Abordagem Fuzzy Valorada Intervalarmente para Classificação de Tráfego de Streaming de Vídeo

Eduardo Maroñas Monks[1], Bruno Moura[1], Guilherme Bayer Schneider[1], Adenauer Correa Yamin[1], Renata Hax Sander Reiser[1], Helida Santos[2]

[1]Universidade Federal de Pelotas (UFPEL) - Brasil
Laboratory of Ubiquitous and Parallel Systems (LUPS/CDTEC)

[2]Universidade Federal do Rio Grande (FURG) - Brasil
C3 - Centro de Ciências Computacionais
Universidad Publica de Navarra (UPNA) - Spain
ISC - Institute of Smart Cities
Email: {emmonks, bmpdmoura, gbschneider, adenauer, reiser}@inf.ufpel.edu.br, helida@furg.br

Abstract. *This paper contributes to classifying video streaming traffic exploring concepts of Interval-Valued Fuzzy Logic. This approach extends related work by considering the uncertainties generated by variations in network conditions and the parameter imprecision affecting the behavior of network flow, which increases the complexity to reach high accuracy on identifying the network traffic. Some evaluations using the interval-valued logic approach for video streaming traffic classification are presented using applications and datasets to validate the proposal.*

Resumo. *Este artigo contribui para a classificação do tráfego de streaming de vídeo explorando conceitos de Lógica Fuzzy Intervalar. Essa abordagem estende os trabalhos relacionados ao considerar as incertezas geradas pelas variações nas condições da rede e a imprecisão dos parâmetros que afetam o comportamento do fluxo da rede, o que aumenta a complexidade para alcançar maior acurácia na identificação do tráfego da rede. Algumas avaliações usando a abordagem de lógica intervalar para classificação de tráfego de streaming de vídeo são apresentadas com o uso de aplicações e datasets para validar a proposta.*

1. Introdução

A classificação do tráfego em redes de computadores (CTR) é um processo fundamental para diversas áreas de pesquisa referentes às infraestruturas computacionais distribuídas, contribuindo para incremento da segurança, qualidade de serviços e contabilização de recursos tecnológicos [Bujlow et al. 2015] .

Na CTR, muitos fatores podem implicar incertezas e imprecisões. Destacam-se a imprevisibilidade da ocorrência de problemas na mídia, as flutuações nos canais de comunicação e os recursos computacionais utilizados para manter a convergência da rede, assim como a sobrecarga nos canais de comunicação, os erros de configuração e os desastres naturais. Da mesma forma, essas incertezas e imprecisões decorrem de métricas aplicadas sobre sensores nos recursos da rede, medindo perturbações e diferentes variações.

Atualmente, cerca de 70% do tráfego de rede na Internet é composto por vídeo *streaming* [Sandvine 2020]. Esse tipo de tráfego possui características específicas, apesar de compartilhar os mesmos protocolos utilizados para outros tipos de serviços, como navegação ou download de arquivos.

Sendo assim, aumenta o interesse por novos métodos e tecnologias em CTR, incentivando desafios na área na perspectiva de colaborar na classificação de vídeos de forma rápida e eficiente. Assim, destacam-se as seguintes questões de pesquisa:

a) Os classificadores de tráfego de rede consideram incertezas relacionadas a características de fluxos tais como *Packet Length Mean* (PLM), *Fwd Packet Length Std* (PLS) e *Backward InterArrival Time Total* (BIAT)?
b) Os classificadores de tráfego de rede estão preparados com técnicas consistentes para lidar com incertezas associadas a ambientes de rede de computadores?
c) Os classificadores de tráfego de rede estão munidos de técnicas consistentes para lidar com tráfego criptografado?
d) Os gerentes de redes dispõem de novas tecnologias para incremento no conhecimento, preciso e confiável, do volume de tráfego de *streaming* de vídeo?

As questões reportadas acima e trabalhos como [Al-Obeidat and El-Alfy 2019, Shifa et al. 2020], que utilizam a Lógica Fuzzy para modelar incertezas no processo CTR, nos motivaram a definir a abordagem denominada *FuzzyNetClass*, explorando Lógica Fuzzy Valorada Intervalarmente e visando classificação de tráfego de rede, com foco na classificação de *streaming* de vídeo.

O artigo está estruturado da seguinte forma. A primeira seção trata dos fundamentos contextuais do trabalho. Trabalhos relacionados são apresentados na Seção 2. A Seção 3 introduz conceitos básicos de lógica fuzzy tipo 2 (T2FL). A Seção 4 trata dos fundamentos conceituais do CTR. Nas Seções 5 e 6 é apresentada a modelagem do componente *FuzzyNetClass*. Seção 7 descreve a avaliação experimental. Finalmente, a Seção 8 apresenta as conclusões e trabalhos futuros.

2. Trabalhos Relacionados

A revisão de literatura realizada identificou nove trabalhos relacionados que utilizam lógica fuzzy na CTR, e são brevemente relatados nesta seção, suas principais características estão resumidas na Tabela 1.

Em [Asmuss and Lauks 2015], foram desenvolvidos métodos de classificação de tráfego de rede e detecção de anomalias baseados na análise de séries temporais de tráfego, usando a técnica de agrupamento fuzzy.

