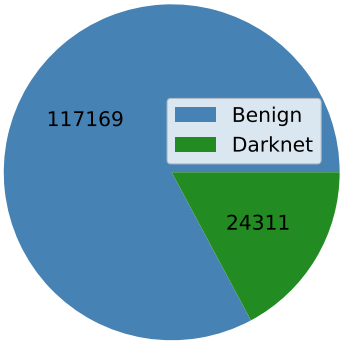


Figura 1. Distribuição pela origem dos dados

O *CIC-Darknet2020* possui uma variedade de campos para análise, trazendo informações sobre IP, porta, duração do tráfego de pacotes e outras medidas relacionadas. No total, são 141.528 registros, dos quais 24.310 provenientes da *Darknet* e 117.218 da rede benigna, diferença de proporção causada pelo fato de os serviços encontrados naquela rede não estarem indexadas pelos motores de busca, além de, necessitar de aplicações especiais, como o Tor, para acessá-la [Gurdip Kaur 2020]. A segunda classe da base de dados diz respeito a aplicação envolvida. A Figura 2 possibilita verificar as aplicações relacionadas à sua origem, sendo possível inferir que classes relacionadas à *Streaming* de áudio e *Chat* são mais comuns para os dados provenientes da *Darknet*, enquanto para a rede benigna são as aplicações minoritárias.

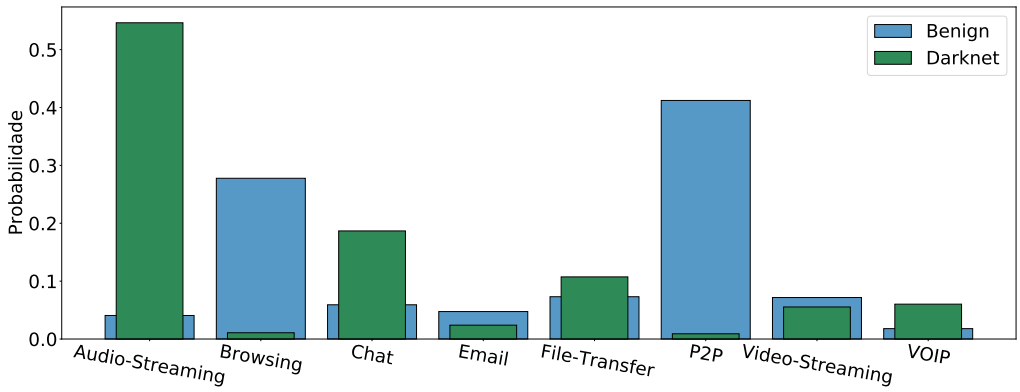


Figura 2. Probabilidade de ocorrência do tráfego de um tipo de serviço

4. Metodologia

Neste trabalho, abordamos as duas tarefas de classificação propostas por [Gurdip Kaur 2020]: a predição da origem do tráfego da rede, com classificações entre *Benign* e *Darknet*, e a caracterização dos serviços do tráfego na *Darknet*. Para estimar o desempenho dos modelos, é utilizada a validação cruzada estratificada onde o *k-fold* tem $k = 10$, esse método se baseia na divisão dos dados em k grupos de tamanho e distribuição dos rótulos aproximadamente iguais, onde o primeiro $fold$ é usado como conjunto de validação e o método é treinado nos $k - 1$ restantes. A acurácia com o *k-fold* é definida como a média de todas combinações de treino e validação. São utilizadas como métricas, em ambas tarefas de classificação, a precisão, o *recall* e o *F-score* [Géron 2019].

4.1. Correção dos rótulos

A primeira correção consiste na normalização dos rótulos contidos nos dois tipos de classes disponibilizadas pela base de dados, a de origem dos dados e do tipo de aplicação do tráfego. Através da inspeção dos rótulos do *dataset* é possível verificar falta de padronização na nomenclatura, assim como a sua redundância. Para corrigir esse problema, realizamos a padronização dos nomes dessas classes pela escolha de apenas um deles entre as duplicatas, como por exemplo, registros com rótulos *AUDIO-STREAMING* foram substituídos por *Audio-Streaming* por terem o mesmo significado.

