Descobrindo perfis de tráfego de usuários: uma abordagem não supervisionada

Ananda G. Streit, Rosa M.M. Leão, Edmundo de Souza e Silva, Daniel S. Menasché

Universidade Federal do Rio de Janeiro (UFRJ), Rio de Janeiro, Brasil

{agstreit,rosam,edmundo,sadoc}@land.ufrj.br

Abstract. *The increasing complexity of home networks calls for novel strategies towards efficient network management and workload characterization. In this work we use unsupervised machine learning techniques with the objective of discovering users' traffic profiles. In partnership with an ISP we collected the download and upload traffic from more than 2,000 home routers of the ISP clients. We then use a tensor decomposition technique (PARAFAC) to extract relevant features from our network traces. With the results of PARAFAC and a hierarchical clustering algorithm, we group users with similar daily traffic patterns. To characterize users' behavior over periods longer than a day, we use the information of the clusters and a Hidden Markov Model.*

Resumo. *As redes domésticas estão cada vez mais complexas. Portanto, é essencial a elaboração de estratégias inovadoras para caracterizar essa nova demanda. Neste trabalho usamos técnicas não supervisionadas de aprendizado de máquina com o objetivo de entender o perfil de tráfego dos usuários. Em parceria com um ISP, coletamos o tráfego de download e upload de mais de 2.000 roteadores domésticos. Usamos uma técnica de decomposição de tensores (PARAFAC) para extrair fatores relevantes de uso da rede e um algoritmo de clusterização para agrupar usuários com padrões de tráfego diário similares. Para caracterizar o comportamento dos usuários em períodos maiores que um dia, usamos a informação dos clusters e um modelo de Markov oculto.*

1. Introdução

Motivação A crescente complexidade da Internet, expressiva após uma explosão sem precedentes do número de dispositivos IoT conectados a roteadores domésticos, exige novas estratégias para o gerenciamento eficiente da rede. Pesquisadores e profissionais de ISPs estão usando técnicas de aprendizado de máquina para entender melhor o comportamento do usuário doméstico. Tradicionalmente, esse tema esteve fora do escopo dos ISPs, seja devido a preocupações com a privacidade ou devido à incapacidade de processamento de grandes volumes de dados. No entanto, o comportamento do usuário doméstico é fundamental para lidar com problemas de segurança e de desempenho da rede.

Pesquisa prévia Entender as características do tráfego gerado pelos usuários é de suma importância para uma variedade de aplicações, como detecção de tráfego anômalo, previsão do tráfego futuro e alocação adequada de recursos da rede [Fumo et al. 2017]. Em particular, devido à sua importância para o planejamento da rede, a classificação de tráfego tem sido um assunto popular há muitos anos (e.g, [Kim et al. 2016, Morichetta and Mellia 2018, Nguyen and Armitage 2008, Soysal and Schmidt 2010, Wright et al. 2004]).

Técnicas de aprendizado de máquina têm sido usadas para analisar a grande quantidade de dados coletada por ferramentas de monitoramento [Fumo et al. 2017]. Trabalhos anteriores focam principalmente na inspeção profunda de pacotes (DPI) e/ou consideram padrões predeterminados para classificar fluxos de tráfego em aplicações específicas [Trevisan et al. 2018]. A literatura sobre técnicas de aprendizado de máquina não supervisionadas para dados coletados sem DPI ainda é escassa, talvez devido às dificuldades para acessar medições reais de usuários residenciais [Crovella and Krishnamurthy 2006].

Considere, por exemplo, o recente trabalho de Morichetta e Mellia (2018), onde o tráfego HTTP foi monitorado para extrair URLs e, em seguida, caracterizar o tráfego do usuário. Embora nosso trabalho compartilhe algumas metas comuns, por razões de privacidade, o conjunto de dados considerado no nosso trabalho não depende de identificadores de solicitação de objeto, o que naturalmente leva a diferentes tipos de técnicas de aprendizado de máquina para extrair características relevantes dos dados.

Objetivos Neste trabalho usamos um conjunto de dados coletado em parceria com um ISP localizado no Brasil. Reunimos dados do tráfego de download e upload de mais de 2.000 roteadores domésticos, medidos a cada minuto durante 28 dias (de 20 de agosto a 16 de setembro de 2018). Apesar da nossa coleta estar limitada a um único país, acreditamos que nosso estudo seja relevante para que os ISPs possam estrategicamente melhorar a alocação de banda.

Esta não é a primeira vez que uma coleta de dados desse tipo é realizada com tal granularidade na borda da rede. No trabalho recente de [Trevisan et al. 2018] foram analisados dados similares aos nossos, reunidos por um período de cinco anos em um ISP na Itália. Porém, diferente do nosso estudo, [Trevisan et al. 2018] realiza uma investigação macro do acesso a rede; um dos seus objetivos foi examinar as mudanças que ocorreram no tráfego agregado dos usuários ao longo desses cinco anos.

Diferentemente de [Trevisan et al. 2018], realizamos uma análise micro do acesso à rede doméstica, respondendo as seguintes questões: (1) Como extrair com eficiência características relevantes do nosso conjunto de dados, sem pré-rotular esses dados e preservando a privacidade dos usuários (por exemplo, assumindo que o tráfego é totalmente criptografado)? (2) Como interpretar e avaliar os resultados obtidos com as ferramentas de aprendizado de máquina?

Contribuições Em resumo, nossas principais contribuições são:

Framework para análise do comportamento do usuário ao acessar sua rede doméstica Propomos uma metodologia simples e direta para detectar estruturas temporais e padrões comportamentais da atividade de tráfego de usuários residenciais. Em suma, extraimos características das séries temporais de download e upload que sugerem padrões diários comuns de tráfego. Como consequência, conseguimos identificar perfis de comportamento dos usuários em períodos maiores que um dia.

