Detecção de anomalias em redes baseada em medições de QoS e rótulos de QoE com ruído

Gustavo H. A. Santos[1], **Gabriel Mendonça**[1],
Rosa M.M. Leão[1], **Edmundo de Souza e Silva**[1]

Universidade Federal do Rio de Janeiro (UFRJ), Rio de Janeiro, RJ.

{gustavo,gabriel,rosam,edmundo}@land.ufrj.br

Resumo. *Detectar anomalias em redes é essencial para a manutenção de uma boa qualidade de serviço (QoS) e de experiência (QoE). No entanto, rótulos para o treinamento de modelos supervisionados são de difícil obtenção. Propomos um método para detectar anomalias baseado em um modelo estatístico que leva em consideração medições de QoS e rótulos de QoE com ruído para inferir a qualidade de uma rede de acesso residencial. Estimamos os parâmetros do modelo utilizando o algoritmo Expectation-Maximization (EM) e correlacionamos espacialmente os resultados para localizar áreas na rede com problemas de desempenho. Mostramos que o nosso modelo é eficaz utilizando um dataset real com medidas coletadas por 6369 roteadores residenciais durante 18 meses.*

Abstract. *Network anomaly detection is essential for maintaining a good quality of service (QoS) and a good quality of experience (QoE). However, it is often hard to obtain labels to train supervised models. We propose a method to detect anomalies based on a statistical model that takes into account QoS measurements and noisy QoE labels to infer the quality of residential access networks. We estimate the model parameters using the Expectation-Maximization (EM) algorithm and we correlate the results spatially to locate network regions with performance issues. We show that our model is effective using a real dataset that contains measures collected from 6369 home-routers during 18 months.*

1. Introdução

Detectar anomalias em uma rede (eventos que fogem do padrão esperado) é essencial para que seja oferecido um serviço de qualidade adequada aos clientes de um provedor (ISP). Todavia, a detecção automática de anomalias não é uma tarefa trivial. Em particular, é notório o problema da falta de rótulos identificando anomalias [Chandola et al. 2009]. Para lidar com a falta de rótulos, muitos trabalhos recorrem à inspeção manual, um procedimento restrito a anomalias conhecidas e que requer um esforço substancial [Chandola et al. 2009]. A criação manual de rótulos é baseada na avaliação subjetiva de experts, ou seja, anomalias desconhecidas ou anomalias difíceis de se detectar visualmente podem ser rotuladas de maneira incorreta. Outros trabalhos consideram dados sintéticos ou emulados, que podem não representar de maneira precisa o comportamento de anomalias reais, i.e., modelos baseados em dados sintéticos podem não ser representativos de cenários reais.

Registros de reclamações de clientes ao *call center* podem ser considerados como uma alternativa para a identificação de anomalias. Além de ser uma informação disponível

para o ISP, reclamações ao *call center* estão frequentemente associadas a eventos de rede devido a uma piora na qualidade de experiência (QoE) percebida pelo reclamante. Entretanto, um usuário da rede pode não realizar um chamado quando um problema ocorre, e vice-versa [Hu et al. 2020]. Um evento que leve a uma piora na qualidade do serviço (QoS) pode não ser notado pelos clientes que não fazem uso da rede no momento. Ao mesmo tempo, um cliente pode ter uma qualidade de experiência ruim por motivos não relacionados à QoS da rede de acesso do ISP por exemplo em decorrência de falhas em dispositivos domésticos ou problemas de conectividade sem fio. Os registros de reclamações por si só não são adequados para a detecção de anomalias quando utilizados como rótulos de métodos de aprendizado supervisionado, devido ao ruído intrínseco a eles.

Propomos neste trabalho um novo método para detectar e localizar anomalias de rede a partir de séries temporais de medições de QoS (no exemplo apresentado, essas são estatísticas de perda de pacote) e rótulos de QoE com ruído (neste caso, registros de reclamações ao *call center*). Mostramos que os parâmetros de nosso modelo podem ser estimados a partir de um conjunto de dados usando o algoritmo de EM. Em seguida, aplicamos um algoritmo de correlação espacial que permite a localização de equipamentos de rede com problemas potenciais a partir de medições de múltiplos usuários.