4.2. Codificação dos atributos

A maioria dos modelos de aprendizagem de máquina trabalham em dados numéricos. Entretanto, muitas vezes encontramos nas bases de dados variáveis categóricas que, como o próprio nome diz, representam categorias ou rótulos. Ao contrário das variáveis numéricas, os valores de uma variável categórica não podem ser ordenados entre eles, ou seja, a sua grandeza não é importante para a tarefa em questão. As categorias de uma variável categórica usualmente não são numéricas, portanto, é necessário um método de codificação para transformar essas categorias em números. Uma das possibilidades é a simples associação das categorias a um inteiro para cada uma das categorias, todavia, os valores resultantes podem se tornar ordenáveis uns com os outros, o que não deveria ser permissível para categorias [Zheng and Casari 2018].

Os atributos correspondentes a endereços de IP presentes na base de dados em questão são exemplos de variáveis categóricas, porque não é importante para o problema o valor de uma sub-rede em relação a outra, já que o valor de uma sub-rede não indica necessariamente a origem do tráfego ou a aplicação que gera esse tráfego. Para abordar o problema da codificação dos IPs com um mapeamento mais genérico é utilizado o conceito de modelos de n-gram. Inicialmente, esses modelos foram propostos para o processamento de linguagem natural e atualmente são as principais representações em muitos sistemas de detecção [Wressnegger et al. 2013].

Uma das possíveis aplicações do modelo de n-grams é a captura direta das sub-redes dos IPs. A RFC 950 [Mogul et al. 1985] define um modelo de 3 níveis para interpretação de endereços da internet, onde o nível mais alto representa a internet como um todo, logo abaixo são as redes individuais e por último são representadas as sub-redes que são úteis para redes pertencentes a organizações moderadamente grandes. Assim, podemos representar cada um desses níveis de interpretação dos endereços IPs com a utilização de modelos de unigramas, bigramas e trigramas, na tentativa de capturar a sub-rede de origem.

Tabela 1. Exemplo da divisão de endereços IP em grams.

IP	Unigrama	Bigrama	Trigrama
172.168.15.14	172	172 168	172 168 15
182.170.224.79	182	170 224	182 170 224

A Tabela 1 exemplifica a divisão dos endereços em grams, possibilitando observar como o processo é similar ao que a máscara de sub-rede faz para identificação da sub-rede, mas, como não temos acesso a máscara, é necessária a criação dos grams para cada nível

de interpretação possível. Em posse desses novos atributos, um modelo de aprendizado de máquina pode ser capaz de aprender os prefixos ou sub-redes mais comuns para um tipo de tráfego como uma informação relevante para a predição de novos exemplos.

A primeira codificação é aplicada sobre os unigramas, bigramas e trigramas criados a partir dos endereços IP. Usamos a técnica de *Hashing Encoding* para criação de 100 novos atributos nomeados com o prefixo $col\_$ seguido pelo ID do atributo gerado pela técnica. Como observado por [Weinberger et al. 2009], o *Hashing Encoding* possibilita a compressão dos atributos em relação ao *One Hot Encoding*. O *One Hot Encoding* faz a codificação dos atributos transformando as categorias do mesmo em novos atributos binários dizendo se aquela categoria pertence ao registro ou não, como mostrado na Figura 3. Isso poderia gerar um número grande de novos atributos dependendo do número de categorias únicas de um atributo processado. Por outro lado, o *Hashing Encoding* tem a desvantagem de que os novos atributos criados perdem no quesito de interpretabilidade, por não ser possível voltar aos valores que originaram os atributos. Além disso, como a pesquisa dos endereços de rede trouxe o país de origem do IP, usamos tal dado como parâmetro de entrada dos modelos, mas convertendo cada país em números ordinais.