Modelo do perfil diário de usuários residenciais Usamos uma técnica de decomposição de tensores (PARAFAC) para obter uma representação mais simples das amostras de tráfego diário. O PARAFAC é um método bem estabelecido e tem sido aplicado em áreas como psicometria [Kroonenberg 1983], quimiometria [Smilde et al. 2005, Stedmon and Bro 2008] e processamento de sinais, classificação

e aprendizado [Rabanser et al. 2017, Sidiropoulos et al. 2017]. Uma vantagem do PARAFAC, em comparação com outros métodos de decomposição fatorial (e.g, PCA, SVD ou Tucker), é a garantia de solução única sob condições moderadas. Dessa forma, capturamos fatores interpretáveis e intrínsicos ao nosso conjunto de dados, sem a necessidade de aplicar métodos externos de rotação (e.g, *Varimax* [Kaiser 1958]) para determinar o espaço fatorial mais adequado ao problema. O resultado obtido pelo PARAFAC pode ser usado para vários fins: classificação, previsão e clusterização. Neste artigo, nos concentramos no último e mostramos como o PARAFAC simplifica a tarefa de agrupamento de séries temporais em perfis diários.

Modelo do perfil de comportamento de usuários residenciais em períodos maiores que um dia Elaboramos um modelo de Markov oculto (HMM) às sequências de perfis diários detectados a partir do modelo PARAFAC. O modelo final obtido indica que os usuários tendem a manter um padrão específico ao longo do tempo. Por ser um modelo generativo, os padrões identificados podem ser reproduzidos para simulações e testes com base em dados reais de tráfego. Além disso, esse resultado pode facilitar tarefas de planejamento e gerenciamento da rede.

Organização O restante deste artigo está estruturado da seguinte forma. A Seção 2 descreve o nosso conjunto de dados de tráfego real. Também detalha a metodologia de pesquisa e os principais aspectos teóricos de cada método de aprendizado de máquina empregado. Os resultados e a análise correspondente estão na Seção 3. Conclusões e indicações de trabalhos futuros estão na Seção 4.

2. Metodologia

O objetivo desta seção é descrever as principais técnicas empregadas para identificar os perfis de tráfego de usuários domésticos a partir dos dados coletados nos roteadores residenciais.

A Figura 1 resume os principais passos da nossa metodologia. No primeiro passo definimos as variáveis representadas em cada dimensão do tensor a partir do conjunto de dados coletado (Seção 2.1). Em seguida, usamos um método de decomposição fatorial para extrair características relevantes dos dados, também conhecidas como *loadings* (Seção 2.2). Na terceira etapa, aplica-se clusterização usando como variáveis os *loadings* obtidos com o método de decomposição fatorial. O objetivo é agrupar séries temporais com características semelhantes (Seção 2.3). Posteriormente, é realizada a modelagem do comportamento do usuário usando um modelo de Markov oculto (*Hidden Markov Model* (HMM)) (Seção 2.5). A metodologia também permite a classificação de novas séries a partir do cálculo dos seus *loadings*, empregando-se um algoritmo de classificação (Seção 2.4).

2.1. Coleta de Dados e Tensores

O conjunto de dados usado em nossa análise experimental consiste no tráfego de download e upload de usuários domésticos. Cada amostra de tráfego de download (upload) contém o tráfego total (número de bytes) recebido (enviado) a cada minuto durante um determinado período de tempo. Neste artigo, consideramos amostras de tráfego coletadas durante 28 dias, de 20 de agosto a 16 de setembro de 2018, diretamente de 2.219 roteadores domésticos de diferentes usuários do ISP parceiro. O provedor possui porte médio, com cerca de 35.000 usuários e com planos variando entre 3 Mbps e 100 Mbps.

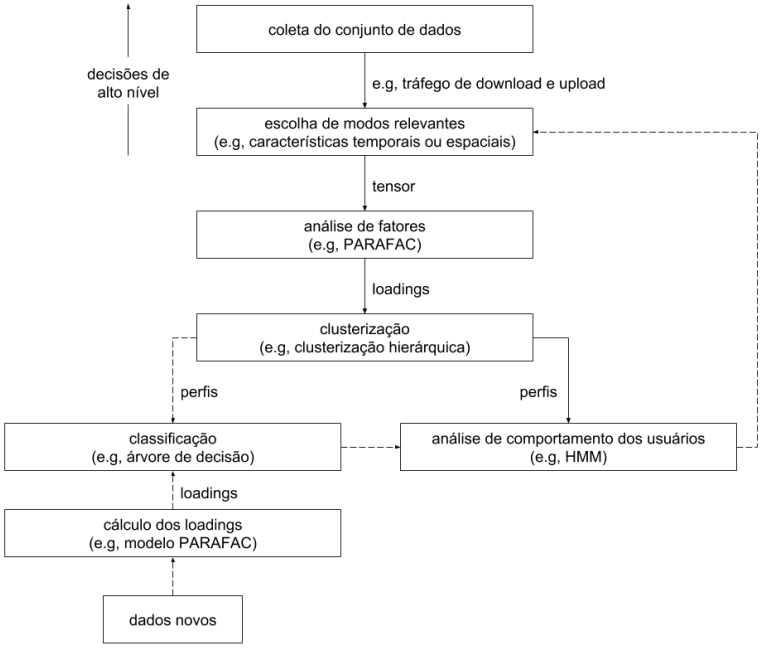

Figura 1. Framework para aprender perfis de tráfego

Em nosso estudo, uma série temporal de tráfego de download (upload) corresponde ao tráfego recebido (enviado) a cada minuto por um determinado usuário em um determinado dia. Uma série temporal corresponde a um par usuário-dia ou simplesmente um par UD. Denotamos por I o número de pares UDs em nosso conjunto de dados.

Conforme citado anteriormente, usamos a análise fatorial visando extrair as características relevantes dos dados de tráfego dos usuários. Para conhecer o perfil de tráfego dos usuários, é necessário analisar, em conjunto, a quantidade de tráfego de download e upload gerado por cada usuário ao longo do tempo. A estrutura mais adequada para representar esses dados é um tensor de ordem 3. As três dimensões do tensor, que também são conhecidas como modos, são definidas da seguinte forma: (1) o primeiro modo representa os pares usuários-dia (UDs), (2) o segundo modo representa os minutos e (3) o terceiro modo representa o tipo de tráfego (download ou upload). Os modos (1), (2) e (3) são denotados pelos índices i, j e k, respectivamente.