Aplicamos nosso método a um dataset real coletado em um ISP de médio porte contendo séries temporais de medições de perda de pacotes, realizadas durante 18 meses, por milhares de roteadores residenciais espalhados pela rede do provedor. Além disso, rótulos de QoE com ruído foram extraídos de uma base de dados de chamados ao *call center*. Apresentamos dois exemplos de anomalias reais detectadas pelo nosso método, envolvendo mudanças na configuração da rede e falhas de equipamentos. Além disso, mostramos que quando uma anomalia é detectada pelo nosso modelo uma reclamação irá ocorrer com probabilidade 4.5 vezes maior do que quando nenhuma anomalia é detectada.

O trabalho é organizado como se segue. A Seção 2 apresenta os trabalhos relacionados. Nossa metodologia é detalhada na Seção 3. Apresentamos nossos resultados na Seção 4, enquanto a Seção 5 conclui o trabalho.

2. Trabalhos Relacionados

Trabalhos de detecção de anomalia em redes podem ser categorizados de acordo com o mecanismo utilizado para a obtenção de rótulos que identificam anomalias, ou de acordo com o tipo de rede para o qual o método foi projetado. Mecanismos para a obtenção de rótulos encontrados na literatura incluem inspeção manual, geração de anomalias sintéticas e utilização de chamados ao *call center*. Além disso, métodos de detecção de anomalia são comumente aplicados a redes de *datacenters* ou a redes de acesso residenciais.

Rótulos precisos e representativos de anomalias são, em geral, difíceis de se obter [Chandola et al. 2009]. No contexto de redes de computadores, muitos métodos se baseiam em inspeção manual [Silveira and Diot 2010, Lakhina et al. 2005]. No entanto, a criação manual de rótulos é custosa, propensa a erros e dependente do conhecimento de *experts*. Por conseguinte, anomalias desconhecidas podem não ser identificadas. Além disso, muitas anomalias podem ser difíceis de detectar a partir de inspeção visual [Lakhina et al. 2004]. Outros trabalhos consideram anomalias sintéticas simuladas através de uma distribuição de probabilidade arbitrária [Xie et al. 2018, Tan et al. 2019].

No entanto, anomalias simuladas são artificiais e bem comportadas, ou seja, resultados obtidos em cenários sintéticos podem não refletir o desempenho dos métodos em um ambiente real. Em [Hu et al. 2020, Jin et al. 2010, Song et al. 2011] são utilizados chamados ao *call center* para o treinamento de modelos de detecção e/ou localização de anomalias. No entanto, estes métodos ignoram o ruído associado a chamados durante a modelagem, considerando métodos supervisionados que utilizam chamados ao *call center* como rótulos.

Trabalhos recentes na literatura consideram a detecção de anomalias a partir de medições de QoS em *datacenters* [Tan et al. 2019, Peng et al. 2017, Herodotou et al. 2014]. No entanto, estes métodos se baseiam em premissas válidas no contexto de *datacenters* que podem ser violadas no contexto de redes de acesso residenciais, como por exemplo: alta capacidade de transmissão da rede, existência de caminhos redundantes entre pares de *hosts*, simetria da topologia e a suposição de que a rede apresenta poucas falhas. Por outro lado, trabalhos da literatura com foco em redes residenciais se baseiam em atributos específicos de um determinado tipo de rede, como HFC [Hu et al. 2020], DSL [Jin et al. 2010] e IPTV [Song et al. 2011].

Nosso trabalho considera uma base de chamados ao *call center* para a obtenção de rótulos com ruído, e o nosso método é capaz de detectar anomalias que degradam a qualidade de experiência sem o custo extra associado à criação manual de rótulos. Consideramos no nosso trabalho um dataset real, com medições de QoS coletadas durante um longo período de tempo. Propomos um método que lida com o ruído associado a chamados de maneira explícita através de um modelo estatístico e utilizamos o algoritmo de EM para estimar os parâmetros deste modelo. Nossa abordagem é flexível e pode ser adotada em qualquer tipo de tecnologia de acesso: exemplificamos a aplicação do nosso método utilizando resultados de medições de perda de pacotes ICMP, ou seja, nossa técnica é agnóstica a tecnologia de acesso subjacente.