Em [Shalaginov and Franke 2015], os autores descreveram o estudo em andamento e os primeiros resultados sobre a aplicação do modelo neuro-fuzzy para apoiar a investigação forense de tráfego em larga escala. Já no artigo de [Qader et al. 2017], três diferentes algoritmos de mineração de dados foram discutidos como parte da solução proposta para classificação de falhas de rede: K-Means, Fuzzy C-Means e Expectation Maximization. Em [Ducange et al. 2017], reporta-se uma abordagem para tratar o problema de classificação de tráfego usando classificadores fuzzy evolutivos multiobjetivos.

Na pesquisa de [Abdullah and Al-Hashmi 2018], foi proposto um sistema fuzzy evolutivo para discriminar anomalias inspecionando o tráfego da rede. Os resultados com-

provaram a adequação do método fuzzy evolutivo de séries temporais para classificação em redes. Na pesquisa de [Al-Obeidat and El-Alfy 2019], foi concebida uma abordagem de aprendizado de máquina híbrida supervisionada, para classificação de tráfego de rede, baseado em árvores de decisão fuzzy multicritério.

Em [Iglesias et al. 2019], foi apresentada uma abordagem para classificação de ataques em rede baseada em árvores de decisão lineares simples, e árvores de decisão fuzzy. Na abordagem de [Shifa et al. 2020] foi proposto um modelo denominado Fuzzy Logic Threat Classification, como base de um método para detectar automaticamente três níveis diferentes de confidencialidade para vídeos transmitidos de dispositivos móveis por meio de servidores de borda.

Na perspectiva de [Parfenov et al. 2020], a pesquisa apresentada teve como objetivo desenvolver um sistema de inferência fuzzy para classificar o tráfego de rede anormal e identificar os ataques atuais por tipo, extraindo regras para o sistema de inferência através de método de árvore de decisão.

Com base nos trabalhos elencados nesta seção, considera-se uma abordagem explorando lógica fuzzy valorada intervalarmente, visando estruturação de um sistema de inferência fuzzy baseado nas relações de efeito e causa entre variáveis da CTR, as quais colaboram na classificação de tráfego de *streaming* de vídeo. A estratégia proposta modela tanto a incerteza na determinação de parâmetros quanto a imprecisão referente aos cálculos envolvendo variáveis relevantes para interpretação do fluxo de redes.

Tabela 1. Trabalhos Relacionados.

Artigo	TTR	ALFG	TCR	ELF
[Asmuss and Lauks 2015]	Classificação de Tráfego e Detecção de Anomalias	fuzzy C-means, Fukuyama and Sugeno index, Xie e Beni index, separation e compactness index		Clustering
[Shalaginov and Franke 2015]	TR anômalo e malicioso	Neuro-Fuzzy (NF), Self-Organizing Maps (SOM), Mean Absolute Error (MAE), Relative Absolute Error (RAE) e Mean Absolute Percent Error (MAPE)		Neuro-Fuzzy
[Qader et al. 2017]	Classificação de Falhas de Rede	K-Means, Fuzzy C Means e Expectation Maximization		Clustering
[Ducange et al. 2017]	Fluxos TR	Multi-objective evolutionary fuzzy classifiers (MOEFCs)	✓	MOEFCs
[Abdullah and Al-Hashmi 2018]	Inspeção de Anomalias em TR	Time Series Evolving Fuzzy Engine (TiSEFE)		T1FL
[Al-Obeidat and El-Alfy 2019]	TR anômalo e malicioso	Abordagem híbrida combinando árvore de decisão e método de classificação multicritério fuzzy	✓	Hybrid
[Iglesias et al. 2019]	Classificação de TR anômalao e identificação de de ataques por tipo	classificações Multiclass Fuzzy Classification e neuro-fuzzy	✓	Árvores de Decisão Fuzzy (Fuzzy Decision Trees)
[Shifa et al. 2020]	Streaming de vídeo em tempo real	modelo Fuzzy Threat Classifcation (FTC)	✓	T1FL
[Parfenov et al. 2020]	Identificação de ataques em TR	Funções de fuzzificação triangulares de pertinência e criação de bases de regras a partir de abordagem com árvores de decisão	✓	Neuro fuzzy/T1FL
FNC	Streaming de Vídeo	Abordagem Fuzzy Valorada Intervalarmente	✓	T2FL

Tipos de Tráfego de Rede (TTR) Abordagem de Lógica Fuzzy Geral (ALFG) Tráfego Criptografado (TCR) Extensões Lógica Fuzzy (ELF) *FuzzyNetClass* (FNC) Tráfego de Rede (TR) T1FL (Lógica Fuzzy Tipo-1) T2FL (Lógica Fuzzy Tipo-2)

3. Fundamentos de Lógica Fuzzy

Lotf Zadeh introduziu a T2FL em 1975 como extensão da FL [Zadeh 1975]. Seu surgimento está relacionado com a insuficiência da teoria dos Conjuntos Fuzzy (*Fuzzy Sets - FS*) tradicionais, atribuindo apenas um número no intervalo unitário $[0, 1]$, na modelagem das incertezas inerentes à definição das funções de pertinência dos antecedentes e consequentes em um Sistema de Inferência Fuzzy (*Fuzzy Inference System - FIS*) [Mendel 2003]. Ao considerar atribuições valoradas por subintervalos do intervalo unitário, busca-se uma modelagem mais ampla, onde conjuntos fuzzy valorados intervalarmente podem ser usados em situações onde existe incerteza sobre o grau, formas, e/ou parâmetros das funções de pertinência [Karnik and Mendel 1998], fornecendo uma estratégia no tratamento das incertezas considernado múltiplos critérios obtidos através de distintos especialistas, ou ainda, manipulando parâmetros extraídos via simuladores.