Definimos cada elemento do tensor como x_{ijk}, representando o tráfego k medido no par UD i no j-ésimo minuto do dia, onde $0 \leq i \leq I$, $0 \leq j \leq 1439$ e $0 \leq k \leq 1$. Fixamos $k = 0$ para o tráfego de download e $k = 1$ para o tráfego de upload.

Como a nossa metodologia é geral, poderíamos utilizar alternativamente outras métricas de interesse (e.g, atrasos e perdas) ou também considerar outros modos no tensor, como a localização geográfica dos usuários em caso de dados móveis. A escolha dos modos depende da aplicação e deve ser ajustada dependendo do conjunto de dados disponível e da interpretação e avaliação dos resultados obtidos (ver Figura 1).

2.2. Análise de fatores

Um dos principais objetivos da análise fatorial é decompor um array em um conjunto de fatores (variáveis latentes) e cargas (*loadings*) que descrevam os dados de forma mais

compacta em comparação com o conjunto original. Entre as diversas abordagens de decomposição de dados multidimensionais (tensor de ordem N), o PARAFAC e o Tucker são dois dos mais populares. Utilizamos o PARAFAC por sua simplicidade e garantia de solução única sob condições moderadas [Harshman and Lundy 1984, Kruskal 1983].

Seja $X \in \mathbb{R}^{I \times J \times K}$ um tensor de ordem 3, tendo x_{ijk} como um dos seus elementos. A decomposição PARAFAC é dada como,

$$x_{ijk} = \sum_{r=1}^{R} a_{ir} b_{jr} c_{kr} + e_{ijk}, \tag{1}$$

onde R é o número de fatores; e, a_{ir}, b_{jr} e c_{kr} são as cargas do fator r correspondentes a UD i, minuto j e tipo de tráfego k, respectivamente. Os residuais são denotados por e_{ijk}. O objetivo do método é calcular as cargas que minimizem a soma dos quadrados dos resíduos utilizando o algoritmo de mínimos quadrados alternantes (*Alternanting Least Squares* (ALS)) [Bro 1997].

Para definir o número de fatores (R) e avaliar se a solução é única e generalizável para outro conjunto de dados similar, aplicamos o método *Split-Half Validation* (SV) [Harshman 1984] junto com o *Tucker Congruence Coefficient* (TCC) [Lorenzo-Seva and Ten Berge 2006]. O método SV consiste em dividir os pares UDs de forma aleatória em quatro grupos (A, B, C e D) e testar a similaridade entre os modelos PARAFAC dos subconjuntos independentes $(A+B)$ vs. $(C+D)$ e $(A+C)$ vs. $(B+D)$.

Para cada uma das validações de similaridade temos dois modelos m_1 e m_2. A diferença entre eles é medida pelo TCC ($\phi_b(r)$ e $\phi_c(r)$, eq. 2) usando o vetor de cargas relacionado aos modos dos minutos ($\boldsymbol{b_r}$) e dos tipos de tráfego ($\boldsymbol{c_r}$),

$$\phi_b(r) = \frac{\sum_{j=1}^{J} b_{jr}^{m_1} \; b_{jr}^{m_2}}{\sqrt{\sum_{j=1}^{J} (b_{jr}^{m_1})^2 \sum_{j=1}^{J} (b_{jr}^{m_2})^2}}, \quad \phi_c(r) = \frac{\sum_{k=1}^{K} c_{kr}^{m_1} \; c_{kr}^{m_2}}{\sqrt{\sum_{k=1}^{K} (c_{kr}^{m_1})^2 \sum_{k=1}^{K} (c_{kr}^{m_2})^2}}, \tag{2}$$

respectivamente, onde $1 \leq r \leq R$. Dessa forma, obtemos um ϕ para cada fator e para cada um dos dois modos. Não calculamos ϕ para o modo relacionado aos UDs ($\boldsymbol{a_r}$), pois estamos comparando modelos gerados a partir de subconjuntos diferentes. Queremos saber se os fatores latentes são similares em $\boldsymbol{b_r}$ e $\boldsymbol{c_r}$ mesmo com populações diferentes de UDs. Quanto mais próximo ϕ é de um, mais semelhantes são as cargas dos fatores. Os resultados de Lorenzo-Seva e Ten Berge (2006) sugerem um valor mínimo de $0,95$ para que os dois vetores possam ser considerados iguais.

Caso nenhuma das duas comparações apresente o mesmo padrão de carga (baixa congruência), conclui-se que o conjunto de dados é inadequado para esse tipo de análise (i.e, não apresenta solução única) ou que o número de fatores deve ser reduzido para deixar de representar informação ruidosa e inerente a cada um dos subconjuntos modelados.

A Figura 2 ilustra a decomposição pelo PARAFAC. Definimos quatro fatores para o nosso modelo e atribuímos um nome para cada um deles a partir da interpretação dos resultados do modelo. Cabe ressaltar que nem sempre é possível obter resultados que possuam alguma interpretação. Mais detalhes são apresentados na Seção 3.

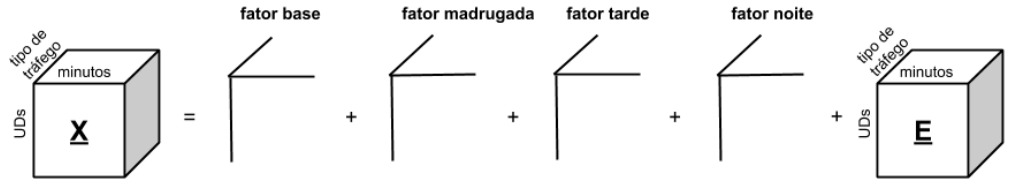

Figura 2. PARAFAC aplicado ao nosso conjunto de dados de treinamento

2.3. Clusterização de pares usuário-dia

O próximo passo da nossa metodologia é o agrupamento das séries temporais de tráfego com base nas cargas dos UDs obtidas a partir da análise fatorial. Um dos resultados da decomposição PARAFAC é o vetor de cargas $\mathbf{a_i} = (a_{i1}, \ldots, a_{iR})$ para cada UD i, onde $0 \leq i \leq I$ e $1 \leq r \leq R$. Portanto, o número de variáveis usadas na clusterização é igual ao número de fatores R do modelo.