Um trabalho anterior do nosso grupo [Streit et al. 2021] propõe uma abordagem para detecção não-supervisionada de anomalias baseada em tensores, mas não considera um método automático de localização de falhas. Por fim, este trabalho estende o nosso trabalho anterior [Santos et al. 2019] em diversos pontos. Consideramos neste trabalho um novo modelo de detecção que utiliza a informação de chamados durante a estimativa dos parâmetros do modelo e que não exige a escolha de hiperparâmetros adotados em [Santos et al. 2019]. Além disso, aplicamos o modelo proposto a um dataset de longa duração com vários meses de dados e milhares de usuários. Mostramos na seção 4 que o método proposto neste trabalho é mais eficiente na detecção de eventos que impactam a qualidade de experiência dos usuários quando comparado ao método de [Santos et al. 2019].

3. Metodologia

Descrevemos nesta seção nosso método para detecção e localização de anomalias utilizando medições de QoS e rótulos de QoE com ruído. Na Seção 3.1, detalhamos nosso modelo para inferência da qualidade latente da rede. Na Seção 3.2, descrevemos nosso algoritmo para localização de anomalias com base na correlação dos resultados de usuários próximos geograficamente.

3.1. Modelo de Detecção de Anomalia

3.1.1. Descrição do Modelo

Nossa abordagem para detecção de anomalias se baseia no conceito de rótulos com ruído (*noisy labels*) [Natarajan et al. 2013]. Desejamos classificar um conjunto de dados utilizando um rótulo Y. No entanto, o rótulo Y é uma variável latente: só é possível observar uma versão imprecisa denotada por $\tilde{Y}$. Se Y é um rótulo latente binário, podemos descrever o rótulo observável com ruído $\tilde{Y}$ a partir da probabilidade condicional

$$P[\tilde{Y} = \neg a \mid Y = a] = y_{a\neg a}; a \in \{0, 1\} \tag{1}$$

O rótulo observado $\tilde{Y}$ com ruído pode ser o inverso do rótulo latente de interesse Y com uma probabilidade que depende do valor de Y. Uma análise teórica para o cenário de classificação binária com ruído descrito acima pode ser encontrada em [Natarajan et al. 2013].

O objetivo do nosso modelo é inferir o valor de uma variável latente Y relacionada a uma série temporal de observações $O = (o_1, o_2, ..., o_T)$ de tamanho T. Além de O, observamos $\tilde{Y}$, uma versão com ruído da variável latente de interesse. Nosso método utiliza a distribuição conjunta das variáveis observáveis O e $\tilde{Y}$ para realizar o ajuste de um modelo capaz de inferir a variável latente Y.

Como O é uma série temporal, é de nosso interesse capturar a dependência temporal entre as amostras. Para isso, modelamos O utilizando Cadeias de Markov Ocultas (HMMs) [Rabiner 1989, de Souza e Silva et al. 2011]. Para cada observação o_t, associamos uma variável aleatória latente q_t identificando o estado oculto (latente) no instante de tempo t.

Descrevemos nosso modelo na Figura 1 como um *Probabilistic Graphical Model* (PGM) [Bishop 2006]. As variáveis aleatórias observáveis são representadas por círculos preenchidos (em azul), enquanto variáveis latentes são representadas por círculos vazios. Uma seta (aresta) indica uma relação de dependência entre 2 variáveis aleatórias. Assumindo que nosso *dataset* possui R amostras, cada variável aleatória é replicada R vezes, o que é indicado através da notação de placas (*plate notation*) e do acréscimo do sub/super índice r.

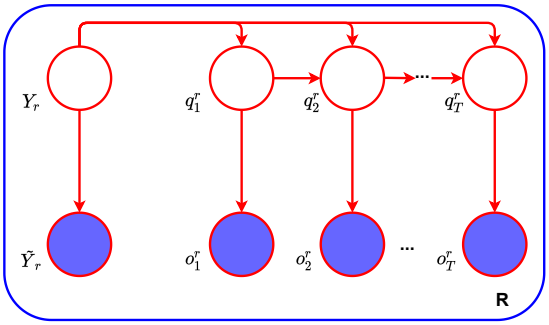


Figura 1. PGM representando os relacionamentos entre as variáveis aleatórias.