Nesta proposta, a Lógica Fuzzy Type-2 Intervalar (IT2FL), baseada na teoria T2FS, é sugerida para o tratamento de incertezas, permitindo atribuir um intervalo como o grau de pertinência de um elemento x em um conjunto fuzzy A [Mendel et al. 2006]. Assim, estendendo a teoria Fuzzy Set (FS), a teoria IT2FS é capaz de modelar a imprecisão com uma habilidade adicional, e a imprecisão (não especificidade) como outro aspecto importante, refletindo essa incerteza no diâmetro do grau de pertinência intervalar.

Definição 1 *[Karnik and Mendel 1998] Um conjunto fuzzy A, cujos elementos são caracterizados por uma função de pertinência ($\mu_A(x, u)$), é definido da seguinte forma:*

$$\tilde{A} = \{(x, \mu_A(x, u)) : x \in \chi, u \in J_x \subseteq [0, 1]\}. \quad (1)$$

Pela Definição 1, para cada elemento x do universo $\chi \neq \emptyset$, tem-se um mapeamento $A(x)\colon [0, 1] \to [0, 1]$. Um T2FS também pode ser dado como $\{(x, A(x, t)) : x \in \chi, t \in [0, 1]\}$ quando $A(x, .)\colon [0, 1] \to [0, 1]$ é dado como $A(x, t) = A(x)(t)$, para cada $x \in \chi$, $t \in [0, 1]$. Em particular, para T1FS $A(x)$, tem-se um número real em $[0, 1]$, $\forall x \in \chi$.

Definição 2 *[Mendel et al. 2006] Se $\forall x \in \chi$ tem-se $\mu_A(x) = 1$, então A é um conjunto fuzzy valorado intervalarmente tipo-2 (IVFS), correspondente a*

$$A(x) = \{(u, 1) : u \in J_x \subseteq [0, 1]\}, \forall x \in \chi.$$

E ainda, um conjunto fuzzy valorado intervalarmente é um caso particular de T2FS) [Gehrke et al. 1996]. Seja um IVFS A, $A(x) = [\underline{A}(x), \overline{A}(x)]$, $\forall x \in \chi$. Além disso, sejam os IVFS A, B, as correspondentes operações de (i) Complemento, (ii) União e (iii) Interseção são também IVFS, respectivamente definidas pelas seguintes expressões:

$$\begin{aligned} A_C(x) &= [1 - \overline{A}(x), 1 - \underline{A}(x)]; \\ A(x) \cup B(x) &= [\max(\underline{A}(x), \underline{B}(x)), \max(\overline{A}(x), \overline{B}(x))]; \\ \mu_{A \cap B}(x) &= [\min(\underline{A}(x), \underline{B}(x)), \min(\overline{A}(x), \overline{B}(x))], \forall x \in \chi. \end{aligned}$$

Neste artigo, denota-se $A(x) = X, B(x) = Y, \forall x \in \chi$, U como o conjunto de todos valores fuzzy do intervalo unitário $[0, 1]$ e $\mathbb{U}$ como o conjunto de valores fuzzy intervalares. A ordem parcial em $\mathbb{U}$ é a ordem do produto [Klement et al. 2004] dada pela expressão: $X \leq Y$ se e somente se $\underline{X} \leq \underline{Y}$, e $\overline{X} \leq \overline{Y}$.

Um sistema baseado em IVFL pode estimar funções de entrada e saída, usando heurísticas e técnicas intervalares. A seguir seus principais blocos são descritos.

(1) Interface de Fuzzificação: O processo é realizado de acordo com a natureza e definição de um IVFS, inserindo no mecanismo de inferência além da incerteza relacionada as funções de pertinência de entrada, também a imprecisão nos cálculos via regras do sistema de inferência. Assim, para cada entrada $A(x)$ um vetor de entrada $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \chi^n$, $n \in \mathbb{N}^*$, está relacionado a um par de vetores em $\mathbb{U}^n$ obtidos da seguinte forma: $\left(\overline{A}(x_1), \ldots, \overline{A}(x_n)\right), (\underline{A}(x_1), \ldots, \underline{A}(x_n))$.

(2) Base de Regras (RB): Composta por regras que classificam variáveis linguísticas (VL) de acordo com o IVFS A;

(3) Unidade de Decisão Lógica: Realiza as operações de inferência entre os dados de entrada e as condições impostas Sistema Baseado em Regras Fuzzy Valorado Intervalarmente (IV-FRBS), gerando a ação a ser realizada;

(4) Defuzzificação: Considera duas etapas principais:

(i) **Redutor de Tipo-1**, transformando um IVFS em FS, ou seja, fornece o melhor conjunto fuzzy que representa o IVFS, satisfazendo a seguinte premissa: quando todas as incertezas desaparecem , o resultado do IV-FRBS é reduzido para FRBS [Wu and Nie 2011];

(ii) **Defuzzificação**: a saída do SIFT2 usa a média dos pontos limites $\underline{Y}$ e $\overline{Y}$:

$$y = (\underline{Y} + \overline{Y})/2 = (\underline{B}(x) + \overline{B}(x))/2, \forall x \in \chi, \quad (2)$$

sendo que os valores $\underline{Y}$ e $\overline{Y}$ são calculados via método iterativo de Karnik e Mendel (algoritmo KM)[Mendel 2013], ou obtido através do uso de um método convencional, como o centroide, no valor final da inferência.