Empregamos clusterização hierárquica aglomerativa, uma estratégia *bottom-up*. Nesse método, as amostras começam em seu próprio *cluster* e são sucessivamente agrupadas a medida que se sobe na hierarquia. O algoritmo termina quando todas as amostras pertencem a um único *cluster*. Uma vantagem desse algoritmo, quando comparado com outras técnicas conhecidas (e.g, K-Means), é que ele não requer o número inicial de *clusters* como dado de entrada.

O método de variação mínima de Ward é empregado para selecionar o melhor par de *clusters* a serem mesclados em cada etapa do algoritmo de clusterização hierárquica. O par escolhido busca minimizar a soma das distâncias quadradas entre as amostras e o centróide do *cluster*. A distância ao quadrado é definida como [Legendre and Legendre 2012]:

$$E_K^2 = \sum_{k=1}^{K} \sum_{i=1}^{n_k} \sum_{r=1}^{R} \left(a_{irk} - \bar{a}_{rk}\right)^2, \tag{3}$$

onde a_{irk} é a carga do fator r do UD i membro do *cluster* k, n_k é o número de UDs no *cluster* k, e $\bar{a}_{rk} = \sum_{i=1}^{n_k} a_{irk}/n_k$.

2.4. Classificação de pares usuário-dia

A partir dos *clusters* obtidos pelo conjunto de dados de treinamento usamos uma árvore de decisão para classificar novos pares UDs. Além de serem apropriadas para tarefas de classificação, as árvores também são úteis para avaliar a influência relativa dos fatores obtidos pelo PARAFAC na tomada de decisão.

A árvore possui três tipos de nós: (1) o nó raiz, representando toda a população de amostras; (2) os nós internos, expressando uma das escolhas possíveis alcançáveis naquele nível; e, (3) os nós folhas, produzindo o resultado final das decisões. Cada nó interno é associado a um critério de divisão e cada nó folha é ligado a um *cluster* (perfil diário).

No processo de construção da árvore, avalia-se o coeficiente de Gini para decidir sobre a divisão em cada nó. Ele pode ser calculado somando-se a probabilidade p_k de um

UD de *cluster* k ser escolhido vezes a probabilidade $\sum_{q \neq k} p_q = 1 - p_k$ de um erro na sua categorização. O objetivo é que essa métrica alcance o valor mínimo possível, para que os nós sejam minimamente impuros/heterogêneos. Quando apenas existirem UDs de um perfil específico em um nó, o coeficiente de Gini atinge seu valor mínimo (zero).

Com a árvore de decisão construída, podemos classificar um novo par UD. O primeiro passo consiste em aplicar novamente o método PARAFAC, agora com base nos resultados obtidos anteriormente pelo conjunto de UDs inicial. O cálculo da carga desse novo par UD_κ, definida como $\tilde{a}_{\kappa r}$, $r = 1, ..., R$, é realizado a partir da Equação 1. Na equação, os valores b_{jr} e c_{kr} são os mesmos dos obtidos para o conjunto de UDs inicial e o valor $\tilde{a}_{\kappa r}$ é computado de forma a minimizar os erros quadráticos.

2.5. Análise de comportamento dos usuários

Nos passos anteriores, obtivemos *clusters* que representam um determinado perfil de tráfego para um usuário em um determinado dia (par UD). O último passo da nossa metodologia consiste em obter a sequência de perfis de tráfego para cada usuário durante todo o período considerado. Adotamos um modelo de Markov oculto (HMM) [Rabiner 1989] com o objetivo de modelar o padrão de tráfego do usuário durante o período de um mês.

Um HMM é composto por dois processos estocásticos, um deles não é visível e é representado por uma cadeia de Markov oculta; o outro processo pode ser observado e é representado pela distribuição de probabilidade da seqüência de observações. Os parâmetros do modelo são o vetor de probabilidades inicial $\boldsymbol{\pi}$ e uma matriz de probabilidade de transição $\boldsymbol{A}$ da cadeia de Markov oculta; e a distribuição condicional de probabilidade de emissão de observações $\boldsymbol{\theta_i}$ associada a cada estado S_i da cadeia de Markov oculta. Dessa forma, um HMM é completamente determinado por $\boldsymbol{\pi}$, $\boldsymbol{A}$ e $\boldsymbol{\theta_i}$.

Definimos uma observação da HMM como o *cluster* ao qual um usuário pertence em um determinado dia. Logo uma sequência de observações corresponde a sequência de perfis de tráfego (*clusters*) de um usuário ao longo do tempo. $\boldsymbol{\theta_i}$ é a distribuição de probabilidade do perfil do usuário associado ao estado S_i, ou seja, a probabilidade do usuário pertencer a cada um dos *clusters* condicionada ao estado S_i.

A partir das sequências de perfis diários de tráfego obtidas com os dados reais de todos os usuários do nosso conjunto de dados, estimamos os parâmetros do modelo HMM usando o algoritmo de *Baum-Welch*. O modelo HMM final obtido possui diversas aplicações. Pode ser usado no planejamento de capacidade e gerenciamento da rede. Além disso, novas sequências de perfis de usuários (observações) podem ser geradas a partir do modelo permitindo fazer previsões futuras de carga da rede.

3. Resultados

Em nosso estudo uma série temporal de tráfego de download (upload) corresponde ao tráfego recebido (enviado) a cada minuto por um determinado usuário em um determinado dia conforme descrito na Seção 2. A análise é precedida por uma normalização do tráfego, em que todas as amostras são consideradas em escala logarítmica. A razão para realizar essa transformação é que tanto o tráfego de download como de upload possuem *outliers* e alta variabilidade, o que pode ter grande influência nos resultados do modelo. A aplicação da transformação pode reduzir o efeito dessas características. Também limitam-se as amostras a um máximo de dez minutos consecutivos de espaços em branco a fim de

manter um equilíbrio entre a quantidade de roteadores ativos analisados e a precisão das técnicas de modelagem empregadas, geralmente sensíveis à ausencia de dados.