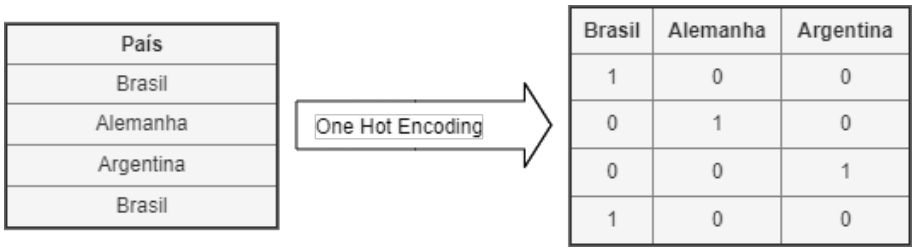


Figura 3. Codificação de um atributo com o *One Hot Encoding*

Uma das transformações mais importantes a serem aplicadas nos dados é o escalonamento dos atributos. Geralmente, poucos algoritmos de aprendizagem de máquina tem um bom desempenho com atributos de escalas muito diferentes. Portanto, se faz necessário colocar os valores dos atributos em um mesmo intervalo numérico. Nesse trabalho fizemos como transformação dos atributos numéricos a padronização dos mesmos através da remoção da média e escalonamento para variância unitária [Zheng and Casari 2018].

4.3. Extração de atributos

A base nos fornece IPs de origem e destino, o que possibilita extrair mais atributos relacionados aos endereços de rede. Uma das possibilidades é o uso de *One Hot Encoder*, entretanto, o uso de *n-grams* pode ajudar na diminuição dos erros do modelo de classificação gerado, consequentemente, reduzindo a percentagem de falsos positivos nas predições. Dessa forma, a base foi expandida usando os IPs, fragmentando-os em *n-grams* (Unigram, Bigram e Trigram). Além disso, foram utilizadas as informações de hospedagem, geolocalização, *bogons* (endereços falsos), entre outros com o auxílio da biblioteca *IpInfo*.

Outra característica extraída foi a hora em que ocorreu a captura dos dados pelo campo *TimeStamp* do *dataset*. A Figura 4 demonstra a relação entre as horas de captura para as duas classes de origem do tráfego usando *TCPDump* e *Wireshark*. É possível verificar dois padrões diferentes no procedimento da criação do *dataset*, um para cada

classe, sendo possível inferir um possível padrão a ser estabelecido pelos modelos e a hora como um relevante atributo para a classificação do tráfego.

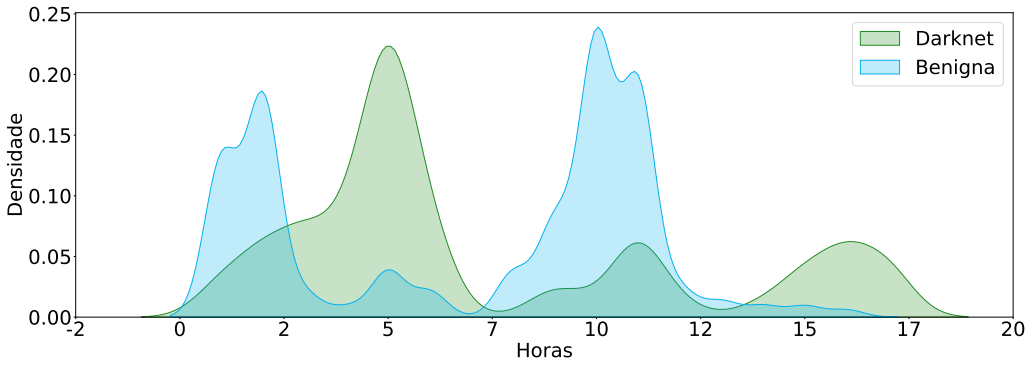


Figura 4. Densidade do tráfego em uma dada hora

Além de ser possível a distinção dos horários, essa relação de tempo permite dizer quando há uma maior probabilidade de utilização de cada rede. Para a rede do tipo *Benign*, constate-se um volume alto do tráfego contido entre às 7 horas e 12 horas, tendo alguns picos durante a madrugada. Para a rede *Darknet*, a distribuição é mais esparsa, não possuindo picos de utilização além da normal, contida entre às 24 horas e 7 horas. Essa relação demonstra uma disjunção exclusiva, ou seja, há uma probabilidade considerável de não haver tráfego da *Darknet* e regular concomitantemente, o que possibilita a construção de padrões bem definidos nos modelos de classificação.