Podemos observar na Figura 1 que a variável latente Y_r afeta as distribuições da série temporal observável $O^r = (o_1^r, o_2^r, ..., o_T^r)$ e do rótulo com ruído observável $\tilde{Y}_r$. Desta

forma, a distribuição condicional das variáveis observáveis O^r e $\tilde{Y_r}$ muda de acordo com o valor da variável latente Y_r. O PGM também representa um Modelo Oculto de Markov (HMM) obtido a partir das séries temporais observadas. Desta forma, para cada observação o_t^r, há uma variável latente q_t^r indicando o estado oculto no tempo t. A correlação entre a série temporal observável O^r e o rótulo com ruído observável $\tilde{Y_r}$, capturada pelo modelo, permite a inferência do valor da variável latente de interesse Y_r a partir de um conjunto de amostras. É importante observar que os rótulos com ruído $\tilde{Y_r}$ apresentam uma granularidade diferente das medições de QoS. No nosso dataset, cada medida de QoS observável é coletada a cada minuto, enquanto uma amostra de $\tilde{Y_r}$ é coletada a cada dia.

Descrevemos a seguir as distribuições associadas a cada variável aleatória do modelo quando o aplicamos ao nosso *dataset*. Nossas séries temporais de QoS são obtidas a partir de medições contínuas de perda de pacotes coletadas a cada minuto por roteadores residenciais de um ISP de médio porte com o qual fizemos parceria. Motivados pelos padrões circadianos apresentados em nossas medições, realizamos a divisão de medições de diferentes residências em R séries temporais diárias independentes. Denotamos cada série temporal como um *par Residência-Dia*, ou *par RD*. Consideramos séries temporais diárias com amostras a cada minuto, perfazendo um total de $T = 1440$ observações. Essas amostras são obtidas pela contagem do número de pacotes perdidos em um trem de 100 pacotes transmitidos por cada roteador (maiores detalhes são apresentados na Seção 4.1). Portanto, a variável aleatória o_t^r segue uma distribuição categórica, onde uma amostra o_t^r indica o número de pacotes perdidos no tempo t para o par RD r. Utilizamos HMMs com um número discreto de estados ocultos. Os chamados técnicos realizados ao *call center* constituem os rótulos com ruído utilizados pelo nosso modelo. Desta forma, é natural definir $\tilde{Y_r}$ como uma variável aleatória indicadora com distribuição de Bernoulli, onde $\tilde{Y_r} = 1$ quando um chamado sobre um problema técnico é associado ao par RD r e $\tilde{Y_r} = 0$ caso contrário. Motivado pela definição de $\tilde{Y_r}$, definimos que a variável latente Y_r assume o valor $Y_r = 1$ quando a rede apresenta degradação suficiente para aumentar a probabilidade de chamado (informalmente, chamamos de "qualidade ruim"). Caso contrário, $Y_r = 0$ (informalmente, "qualidade boa").

Note que o modelo gráfico não define a distribuição das variáveis aleatórias, descrevendo apenas os relacionamentos entre elas. Desta forma, o PGM apresentado é geral e poderia ser aplicado em qualquer problema em que uma série temporal de observações O^r e um rótulo com ruído $\tilde{Y_r}$ são relacionados através de uma variável latente Y_r. Por exemplo, quando consideramos modelos capazes de extrair o relacionamento entre métricas de QoS e métricas de QoE, O^r pode representar uma série temporal de qualquer métrica de QoS como latência, perda e vazão (*throughput*), enquanto $\tilde{Y_r}$ pode representar qualquer métrica de QoE categórica ou binária em particular, como um *score* de qualidade obtido a partir de avaliações subjetivas.

3.1.2. Estimativa dos Parâmetros do Modelo

Descrevemos a seguir um algoritmo para estimar os parâmetros do modelo a partir de um conjunto de amostras. O PGM (Figura 1) indica um conjunto de variáveis aleatórias latentes e um conjunto de variáveis aleatórias observáveis. Desta forma, podemos ma-

ximizar a verossimilhança do modelo utilizando o algoritmo Expectation-Maximization (EM) [Dempster et al. 1977]. Em outras palavras, podemos encontrar os parâmetros que maximizam a probabilidade de observar nosso conjunto de amostras. Apresentamos nesta seção uma breve descrição do algoritmo proposto.