3.1. Análise de fatores e clusterização

Validamos o modelo PARAFAC com quatro fatores ($R = 4$). Usamos como medida de acurácia a porcentagem de variância explicada e obtivemos o valor de $98,55\%$. O algoritmo usado para obter a solução do modelo PARAFAC é o método dos mínimos quadrados alternantes (ALS). Para obtenção de um resultado confiável, realizamos 100 inicializações randômicas do modelo. Escolhemos o modelo cuja soma do erro quadrático possui o menor valor.

A Figura 3 ilustra o modelo final da análise fatorial. Os quatro valores das cargas (*loadings*), um para cada fator e para cada uma das UDs modeladas estão na Figura 3(a). Esse resultado indica a intensidade do tráfego de download e upload dos UDs, isto é, para um determinado usuário-dia. Na Figura 3(b) temos os valores dos *loadings* para o tipo de tráfego e na Figura 3(c) são apresentadas os *loadings* de cada minuto para cada um dos quatro fatores no período de um dia.

Um dos fatores (de cor rosa no gráfico) nitidamente não está associado a horários de uso da rede para os tráfegos de download e upload. Os três fatores restantes estão associados ao uso mais intenso da rede em diferentes períodos do dia (madrugada (azul), tarde (verde) e noite (cinza)) e, apesar de serem presentes em ambos os tipos de tráfego, possuem valores mais altos para o download. A partir da observação dos valores dos *loadings* para um UD, é possível identificar o padrão de uso da rede deste UD. Por exemplo, se o valor do *loading* de um UD para o fator cinza for superior ao valor dos outros fatores, esta é uma indicação que esse usuário gera tráfego maior no período da noite para o dia correspondente ao par UD.

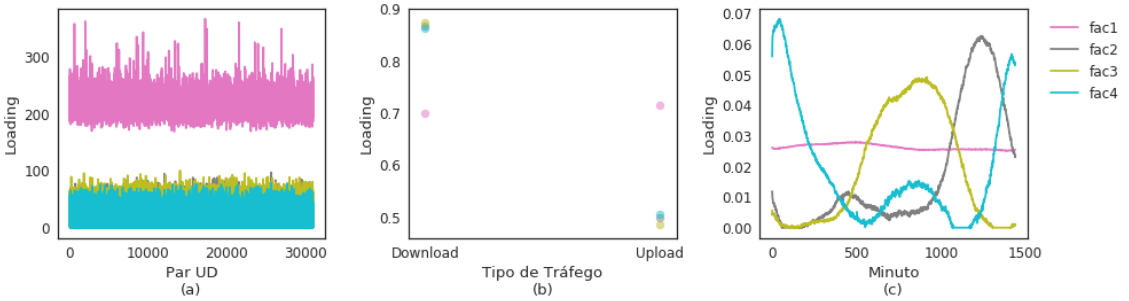

Figura 3. PARAFAC aplicado ao nosso conjunto de dados de treinamento

Na próxima etapa, executamos o algoritmo de clusterização hierárquica aglomerativa. A métrica de similaridade é a carga dos UDs para cada fator, ou seja, o vetor de cargas $\mathbf{a_i} = (a_{i1}, a_{i2}, a_{i3}, a_{i4})$ de cada UD_i. Devido aos valores das cargas variarem em até três ordens de grandeza, aplicamos a normalização Min-Max. Essa normalização altera a escala das cargas para $[0, 1]$ com base nos valores mínimo e máximo de cada fator.

A partir do dendograma obtido, decidimos selecionar cinco *clusters*. A Figura 4 mostra a mediana do tráfego de download e de upload por minuto para todos os UDs de cada *cluster*. A figura mostra o resultado da clusterização pelo conjunto de treinamento

(linha sólida) e da classificação pelo conjunto de testes (linha tracejada). Este último é explicado na próxima subseção.

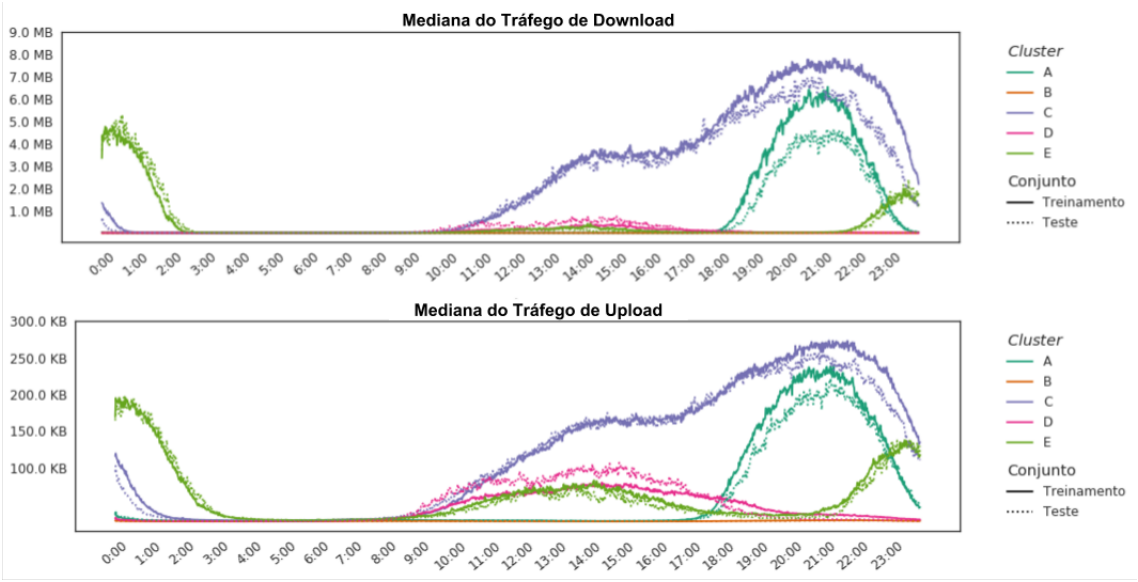

Figura 4. Mediana do tráfego de download e upload por minuto para todos os UDs de cada *cluster*: conjuntos de treinamento e de teste

Cada *cluster* representa um perfil de uso residencial da Internet. O *Cluster* C (roxo) agrupa UDs com o maior tráfego, concentrado entre a tarde e a noite. Em contraste, o *Cluster* B (laranja) é composto de UDs com o menor tráfego durante as 24h do dia. As UDs do *Cluster* A (verde escuro) são caracterizadas por uma alta demanda de banda à noite e, do *Cluster* D (rosa), das 10 às 18 horas. O *Cluster* E (verde claro) se distingue do *Cluster* D por apresentar uma demanda maior da rede durante a madrugada. Observe que o *Cluster* E apresenta padrão temporal similar ao Fator 4 do modo minuto ilustrado na Figura 3(c). É importante notar que existe correlação temporal entre o tráfego de download e upload. Se o valor da mediana do tráfego de download para um determinado *cluster*/período do dia é alto, o valor da mediana do tráfego de upload para esse mesmo *cluster*/período do dia também será alto.