5. Experimentos

Os modelos escolhidos para os experimentos são baseados em árvores de decisão e foram selecionados devido a sua simplicidade e facilidade na interpretação, além de, em conjunto com uma seleção de características, ser possível estimar a importância dos atributos de acordo com a sua influência na classificação. A seguir, são brevemente descritos os modelos escolhidos [Géron 2019].

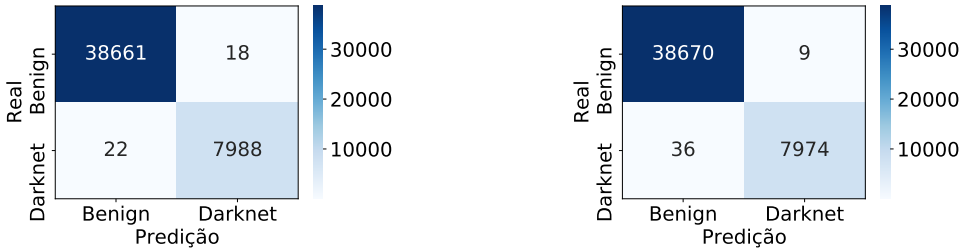
- *Decision Trees* (DT): modelos de aprendizado supervisionado não paramétrico que podem ser usados para classificação e regressão. Funciona pelo aprendizado de regras de decisão simples inferidas dos dados para a predição da variável alvo. São modelos simples de entender e interpretar e as árvores geradas podem ser visualizadas.
- *Random Forest* (RF): é um modelo *ensemble* que usa DTs como classificadores fracos com o objetivo de gerar um classificador forte, a RF treina cada uma das DTs com a técnica de *Bagging* com o objetivo de gerar um classificador com uma performance melhor que a de seus componentes individuais.

No entanto, uma desvantagem dos modelos utilizados é que eles não podem ser treinados de forma *online*, ou seja, não são capazes de aprender com um novo exemplo a não ser que o modelo seja retreinado com todos os dados anteriores e os novos exemplos. Todos experimentos foram feitos com a biblioteca *sklearn* , na linguagem de programação Python, em um computador com processador *Intel Core i5-7200U* com 4 núcleos de $2.5GHz$, $20GB$ de RAM e sistema operacional *Ubuntu* 20.04. Além disso, foi usada a semente

de valor 42 nos algoritmos com alguma aleatoriedade, afim de permitir a reprodutibilidade dos resultados. Cada um dos modelos foi treinado mantendo os parâmetros padrões definidos pelo *sklearn*.

5.1. Detecção do tráfego da *Darknet*

Nas matrizes de confusão das Figuras 5a e 5b, correspondentes aos modelos de *Decision tree* e *Random Forest*, é possível observar a matriz de confusão na tarefa de detecção do tráfego. É possível observar que, para ambas as classes, a grande maioria das amostras são corretamente classificadas. Utilizando o melhor modelo, pelas matrizes de confusão da Figura 5 é possível perceber que para a classe *Darknet*, há um percentual de 0,45% classificados erroneamente como *Benign*, e para o caso contrário, 0,02%.