Considerando as distribuições das variáveis aleatórias definidas na Seção 3.1.1, o modelo apresenta os seguintes parâmetros a serem estimados:

$$\begin{aligned} p_{yl} &\triangleq P[Y_r = l] \quad y_{ml} \triangleq P[\tilde{Y_r} = m | Y_r = l] \quad \pi_{kl} \triangleq P[q_1^r = k | Y_r = l] \\ a_{jkl} &\triangleq P[q_t^r = j, q_{t+1}^r = k | Y_r = l] \quad b_{jol} \triangleq P[o_t^r = o | q_t^r = j, Y_r = l] \end{aligned} \tag{2}$$

Para estimar estes parâmetros é necessária a definição de um conjunto de variáveis auxiliares relacionadas à distribuição das variáveis latentes:

$$\begin{aligned} \alpha_{rtl}(j) &\triangleq P[o_1^r, o_2^r, ..., o_t^r, q_t^r = j | Y_r = l] \quad \beta_{rtl}(j) \triangleq P[o_{t+1}^r, o_{t+2}^r, ..., o_T^r | q_t^r = j, Y_r = l] \\ \gamma_{rtl}(j) &\triangleq P[q_t^r = j | O^r, Y_r = l] \quad \xi_{rtl}(j,k) \triangleq P[q_t^r = j, q_{t+1}^r = k | O^r, Y_r = l] \\ \tau_{rl} &\triangleq P[Y_r = l | O^r, \tilde{Y_r}] \end{aligned} \tag{3}$$

Utilizamos o algoritmo EM para estimar os parâmetros do modelo (2) a partir das variáveis auxiliares (3). No passo de *Expectation* o valor de $\gamma_{rtl}(j)$ é calculado a partir do algoritmo de Forward-Backward [Rabiner 1989] e τ_{rl} (definido acima) é dado por:

$$\tau_{rl} = \frac{\sum_{j=1}^{N} \alpha_{rTl}(j) y_{ml} p_{y_l}}{\sum_{u=0}^{1} \sum_{j=1}^{N} \alpha_{cTu}(j) y_{mu} p_{y_u}} \tag{4}$$

No passo de *Maximization* é realizada a otimização de cada parâmetro de interesse em função das variáveis auxiliares. Utilizando o algoritmo EM, obtemos:

$$\begin{aligned} p_{y_l} &= \frac{C_l}{C} \qquad y_{ml} = \frac{\sum_{r=1}^{R} \tau_{rl} \mathbb{I}(\tilde{Y_r} = m)}{C_l} \qquad \pi_{kl} = \frac{\sum_{r=1}^{R} \tau_{rl} \gamma_{r1l}(k)}{C_l} \\ a_{jkl} &= \frac{\sum_{r=1}^{R} \sum_{t=1}^{T-1} \tau_{rl} \xi_{rtl}(j,k)}{\sum_{r=1}^{R} \sum_{t=1}^{T-1} \tau_{rl} \gamma_{rtl}(j)} \qquad b_{jol} = \frac{\sum_{r=1}^{R} \sum_{t=1}^{T} \tau_{rl} \gamma_{rtl}(j) w_{rto}}{\sum_{r=1}^{R} \sum_{t=1}^{T} \tau_{rl} \gamma_{rtl}(j)} \end{aligned} \tag{5}$$

Onde definimos $C_l = \sum_{r=1}^{R} \tau_{rl}$.

3.1.3. Inferência da Qualidade de Rede Latente

Dado um par RD r e sua série temporal de medições de QoS O^r, o objetivo final do nosso modelo é inferir a qualidade de rede latente Y_r para possibilitar a detecção automática de anomalias de rede. Após a construção do modelo utilizando o algoritmo de EM (Seção 3.1.2), o processo de inferência é simples, usando a regra de Bayes:

$$P[Y_r = i|O^r] = \frac{P[O^r|Y_r = i]P[Y_r = i]}{\sum_{j=0}^{1} P[O^r|Y_r = j]P[Y_r = j]}, \quad (6)$$

onde $P[O^r|Y_r = i]$ é obtido através do algoritmo *Forward* [Rabiner 1989]. A partir da Equação 6, é possível calcular a probabilidade de ocorrência de um evento que degrada a qualidade da rede dada uma série temporal de medições de QoS, isto é, $P[Y_r = 1|O^r]$.

3.2. Correlação espacial

Descrevemos nesta seção o método de correlação espacial utilizado para identificar quais equipamentos de rede são afetados por uma anomalia de rede. Adotamos uma abordagem de votação majoritária baseada nos resultados do modelo de detecção, similar à adotada em [Santos et al. 2019].