4. Fundamentos da Classificação de Tráfego de Rede

Esta seção aborda os principais tópicos relacionados aos fundamentos da classificação do tráfego de rede, particularmente aqueles relevantes para proposta deste trabalho.

4.1. Fluxos de Rede

As informações dos cabeçalhos de controle de pacotes determinam a relação entre eles, possibilitando a construção de um fluxo de pacotes. O fluxo consiste em cinco campos, formados pelo endereço IP de origem, endereço IP de destino, protocolo da camada de transporte, endereço da porta de origem e endereço da porta de destino [Claise et al. 2004]. A partir destes dados torna-se possível coletar e analisar outras informações relevantes para classificar o tráfego da rede, como o número de bytes gerados, o tempo de duração e a diferença de tempo entre cada pacote.

4.2. Tráfego de rede: estratégias de classificação

A revisão da literatura aponta que as diferentes estratégias de classificação de tráfego sofreram uma evolução a partir da complexidade dos protocolos e serviços [Velan et al. 2015]. O primeiro método utilizado foi identificar diretamente as informações contidas nos cabeçalhos, por exemplo, relacionando a porta de comunicação com o tipo de aplicação ou protocolo.

Com o uso generalizado de criptografia no tráfego de aplicativos, a análise por conteúdo de pacotes tornou-se uma estratégia imprecisa para classificar o tráfego de rede [Draper-Gil et al. 2016].

4.3. Streaming de Vídeo

O streaming de vídeo em rede é caracterizado pelo envio de blocos denominados chunks [Sani et al. 2017]. Chunks são segmentos de dados enviados de acordo com as condições da rede e os recursos disponíveis no cliente e no servidor. Devido à possibilidade de alteração da qualidade do vídeo durante a transmissão e ao uso de protocolos como HTTPS, HTTP/2 e, QUIC, a aplicação de métodos tradicionais de identificação e classificação de tráfego torna-se menos eficaz [Bentaleb et al. 2018].

4.4. CicFlowMeter

CicFlowMeter[1] é um gerador e analisador de fluxo de tráfego de rede, sua escolha deu-se pela ampla adoção na comunidade acadêmica. Esta ferramenta pode ser utilizada para gerar fluxos bidirecionais, onde o primeiro pacote determina as direções para frente (origem para destino) e para trás (destino para origem), desta forma mais de 70 informações estatísticas são fornecidas para análise.

4.5. Seleção de Atributos

A ferramenta WEKA[2] e o algoritmo de seleção de atributos CfsSubsetEval [Hall 1999] são amplamente usados pela comunidade tecno-científica e foram aplicados para seleção dos atributos mais relevantes para a classificação do tráfego de rede, juntamente com o método de pesquisa BestFirst aplicando os parâmetros padrões. Os resultados foram analisados e filtrados com auxílio de um especialista em rede.

A Figura 1 mostra o procedimento aplicado para descobrir os atributos mais relevantes para classificar o tráfego de streaming de vídeo. Os conjuntos de dados[3] utilizados para realizar a descoberta de atributos foram gerados a partir das capturas realizadas, e ainda, através de conjuntos de dados disponíveis publicamente.

Os atributos selecionados foram "*Packet Length Mean*", "*Fwd Packet Length Std*" e "*Bwd IAT Total*". O atributo *Packet Length Mean* exibe o valor médio do tamanho do pacote no fluxo da rede, *Fwd Packet Length Std* está relacionado ao desvio padrão do valor médio do tamanho do pacote na direção de upload do fluxo, e por fim, *Bwd IAT Total* mostra o tempo total entre pacotes na direção de download do fluxo.

5. *FuzzyNetClass*: Proposta de Arquitetura

FuzzyNetClass[4] foi concebido para classificar tráfego de rede e identificar fluxos de *streaming* de vídeo. A proposta do *FuzzyNetClass* considera uma base de regras atuando em três etapas: Fuzzificação, Inferência e Defuzzificação retornando como saída o nível de classificação do fluxo da rede analisado, permitindo (ou não) sua identificação como um vídeo. A modelagem do sistema *FuzzyNetClass* foi realizada utilizando a plataforma Juzzy[5] [Wagner 2013].