(a) Decision tree **(b) Random forest**

Figura 5. Matrizes de confusão da detecção de tráfego da *Darknet*

Tabela 2. Sumário das métricas de avaliação dos modelos

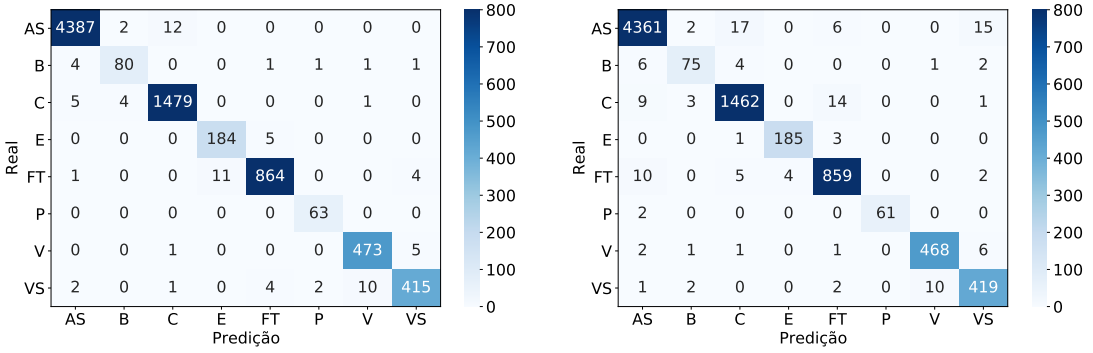
		Precisão	Recall	F-score	10-fold
Decision tree	***Benign***	**0.9994**	0.9992	**0.9993**	99.89%
	Darknet	0.9964	**0.9971**	**0.9967**	
Random forest	***Benign***	0.9989	**0.9997**	**0.9993**	99.90%
	Darknet	**0.9987**	0.9947	**0.9967**	

A Tabela 2 sumariza os valores das métricas de cada modelo na classificação entre os rótulos *Benign* e *Darknet*, tendo os valores em negrito como os melhores resultados obtidos para comparação entre os modelos. Além disso, o modelo conseguiu uma acurácia de 99.89% no *10-fold* e uma boa capacidade de generalização, sendo assim resultados melhores do que a literatura, onde [Gurdip Kaur 2020] conseguiram uma acurácia de 94% na detecção do tráfego da *darknet*.

5.2. Caracterização do tráfego da *Darknet*

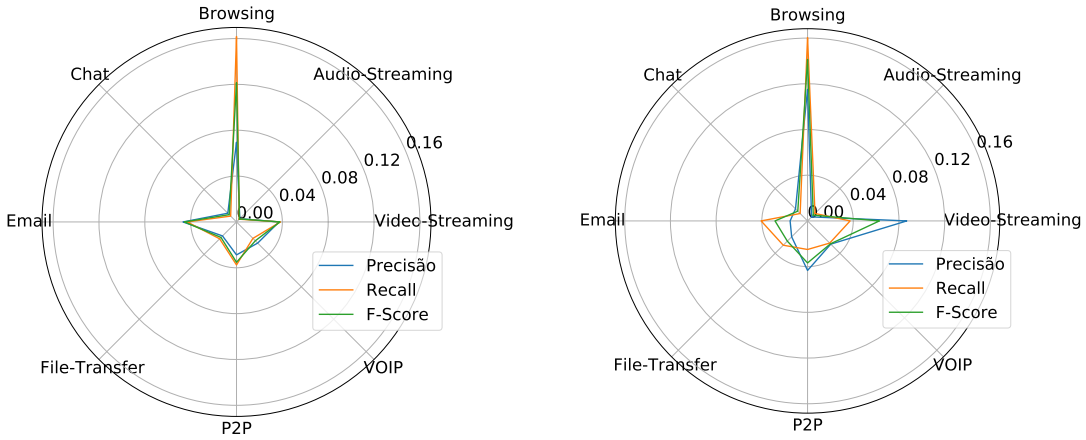
As Figuras 7a e 7b relacionam, em coordenadas polares, os valores de complemento das métricas de precisão, *recall* e *F-score* para os modelos de *Decision Tree* e *Random Forest* respectivamente, para a caracterização do tráfego. Para melhor visualização, foram apresentados complementos dessas métricas. É possível verificar que apenas a classe *Browsing* ficou com uma distância discrepante do valor máximo das métricas. As matrizes de confusão das Figuras 6a e 6b, respectivamente correspondentes a *Decision Tree* e *Random Forest* e as siglas das linhas e colunas correspondem aos rótulos da Tabela 3, fica

evidente que os erros comuns estão relacionados ao tráfego com rótulos de *Chat* e *Audio-Streaming*, podendo indicar que existe alguma similaridade nos rótulos, o que pode causar certa confusão nos modelos.