3.2. Classificação e análise de perfis

Um dos objetivos do nosso trabalho é classificar um novo UD em um dos perfis definidos pela análise fatorial e clusterização. O primeiro passo é calcular as cargas do modo UD para esse novo par UD_κ utilizando o PARAFAC. Em seguida, classificamos a nova UD pelo vetor de cargas obtido, $\tilde{\boldsymbol{a}}_{\boldsymbol{\kappa}}$, seguindo o caminho de decisão proposto pela árvore ilustrada na Figura 5.

Podamos a árvore de modo a alcançar uma configuração generalizável para a classificação de novos UDs, sem realizar *overfitting* nos dados. Após a poda, a precisão da árvore é de 84% para o conjunto de treinamento. Selecionamos um grupo de 31.017 e 9.890 UDs para os conjuntos de treinamento e teste, respectivamente, totalizando 40.907 UDs. A quantidade de UDs em cada *cluster* de ambos os conjuntos estão organizados na Tabela 1.

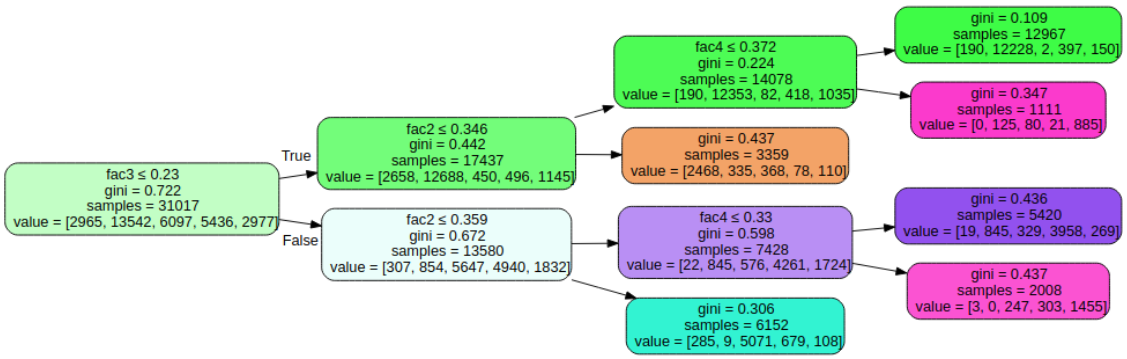

Figura 5. Árvore de decisão obtida a partir do conjunto inicial de UDs.

Tabela 1. Quandidade de UDs em cada cluster

	Cluster A	*Cluster B*	*Cluster C*	*Cluster D*	*Cluster E*
Treinamento	2.965	13.542	6.097	5.436	2.977
Teste	1.649	3.637	2.462	1.312	830

Como a nossa análise é **não supervisionada**, verificamos o resultado da classificação de forma visual pela Figura 4. As linhas tracejadas mostram a mediana do tráfego de download e de upload por minuto para as séries de teste classificadas. A mediana do tráfego para ambos os conjuntos de treinamento e teste é muito semelhante ao longo do dia. Resultados parecidos foram obtidos usando outros algoritmos de classificação como aquele que associa cada UD ao *cluster* de centróide mais próximo (i.e, *Nearest Centroid*) ou pela regra dos k vizinhos mais próximos (i.e, *k-Nearest Neighbor*).

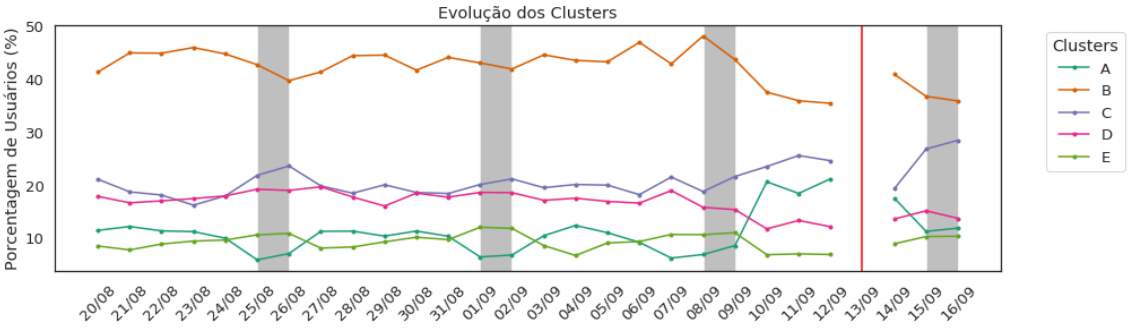

Figura 6. Evolução dos *clusters* ao longo dos 28 dias

É interessante observar como o perfil diário dos usuários evolui ao longo dos 28 dias. Na Figura 6 é apresentado o percentual de usuários pertencentes a cada um dos *clusters* para cada dia. Para facilitar a interpretação, colocamos no gráfico um fundo cinza nos finais de semana. Não existem pares UDs no dia 13 de setembro, provavelmente por algum erro no servidor de coleta de dados. O *Cluster B* é predominante em qualquer um dos dias, indicando que aproximadamente 45% dos usuários gera pouco tráfego na rede em qualquer período do dia.