[1] `https://github.com/ahlashkari/CICFlowMeter`
[2] `https://www.cs.waikato.ac.nz/ml/weka/`
[3] `https://github.com/emmonks/datasets`
[4] `https://github.com/brunomourapaz/FuzzyNetClass`
[5] `http://juzzy.wagnerweb.net/`

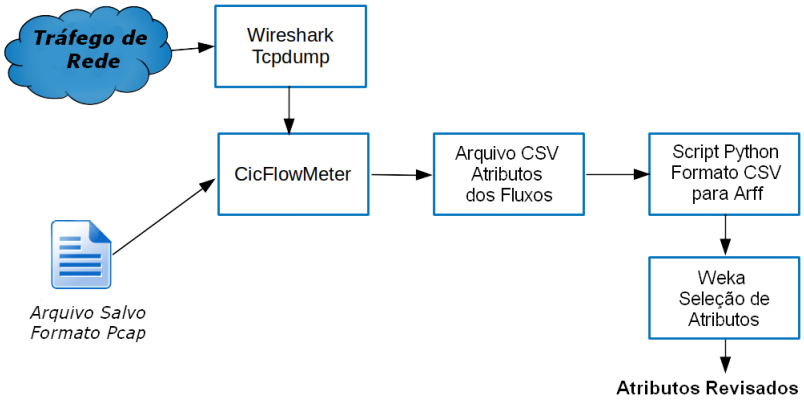


Figura 1. Procedimentos para a escolha de atributos

5.1. Base de Dados - Funções de Pertinência

No estudo das variáveis considerou-se a opinião dos especialistas, e cada VL associada a um IVFS tem uma representação gráfica trapezoidal das funções de pertinência.

Os valores dos atributos são aplicados à escala padrão no intervalo $[0, 10]$, estabelecendo o valor 10 como limite e assim, contribuindo para melhor visualização gráfica. Assim, para *Packet Length Mean*, foi usado a Eq. (3), Fwd Packet Length Std a Eq. (4) e Backward Iat Total Eq. (5), na obtenção dos graus de pertinência:

$$PLM = (nf_i(PLM)/MaxPLM * 10) \quad (3)$$

$$PLS = (nf_i(PLS)/MaxPLS * 10) \quad (4)$$

$$BIAT = (nf_i(BIAT)/MaxBIAT) * 10 \quad (5)$$

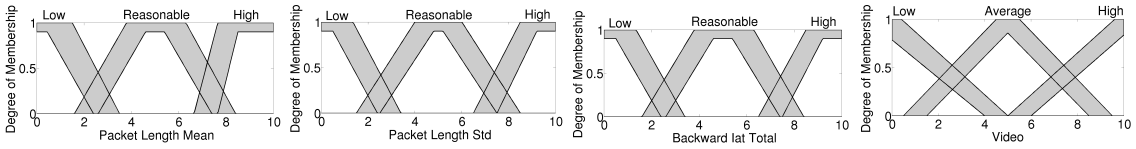
considerando os seguintes parâmetros para cada fluxo de rede:

- nf_i representa um fluxo de rede capturado;
- PLM é um atributo médio de tamanho de pacote;
- PLS é um atributo de desvio padrão de comprimento de pacote;
- $BIAT$ considera o tempo total de chegadas de pacotes no sentido de recebimento;
- $\max PLM$ é o valor total do atributo médio de tamanho de pacote mais alto identificado;
- $\max PLS$ é o valor total do maior desvio padrão do tamanho do pacote de encaminhamento atribuído identificado;
- $\max BIAT$ é o valor total do atributo de tempo total entre chegadas de recebimentos de pacotes mais alto identificado.

Os termos linguísticos que definem os conjuntos para variável *Packet Length Mean* (PLM) são os seguintes: "*Low*" (PLML), "*Reasonable*" (PLMR) e "*High*" (PLMH - melhor caso). Denota-se $PLM = a$ e $a \in [0, 10]$. Veja a representação das correspondentes funções de pertinência na Figura 2(a).

O atributo *Fwd Packet Length Std* (PLS) é usado como entrada e obtido pela leitura do fluxo de rede analisado. Os termos para os conjuntos definidos para esta variável são: "*Low*" (PLSL), "*Reasonable*" (PLSR - melhor caso) e "*High*" (PLSH). Denota-se $PLS = b$ e $b \in [0, 10]$. Estas funções de pertinência são apresentadas na Figura 2(b).

No projeto dos FSs para *Backward Iat Total* (BIAT), foram criados os seguintes termos: "*Low*" (BIATL - melhor caso), " *Reasonable*" (BIATR) e "*High*" (BIATH). Denota-se $BIAT = c$ e $c \in [0, 10]$. Estas funções de pertinência são vistas na Figura 2(c).



(a) PLM (b) PLS (c) BIAT (d) Video

Figura 2. PLM, PLS, BIAT, e Video na escala padrão

5.2. Fuzzificação

Nessa etapa, ocorre o mapeamento dos valores de entrada (já ajustados para escala observada na seção 5.1) para o domínio fuzzy, como apresenta a Figura 3.

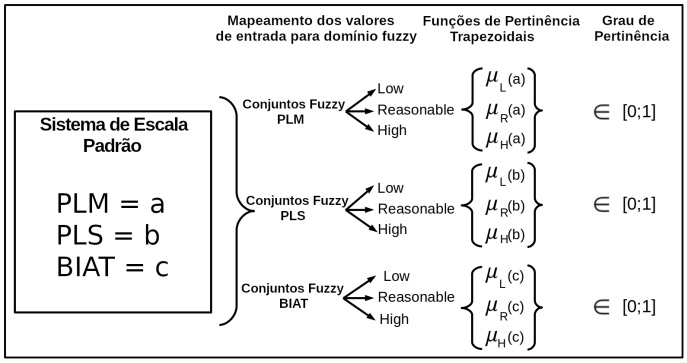


Figura 3. Processo de Fuzzificação

5.3. Base de Regras

A Base de Regras do *FuzzyNetClass* foi desenvolvida com intuito de ser facilmente compreensível e editável descrevendo de maneira consistente a estratégia de controle considerando três fatores: (i) as VL nomeiam os CF, tornando a modelagem do sistema mais próxima do mundo real; (ii) são utilizadas conexões lógicas do tipo "AND" para criar a relação entre as variáveis de entrada; (iii) as implicações são do tipo $Modus\ Ponens$ (modo afirmativo): "Se ((x é A) e (y é B)) então (z é C)."