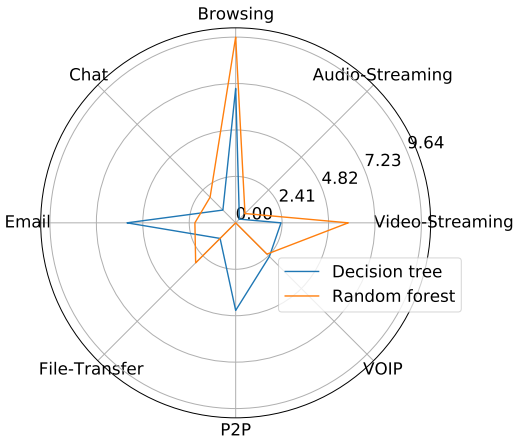


(a) Decision tree **(b) Random forest**

Figura 6. Matrizes de confusão da detecção de tráfego da *Darknet*



(a) Decision tree **(b) Random forest**



(c) Erro no conjunto de teste

Figura 7. Valores das métricas para caracterização do tráfego da *Darknet*

A Figura 7c mostra os erros obtidos pelos modelos na classificação de cada tipo de serviço associado aos tráfegos, ficando evidente que as classes que obtiveram maiores er-

Tabela 3. Rótulos da matriz de confusão

Sigla	Rótulo	Sigla	Rótulo
AS	*Audio Streaming*	**P**	*P2P*
B	*Browsing*	**FT**	*File Transfer*
VS	*Video Streaming*	**C**	*Chat*
V	*VOIP*	**E**	*Email*

ros de classificação são aquelas menos representadas no *dataset*. A *Random forest* foi capaz de atenuar os erros dessas classes, sendo mais adequada para a classificação do tráfego supondo que a adição de novos exemplos siga a mesma distribuição de probabilidade do conjunto de treinamento. Obtivemos um erro de classificação menor, até mesmo nas classes com menor representação, que o modelo proposto por [Gurdip Kaur 2020] com *deep learning* em que obtiveram uma acurácia geral de 86% contra 99.03% da DT, ou seja, conseguimos uma melhora de 13.03%, podendo indicar que o problema da caracterização do tráfego da *Darknet* é melhor resolvido com modelos mais simples e mais fáceis de interpretar sem a necessidade de recorrer para modelos mais complexos e que demandam mais recursos computacionais.

5.3. Comparação dos métodos

Afim de obter uma comparação entre os métodos empregados, a Tabela 4 sumariza o desempenho dos modelos estimados pelo *10-fold* nas tarefas de detecção da origem e caracterização da aplicação e também destaca as melhores acurácias em negrito.

Tabela 4. Comparação dos métodos para detecção e caracterização do tráfego

	Decision Tree	***Random Forest***	**DIDarknet**
Detecção	99,91%	99,90%	94%
Caracterização	99,03%	98,34%	86%

Percebe-se que os modelos de *Decision Tree* e *Randon Forest* apresentam resultados próximos a 100% em ambas tarefas de classificação. Apesar dos resultados parecidos na caracterização, observamos que entre os rótulos menos representados os modelos apresentaram performances diferentes, enquanto a DT teve um erro menor nas classes *Browsing* e *Video-Streaming*, a RF teve melhores resultados na classificação do tráfego de *Email* e *P2P*, portanto, a escolha do modelo a ser utilizado em uma rede depende do tráfego mais provável e também do objetivo que se deseja alcançar. Quanto a tarefa de detecção do tráfego, a DT apresenta menores chances de confundir o tráfego da *Darknet* como regular, o que poderia levar a ocorrência de tráfego malicioso na rede.

5.4. Seleção de atributos

Os algoritmos de seleção de atributos tem como objetivo selecionar, segundo algum critério, um subconjunto a partir do conjunto original de atributos do problema através da remoção de atributos irrelevantes ou redundantes, visando manter os mesmos, ou quase os mesmos, resultados [Villela et al. 2011]. Como os modelos analisados nesse trabalho já tem um desempenho quase ótimo, o objetivo de executar a seleção de atributos

é principalmente reduzir o custo computacional do modelo e analisar os atributos mais importantes para o problema.