Além disso, pode-se observar um comportamento diferenciado entre os dias da

semana e os finais de semana. A porcentagem de usuários pertencentes ao *Cluster* A (uso maior da rede a noite) durante a semana diminui ao mesmo tempo que a porcentagem de usuários no *Cluster* E (uso maior da rede na madrugada) aumenta durante os finais de semana e no feriado do dia 7 de setembro. Em geral, os usuários que trabalham durante a semana só conseguem acessar a rede doméstica durante a noite, entre 19h e 23h. Nos finais de semana, por outro lado, uma quantidade maior de usuários passa a utilizar a Internet de forma mais intensa a partir das 23h. No entanto, reparamos que não são necessariamente os mesmos usuários que migram do *Cluster* A para o *Cluster* E.

Por fim, existe uma mudança de comportamento na última semana do nosso conjunto de dados. O acesso a rede parece se tornar mais intenso, já que a proporção de usuários do *Cluster* B diminui e a de usuários dos *Clusters* A e C aumenta consideravelmente. Não se sabe ao certo que evento causou esse comportamento, mas acreditamos ter relação com o atentado a um dos candidatos a presidência ocorrido no dia 6/9, possivelmente pela procura por notícias imediatamente após o feriado.

Na última etapa do nosso trabalho caracterizamos as sequências dos perfis diários dos usuários através de um HMM. Escolhemos um modelo com cinco estados ocultos. Os parâmetros $\boldsymbol{\pi}$ e $\boldsymbol{A}$ estimados a partir dos dados reais estão na Tabela 2. A Figura 7 apresenta $\boldsymbol{\theta_i}$, ou seja, a distribuição de probabilidade do perfil diário do usuário associada ao estado S_i, $i = 1, ..., 5$.

Tabela 2. Modelo de Markov oculto com cinco estados

de/para	$s0$	$s1$	$s2$	$s3$	$s4$
início (π)	$0,1548$	$0,2018$	$0,3416$	$0,1141$	$0,1878$
$s0$	$0,9670$	$0,0207$	$0,0065$	$0,0034$	$0,0024$
$s1$	$0,0134$	$0,9791$	$0,0000$	$0,0046$	$0,0029$
$s2$	$0,0131$	$0,0000$	$0,9725$	$0,0080$	$0,0064$
$s3$	$0,0142$	$0,0078$	$0,0208$	$0,9550$	$0,0023$
$s4$	$0,0050$	$0,0106$	$0,0101$	$0,0062$	$0,9680$

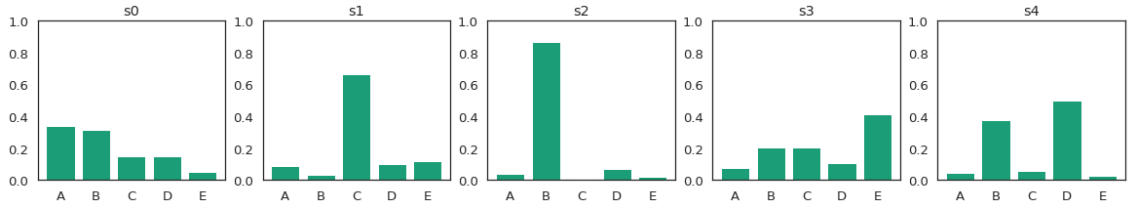

Figura 7. Distribuição de probabilidade do perfil diário para cada estado do HMM

A partir da Tabela 2, observamos que os usuários costumam se manter com uma probabilidade alta no mesmo estado do HMM. Esse resultado indica que usuários tendem a manter perfis diários de tráfego específicos que estão associados a um determinado

estado do HMM. Por exemplo, usuários que se mantém na maior parte do tempo no estado S_2 têm o seu perfil diário de tráfego melhor caracterizado pelo *Cluster* B, ou seja, são usuários que geram pouco tráfego. Esse tipo de resultado permite que os ISPs realizem um planejamento mais adequado das suas redes com base na provável demanda diária dos usuários.

Aplicamos o algoritmo de *Viterbi* para todos os usuários do conjunto de dados visando obter a sequência de estados mais prováveis do HMM, a partir dos perfis diários obtidos pela clusterização. Na Figura 8, mostramos os resultados obtidos para um pequeno subconjunto de usuários. Cada linha corresponde a um usuário; as cores, ao estado do HMM; e, a letra dentro de cada quadrado, ao *cluster* ao qual o usuário está associado naquele dia. Pode-se notar que os três primeiros usuários se mantém no mesmo estado do HMM durante as 4 semanas. O primeiro usuário, por exemplo, se mantém no estado S_4 durante os 28 dias. No estado S_4 existe uma alta probabilidade do perfil diário de tráfego ser o associado aos *Clusters* B e D, indicando um usuário que gera pouco tráfego na rede. Outros usuários podem mudar de estado ao longo do tempo (quarta e quinta linhas da Figura 8).

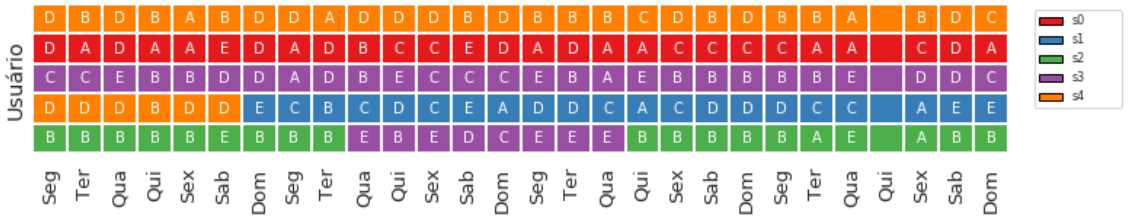

Figura 8. Sequência de estados do HMM e perfil diário de tráfego para um grupo de usuários.

A Tabela 3 ilustra o percentual de usuários que mudam de estado no HMM. Verifica-se que a maioria dos usuários se mantêm no mesmo estado durante todo o período de coleta. No caso de mudança de estado, a maior parte dos usuários realiza apenas uma transição de estado em 28 dias. Além disso, a chance de um usuário mudar de estado em determinado dia da semana é aproximadamente uniforme, sendo pouco mais elevada no sábado, o que indica uma chance maior do usuário mudar de perfil de tráfego no final de semana. Esse resultado novamente demonstra que os ISPs podem utilizar o HMM obtido para a caracterização e a previsão dos perfis de demanda diária dos usuários ao longo do tempo.