5.4. Inferência

No processo de Inferência, tem-se operações entre conjuntos, combinação dos antecedentes das regras e implicações modeladas pelo operador *Modus Ponens Generalizado*, ocorrendo em três etapas:

(i) Aplicação da Operação Fuzzy: Nas regras agregadas através do operador "AND", utiliza-se o método MIN (mínimo) sobre os valores retornados da fuzzificação;

(ii) Aplicação do Método de Implicação Fuzzy: realizada pela combinação entre o valor obtido na aplicação do operador fuzzy e os valores do conjunto fuzzy de saída da regra, utilizando o método MIN (mínimo) sobre estas combinações;

(iii) Aplicação do Método de Agregação Fuzzy: considera a composição dos resultados fuzzy da saída de cada regra, utilizando o método MAX (máximo), assim criando uma única região fuzzy para ser analisada pelo próximo processo do módulo.

5.5. Defuzzificação

Na transformação da região modelando o processo de inferência, aplica-se o Centro da Área, ou seja, o método calcula o centroide, pela união das contribuições de regras discutidas nas seções 5.3 e 5.4, dado pela expressão: $u = \sum_{i=1}^{N} u_i \mu_{OUT}(u_i) / \sum_{i=1}^{N} \mu_{OUT}(u_i)$.

6. *FuzzyNetClass*: Visão Geral da Operação

A Figura 4 apresenta um fluxograma da arquitetura *FuzzyNetClass* proposta, onde cada etapa realizada na classificação do tráfego de streaming de vídeo é ilustrada. O módulo Wireshark/Tcpdump é responsável por utilizar as ferramentas para capturar o tráfego de rede e gerar arquivos no formato pcap.

Em seguida, a etapa da ferramenta CicFlowMeter visa extrair os fluxos da rede processando os arquivos capturados e produzindo arquivos CSV contendo 77 atributos extraídos de cada fluxo da rede.

Na etapa de Filtro e Seleção de Atributos, um *script* Python é executado usando a biblioteca Pandas[6], que tem o objetivo extrair e normalizar os atributos selecionados e, em seguida, tornando os valores PLS, PLM e BIAT prontos para uso como entradas.

A fuzzificação aplica as funções de pertinência trapezoidais. Em seguida, o processo de inferência considera a base de regras, e após o redutor tipo-1, transformam-se o conjunto fuzzy tipo-2 de saída em um conjunto fuzzy tipo-1. A defuzzificação é alcançada, retornando um único valor (crisp) como saída.

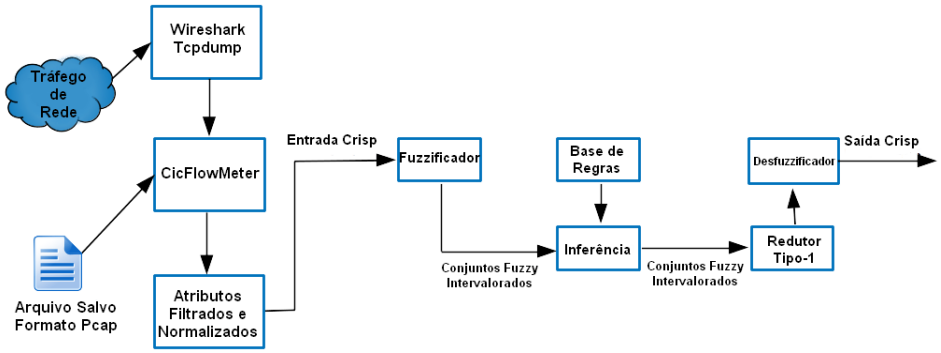


Figura 4. *FuzzyNetClass*: Proposta de Arquitetura

7. *FuzzyNetClass*: Avaliação

Para validar a proposta, foram capturados 339 vídeos da plataforma Youtube no formato VoD (*Video on Demand*). O tempo de captura de cada vídeo foi de 2 a 4 minutos para permitir o registro de possíveis alterações de qualidade devido à intermitência e variações nas condições da rede. A ferramenta Tcpdump[7] foi utilizada para salvar os arquivos

[6] https://pandas.pydata.org/
[7] https://www.tcpdump.org/

capturados. O navegador Firefox foi utilizado para acessar os vídeos em sua configuração padrão. Os vídeos foram capturados ao longo de três meses com sessões de captura em diferentes horários do dia, em dias de semana e finais de semana.