Neste trabalho utilizamos o método *Recursive Feature Elimination* (RFE) que funciona através da remoção recursiva de um número fixo de atributos e do retreinamento do modelo. Para avaliar a qualidade dos subconjuntos gerados pelo RFE é feita uma validação cruzada estratificada com *10-fold* e, no final da execução, é selecionado o subconjunto com maior acurácia e menor número de atributos. Devido aos resultados obtidos anteriormente, decidimos utilizar o modelo de *Random Forest* como classificador interno do RFE. Outro motivo para a utilização do RF é que ele também permite saber a importância dos atributos, chamada de importância de Gini, após o treinamento do modelo. Dessa forma, após a seleção de características também fazemos uma análise dos atributos mais importantes considerando os novos atributos inseridos. Como o intuito desse trabalho não é classificar o tráfego em tempo real, não há problema em manter atributos que só podem ser obtidos ao fim do fluxo como o *Flow Duration*.

Tabela 5. Sumário dos resultados da seleção de atributos

	# atributos	Validação final	10-fold
Detecção	28	99.95%	99.91%
Caracterização	73	99.12%	98.94%

Tabela 6. Caracterização

Atributo	Importância
col_91	0.7628
col_49	0.1205
Bwd Init Win Bytes	0.0418
col_24	0.0408
Idle Min	0.0141
col_96	0.0034
Idle Std	0.0021
col_45	0.0018
hour	0.0017
Average Packet Size	0.0015
Flow IAT Std	0.0011
col_1	0.0009
Idle Mean	0.0009
Flow IAT Min	0.0008
Fwd Packet Length Max	0.0008
FIN Flag Count	0.0007
Src Port	0.0006
FWD Init Win Bytes	0.0006
Fwd IAT Total	0.0005
Flow Duration	0.0004
col_71	0.0004
Fwd Packets/s	0.0004

Tabela 7. Detecção do tráfego

Atributo	Importância
col_76	0.4287
hour	0.1455
Bwd Packet Length Min	0.1262
Idle Max	0.0517
Fwd Header Length	0.0338
Idle Min	0.0335
col_58	0.0312
Packet Length Max	0.0245
Flow Duration	0.0158
col_75	0.0142
col_11	0.0128
col_21	0.0112
col_45	0.0107
Src Port	0.0083
Dst Port	0.0078
Flow IAT Max	0.0053
Fwd Seg Size Min	0.0042
Flow IAT Min	0.0039
col_91	0.0037
FWD Init Win Bytes	0.0029
Subflow Fwd Bytes	0.0029
Fwd IAT Max	0.0026

Devido a natureza aleatória do RFE, cada execução pode gerar um subconjunto de atributos diferentes como resultado, assim, o subconjunto resultante deve ser considerado como uma aproximação do subconjunto ótimo de atributos. Na Tabela 5 resumimos os resultados, como os atributos foram selecionados tendo como base o *10-fold*, também foi separado um conjunto de validação com 33% das amostras com uma amostragem aleatória estratificada, podemos ver que houve uma redução significativa para o número de atributos do *dataset* sem que houvesse perda na acurácia da classificação em ambas tarefas. Na primeira obtivemos uma redução de 83% do total de atributos e na caracterização do tráfego o número de atributos foi reduzido em 72%.

As Tabelas 6 e 7 mostram os 22 atributos mais importantes nos conjuntos selecionados pelo RFE em ambas tarefas de classificação. Fica evidente que em ambos conjuntos os atributos inseridos pelo pré-processamento dos dados estão nos primeiros lugares em relação a sua importância para a classificação do modelo. Isso indica que os atributos inseridos são relevantes e que é mais vantajoso fazer o processamento dos atributos do que removê-los quando não parecem ser relevantes, como feito em [Gurdip Kaur 2020].

6. Conclusão

Neste trabalho abordamos os problemas da detecção e caracterização do tráfego proveniente da *Darknet* através da utilização de modelos de aprendizagem baseados em árvores de decisão, sendo eles a DT e a RF, que se mostraram capazes de classificar novos registros de tráfego com uma acurácia superior a 98% para cada uma das tarefas de classificação.