Tabela 3. Percentual de usuários que mudam de estado no HMM

% de usuários que mudam de estado do HMM x vezes	$x = 0$	$x = 1$	$x = 2$	$x = 3$			
	75,3552	20,9389	3,6442	0,0618			
% de mudanças de estado por dia da semana	segunda	terça	quarta	quinta	sexta	sábado	domingo
	13,9130	13,9130	12,1739	15,0000	13,6957	16,9565	14,3478

4. Conclusão e Trabalhos Futuros

Neste trabalho propomos um framework simples para detectar estruturas temporais e padrões comportamentais da atividade de tráfego de usuários residenciais. O framework

é composto por um conjunto de técnicas não supervisionadas de aprendizado de máquina. Dessa forma, mostramos como é possível extrair com eficiência características relevantes de nosso conjunto de dados, sem pré-rotular os dados e preservando a privacidade dos usuários.

Em suma, apresentamos um modelo do perfil diário de usuários residenciais com base nas séries temporais de download e upload. Para isso, usamos uma técnica de decomposição de tensores (PARAFAC) para capturar fatores interpretáveis e intrínsicos ao nosso conjunto de dados. Esses fatores sugerem padrões diários comuns de tráfego ao longo do dia.

Também propomos um modelo do perfil de comportamento de usuários residenciais em períodos maiores que um dia. Elaboramos um modelo de Markov oculto (HMM) a partir das sequências de perfis diários dos usuários obtidas a partir do modelo PARAFAC. O modelo final obtido indica que os usuários tendem a manter um padrão específico ao longo do tempo.

Uma possível continuação deste trabalho é a análise dos modelos propostos e dos padrões de comportamento dos usuários em períodos de férias e festividades (e.g, natal e ano novo). Dentre as aplicações em potencial citamos, por exemplo: (a) o uso do modelo HMM (generativo) em simulações e estudo de cenários para avaliar o impacto na rede devido ao aumento de usuários com determinados perfis; (b) auxílio nas tarefas de planejamento de capacidade e gerenciamento da rede; (c) relacionar a topologia da rede aos perfis de tráfego identificados a fim de se estudar possíveis correlações com outras medidas de desempenho, como perda e latência.

Um exemplo de aplicação do item (b) é a possibilidade dos ISPs melhorarem a utilização da sua rede, mesclando no mesmo conjunto de recursos os usuários cujos períodos de maior consumo ocorram em intervalos de tempo separados. Além disso, o conhecimento pelos ISPs sobre o comportamento dos usuários pode auxiliar na definição de políticas tarifárias baseadas em horários.

Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001. Este trabalho é parcialmente suportado por projeto de cooperação MCTIC-RNP/NSF, MCTIC/FAPESP, e ainda por projetos do CNPq e FAPERJ.

Referências

Bro, R. (1997). Parafac. tutorial and applications. *Chemometrics and intelligent laboratory systems*, 38(2):149–171.

Crovella, M. and Krishnamurthy, B. (2006). *Internet measurement: infrastructure, traffic and applications*. John Wiley & Sons, Inc.

Fumo, A., Fiore, M., and Stanica, R. (2017). Joint spatial and temporal classification of mobile traffic demands. In *INFOCOM*, pages 1–9. IEEE.

Harshman, R. A. (1984). "how can i know if it's real?"a catalogue of diagnostics for use with three-mode factor analysis and multidimensional scaling. *Research methods for multimode data analysis*, pages 566–591.

Harshman, R. A. and Lundy, M. E. (1984). The parafac model for three-way factor analysis and multidimensional scaling. *Research methods for multimode data analysis*, 46:122–215.

Kaiser, H. F. (1958). The varimax criterion for analytic rotation in factor analysis. *Psychometrika*, 23(3):187–200.

Kim, J., Hwang, J., and Kim, K. (2016). High-performance internet traffic classification using a markov model and kullback-leibler divergence. *Mobile Information Systems*, 2016.

Kroonenberg, P. M. (1983). *Three-mode principal component analysis: Theory and applications*, volume 2. DSWO press.

Kruskal, J. (1983). Multilinear methods. In *Proc. Symp. Appl. Math*, volume 28, page 75.

Legendre, P. and Legendre, L. (2012). Numerical ecology. 3rd. *Elsevier*.

Lorenzo-Seva, U. and Ten Berge, J. M. (2006). Tucker's congruence coefficient as a meaningful index of factor similarity. *Methodology*, 2(2):57–64.

Morichetta, A. and Mellia, M. (2018). Lenta: Longitudinal exploration for network traffic analysis. In *ITC*.

Nguyen, T. T. and Armitage, G. (2008). A survey of techniques for internet traffic classification using machine learning. *IEEE Communications Surveys & Tutorials*, 10(4):56–76.

Rabanser, S., Shchur, O., and Günnemann, S. (2017). Introduction to tensor decompositions and their applications in machine learning. *arXiv preprint arXiv:1711.10781*.

Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

Sidiropoulos, N. D., De Lathauwer, L., Fu, X., Huang, K., Papalexakis, E. E., and Faloutsos, C. (2017). Tensor decomposition for signal processing and machine learning. *IEEE Transactions on Signal Processing*, 65(13):3551–3582.

Smilde, A., Bro, R., and Geladi, P. (2005). *Multi-way analysis: applications in the chemical sciences*. John Wiley & Sons.

Soysal, M. and Schmidt, E. G. (2010). Machine learning algorithms for accurate flow-based network traffic classification: Evaluation and comparison. *Performance Evaluation*, 67(6):451–467.

Stedmon, C. A. and Bro, R. (2008). Characterizing dissolved organic matter fluorescence with parallel factor analysis: a tutorial. *Limnology and Oceanography: Methods*, 6(11):572–579.

Trevisan, M., Giordano, D., Drago, I., Mellia, M., and Munafo, M. (2018). Five years at the edge: Watching internet from the isp network. In *Proceedings of the 14th International Conference on Emerging Networking EXperiments and Technologies*, CoNEXT '18, pages 1–12.

Wright, C., Monrose, F., and Masson, G. M. (2004). Hmm profiles for network traffic classification. In *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*, pages 9–15. ACM.