Os arquivos capturados foram submetidos à ferramenta CicFlowMeter com os seguintes parâmetros: 1200s de tempo limite de fluxo e tempo de inatividade em 5s. Esses parâmetros têm a função de limitar o tempo de fluxo da rede. No caso deste trabalho, o tempo de captura foi limitado ao tempo máximo do vídeo original. O protocolo QUIC predominou na transmissão dos vídeos analisados, mas o protocolo HTTPS foi utilizado em alguns casos. Os protocolos QUIC e HTTPS são criptografados por padrão. Os fluxos foram capturados em um ambiente de acesso doméstico à Internet com um link de 240 Mbit/s com tecnologia *Gigabit Passive Optical Network* (GPON).

Arquivos adicionais com capturas foram coletados, gerando outros tipos de fluxos de tráfego de rede. A captura refere-se a uma rede acadêmica de uma universidade no sul do Brasil, e também via conjuntos de dados públicos [Moustafa and Slay 2015, Cho et al. 2000]. Os fluxos de rede dos seguintes protocolos foram capturados: DNS, NTP, FTP, SSH, HTTP, HTTPS e QUIC.

Para validar os fluxos de vídeo em HTTPS e QUIC foram realizadas análises na ferramenta Wireshark[8] por um especialista em redes. A partir de capturas de dados reais em ambiente de redes, foram coletados 3100 fluxos do conjunto de dados 20211017 e 11250 do conjunto de dados 20211024 modelados como "ruído". No processo de classificação, no conjunto de dados 20211017, foram usados 247 fluxos de streaming de vídeo e 92 no conjunto de dados 2021124.

Os arquivos capturados foram submetidos na ferramenta CicFlowMeter, e a saída em formato CSV foi aplicada a um script Python com a biblioteca Pandas. A saída gera um arquivo CSV com os atributos selecionados e valores de atributos normalizados.

Na sequência do processo, os conjuntos de dados gerados são processados pelo *FuzzyNetClass*, que realiza todas as etapas do sistema de inferência fuzzy intervalar. A saída fornece o nível do fluxo analisado, que será utilizado para a classificação relacionada ao tipo de *streaming* de vídeo. Na Tabela 2 são apresentados os resultados da execução do *FuzzyNetClass*, destacando-se os percentuais e a quantidade de fluxos classificados em cada grupo quanto ao grau de pertinência para cada conjunto de saída.

Na faixa de 0,0 a 5,6, os fluxos que se enquadram no conjunto Baixo são classificados. De 5,61 a 8,0, agrupa os fluxos do conjunto Médio. Por fim, de 8,01 a 10, são considerados os fluxos do conjunto Alto. Na validação dos fluxos de *streaming* de vídeo existentes nos conjuntos de dados houve 78,13% de precisão em relação ao total de 247 fluxos para o caso do 20211017 e no caso do 20211024 houve 77,17% do total de 92 fluxos classificados nas faixas Média ou Alta respectivamente. O restante dos fluxos obtiveram classificação na faixa Baixa para *streaming* de vídeo. Os resultados obtidos são promissores e apontam para a continuidade do esforço de estudo e pesquisa.

8. Conclusões

Neste artigo, foi apresentado o *FuzzyNetClass*, uma nova abordagem para classificação de tráfego de *streaming* de vídeo usando T2FL. Os resultados preliminares mostraram

[8] `https://www.wireshark.org/`

Tabela 2. Classificação de Streaming de Vídeo - Resultados Sumarizados

Dataset	De 0,0 a 5,6		De: 5,61 to 8,0		De: 8,01 to 10,0	
	RFx1	PFx1	RFx2	PFx2	RFx3	PFx3
20211017	3044	94,00%	86	2,66%	107	3,3%
20211024	11296	99,38%	38	0,33%	33	0,29%

(RFx) Total de Fluxos (PFx) Porcentagem de fluxos dentro da faixa

uma taxa de precisão razoável usando conjuntos de dados de fluxos de rede conhecidos na literatura e capturas de fluxos atuais. A principal vantagem do *FuzzyNetClass* é o tratamento das incertezas na modelagem das funções de pertinência fuzzy, abordando as imprecisões obtidas tanto na mensuração das variáveis do ambiente da rede, quanto nos cálculos do sistema de inferência fuzzy intervalar.

Para a continuação do trabalho considera-se uma extensão que suporte ordens admissíveis [Bustince et al. 2013, Zapata et al. 2017, Moura et al. 2019] para comparar diferentes métodos de ordenação nos conjuntos de dados, adicionando ao *FuzzyNetClass* uma etapa dinâmica para geração de regras. Além disso, possibilitar ajustes permitindo a classificação do tráfego da rede de *streaming* de vídeo nos formatos ao vivo (Live) e VoD.

Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (**CAPES**)

Referências

Abdullah, S. A. and Al-Hashmi, A. S. (2018). TiSEFE: Time series evolving fuzzy engine for network traffic classification. *Int. J. Commun. Netw.*, 10(1):116–124.

Al-Obeidat, F. and El-Alfy, E.-S. (2019). Hybrid multicriteria fuzzy classification of network traffic patterns, anomalies, and protocols. *Pers Ubiquitous Comput*, 23(5):777–791.

Asmuss, J. and Lauks, G. (2015). Network traffic classification for anomaly detection fuzzy clustering based approach. In *2015 12th (ICNC-FSKD)*, pages 313–318. IEEE.