Também foram extraídos novos atributos do *dataset* original pela busca de informações dos IPs de origem e destino do tráfego e pela codificação dos mesmos com o *hashing encoding*. Outro atributo gerado foi o horário em que o tráfego ocorreu pelo *timestamp* incluído no *dataset* que, pelas nossas análises iniciais, mostraram potencial para contribuir na eficiência dos modelos treinados devido a tendência de ocorrência dos tráfegos, da internet comum e da *Darknet*, em horários distintos. Por fim, também fizemos uma seleção de atributos com o RFE e verificamos que os novos atributos inseridos tiveram relevância para a predição dos modelos ficando evidente que, em alguns casos, é preferível o processamento de atributos que a primeira vista tem pouca relevância. Além disso, foi possível obter uma grande redução do número de atributos do *dataset* original.

Fica evidente que algoritmos de aprendizagem de máquina simples, como os baseados em árvore decisória, são bons candidatos para obtenção de resultados competitivos para problemas do mundo real. Nesse trabalho, observamos que a DT e a RF obtiveram um resultado até 13% maior que o modelo de [Gurdip Kaur 2020], além de sua eficiência poder ser melhorada com um pré-processamento cuidadoso dos atributos já existentes. Uma proposta de trabalho futuro é a utilização de modelos de aprendizado *online*.

7. Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Referências

Crotti, M., Dusi, M., Gringoli, F., and Salgarelli, L. (2007). Traffic classification through simple statistical fingerprinting. *ACM SIGCOMM Computer Communication Review*, 37(1):5–16.

Draper-Gil, G., Lashkari, A. H., Mamun, M. S. I., and Ghorbani, A. A. (2016). Characterization of encrypted and vpn traffic using time-related. In *Proc. of the Int. conference on information systems security and privacy (ICISSP)*, pages 407–414.

Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media.

Gurdip Kaur, Arash Habibi Lashkari, A. R. (2020). aDIDarknet: A Contemporary Approach to Detect and Characterize the Darknet Traffic using Deep Image Learning. In *10th International Conference on Communication and Network Security (ICCNS 2020)*.

Lashkari, A. H., Draper-Gil, G., Mamun, M. S. I., and Ghorbani, A. A. (2017). Characterization of tor traffic using time based features. In *Proc. of the Int. conference on information systems security and privacy (ICISSP)*, pages 253–262.

Lotfollahi, M., Siavoshani, M. J., Zade, R. S. H., and Saberian, M. (2020). Deep packet: A novel approach for encrypted traffic classification using deep learning. *Soft Computing*, 24(3):1999–2012.

Medeiros, D., Cunha Neto, H., Andreoni Lopez, M., Magalhaes, L., Silva, E., Vieira, A., Fernandes, N., and Mattos, D. (2019). Análise de dados em redes sem fio de grande porte: Processamento em fluxo em tempo real, tendências e desafios. *Minicursos do Simpósio Brasileiro de Redes de Computadores-SBRC*, 2019:142–195.

Mirea, M., Wang, V., and Jung, J. (2019). The not so dark side of the darknet: a qualitative study. *Security Journal*, 32(2):102–118.

Mogul, J. et al. (1985). Internet standard subnetting procedure.

Parchekani, A., Naghadeh, S. N., and Shah-Mansouri, V. (2020). Classification of traffic using neural networks by rejecting: a novel approach in classifying vpn traffic. *arXiv preprint arXiv:2001.03665*.

Villela, S. M., Xavier, A. E., and Neto, R. F. (2011). Seleção de características com busca ordenada e classificadores de larga margem. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação.

Weinberger, K., Dasgupta, A., Langford, J., Smola, A., and Attenberg, J. (2009). Feature hashing for large scale multitask learning. In *Proc. of the 26th annual international conference on machine learning*, pages 1113–1120.

Wressnegger, C., Schwenk, G., Arp, D., and Rieck, K. (2013). A close look on n-grams in intrusion detection: anomaly detection vs. classification. In *Proc. of the 2013 ACM workshop on Artificial Intelligence and Security*, pages 67–76.

Zheng, A. and Casari, A. (2018). *Feature engineering for machine learning: principles and techniques for data scientists*. "O'Reilly Media, Inc.".