Bentaleb, A., Taani, B., et al. (2018). A survey on bitrate adaptation schemes for streaming media over http. *IEEE Commun. Surv. Tutor.*, 21(1):562–585.

Bujlow, T., Carela-Español, V., and Barlet-Ros, P. (2015). Independent comparison of popular DPI tools for traffic classification. *Computer Networks*, 76:75–89.

Bustince, H., Fernández, J., et al. (2013). Generation of linear orders for intervals by means of aggregation functions. *Fuzzy Sets and Systems*, 220:69–77.

Cho, K., Mitsuya, K., and Kato, A. (2000). Traffic data repository at the WIDE project. In *Proc. of the Freenix Track: 2000 USENIX ATC, June 18-23, 2000, San Diego, CA, USA*, pages 263–270. USENIX.

Claise, B., Sadasivan, G., et al. (2004). Rfc 3954: Cisco systems netflow services export version 9. *IETF http://www. ietf. org/rfc/rfc3954. txt*.

Draper-Gil, G., Lashkari, A. H., et al. (2016). Characterization of encrypted and vpn traffic using time-related. In *Proc. of the 2nd ICISSP*, pages 407–414.

Ducange, P., Mannarà, G., et al. (2017). A novel approach for internet traffic classification based on multi-objective evolutionary fuzzy classifiers. In *2017 FUZZ-IEEE*, pages 1–6. IEEE.

Gehrke, M., Walker, C., and Walker, E. (1996). Some comments on interval valued fuzzy sets. *Int. Journal of Intelligent Systems*, 11(10):751–759.

Hall, M. A. (1999). Correlation-based feature selection for machine learning.

Iglesias, F., Milosevic, J., and Zseby, T. (2019). Fuzzy classification boundaries against adversarial network attacks. *Fuzzy Sets and Systems*, 368:20–35.

Karnik, N. N. and Mendel, J. M. (1998). Introduction to type-2 fuzzy logic systems. In *1998 IEEE Int. Conf. on Fuzzy Systems Proc. IEEE World Congress on Computational Intelligence*, volume 2, pages 915–920 vol.2.

Klement, E., Mesiar, R., and Pap, E. (2004). Triangular norms. position paper I: basic analytical and algebraic properties. *Fuzzy Sets and Systems*, 143(1):5–26.

Mendel, J. M. (2003). Fuzzy sets for words: a new beginning. In *Fuzzy Systems, 2003. FUZZ '03. The 12th IEEE Int. Conf. on*, volume 1, pages 37–42.

Mendel, J. M. (2013). On km algorithms for solving type-2 fuzzy set problems. *IEEE Transactions on Fuzzy Systems*, 21(3):426–446.

Mendel, J. M., John, R. I., and Liu, F. (2006). Interval type-2 fuzzy logic systems made simple. *IEEE Trans. Fuzzy Systems*, 14(6):808–821.

Moura, B. M. P., Schneider, G. B., et al. (2019). Allocating virtual machines exploring type-2 fuzzy logic and admissible orders. In *2019 IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE)*, pages 1–6.

Moustafa, N. and Slay, J. (2015). Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set). In *2015 Int. Conf. Mil. Commun. Inf. Syst. ICMCIS*, pages 1–6.

Parfenov, D., Zabrodina, L., et al. (2020). Research of multiclass fuzzy classification of traffic for attacks identification in the networks. In *J. Phys. Conf. Ser.*, volume 1679, page 042023. IOP Publishing.

Qader, K., Adda, M., and Al-Kasassbeh, M. (2017). Comparative analysis of clustering techniques in network traffic faults classification. *Int. j. innov. res. comput. commun. eng.*, 5(4):6551–6563.

Sandvine (2020). The global internet phenomena report covid-19 spotlight.

Sani, Y., Mauthe, A., and Edwards, C. (2017). Adaptive bitrate selection: A survey. *IEEE Communications Surveys & Tutorials*, 19(4):2985–3014.

Shalaginov, A. and Franke, K. (2015). Automated generation of fuzzy rules from large-scale network traffic analysis in digital forensics investigations. In *2015 7th Int. Conf of Soft Computing and Pattern Recognition (SoCPaR)*, pages 31–36. IEEE.

Shifa, A., Asghar, M. N., et al. (2020). Fuzzy-logic threat classification for multi-level selective encryption over real-time video streams. *J. Ambient Intell. Humaniz. Comput.*, 11(11):5369–5397.

Velan, P., Čermák, M., et al. (2015). A survey of methods for encrypted traffic classification and analysis. *Int. Journal of Network Management*, 25(5):355–374.

Wagner, C. (2013). Juzzy - a java based toolkit for type-2 fuzzy logic. In *2013 IEEE Symp. on Advances in Type-2 Fuzzy Logic Systems (T2FUZZ)*, pages 45–52.

Wu, D. and Nie, M. (2011). Comparison and practical implementation of type-reduction algorithms for type-2 fuzzy sets and systems. In *FUZZ-IEEE*, pages 2131–2138. IEEE.

Zadeh, L. (1975). The concept of a linguistic variable and its application to approximate reasoning—i. *Information Sciences*, 8(3):199 – 249.

Zapata, H., Bustince, H., et al. (2017). Interval-valued implications and interval-valued strong equality index with admissible orders. *Int J Approx Reason*, 88:91–109.