Detalhamos a seguir as premissas consideradas pelo método proposto. Assumimos que as rotas entre cada ponto de medição e o servidor de medição são estáticas durante o intervalo da medição, uma vez que este é o caso de nosso *dataset* real[1]. Consequentemente, as rotas na rede podem ser representadas por uma árvore onde cada nó E representa um ou mais equipamentos de rede. Considera-se ainda que uma degradação de desempenho causada pelo nó E afeta todas as residências cujas rotas na rede passam por E, isto é, todos os clientes conectados a nós da subárvore de raiz E.

A correlação é baseada nos resultados obtidos pelo modelo descrito na Seção 3.1. Utilizamos o modelo para calcular a probabilidade de que cada residência observe uma qualidade de rede "ruim" (Equação 6). Em seguida, associamos a cada par RD $= r$ o valor mais provável para a qualidade de rede latente Y_r, denotando esta estimativa através da variável $\hat{Y_r}$. Desta forma, quando $P[Y_r = 1|O^r] > P[Y_r = 0|O^r]$ obtemos $\hat{Y_r} = 1$. Caso contrário, $\hat{Y_r} = 0$.

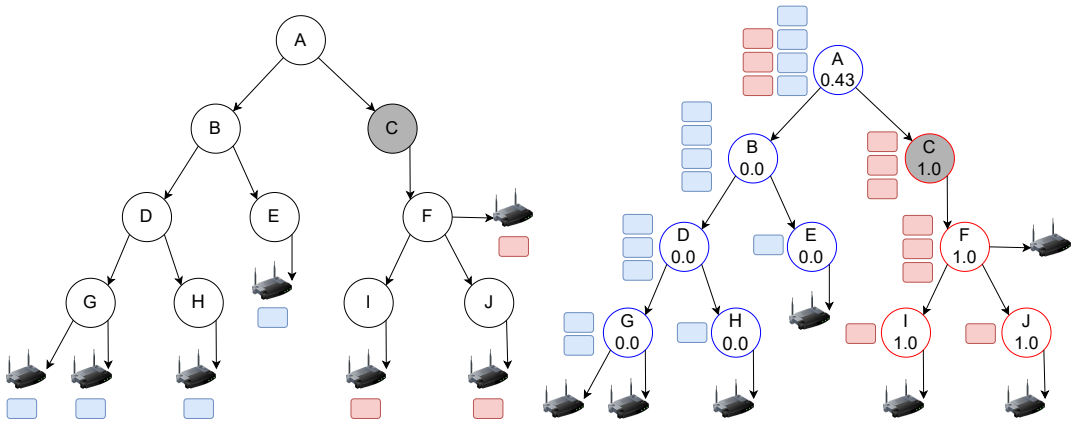
Para correlacionar os resultados de diferentes residências, usamos a qualidade de rede estimada $\hat{Y_r}$ como um "voto" para a qualidade da rede observada na rota entre a residência e o servidor de medição. A partir desta interpretação, utilizamos um mecanismo de votação simples para localizar os nós afetados por uma anomalia. Realizamos o processo de votação em cada nó da árvore, onde os votos associados ao nó E são dados pelas residências conectadas à subárvore com raiz E.

Diferentes métodos de votação podem ser considerados, conforme descrito em [Parhami 1994]. Utilizamos neste trabalho um mecanismo de votação majoritária, um mecanismo simples que apresenta bons resultados (como mostrado na Seção 4.2.2). Desta forma, o algoritmo considera que o nó E apresenta degradação de desempenho se a maioria das residências cujas rotas passam por E experimenta uma qualidade "ruim" de acordo com nosso modelo. A Figura 2 exemplifica a aplicação do método de correlação espacial.

4. Resultados

Nesta seção apresentamos os resultados obtidos quando aplicamos o método proposto a um conjunto de dados reais. Na Seção 4.1 descrevemos o *dataset* real obtido a partir

[1]O método pode ser facilmente estendido para o caso de rotas dinâmicas associando probabilidades para cada rota [Herodotou et al. 2014].



(a) Estado inicial: cada ponto de medição estima a qualidade de sua rede

(b) Processo de votação identifica degradação de desempenho a partir dos votos

Figura 2. Correlação espacial utilizando votação majoritária. Quadrados vermelhos indicam degradação detectada ($\hat{Y}_r = 1$), enquanto quadrados azuis indicam ausência de problema ($\hat{Y}_r = 0$). O nó C (em cinza) apresenta falha. O algoritmo de correlação identificou a subárvore cuja raiz é C como uma região potencialmente afetada.

de uma parceria com um ISP de médio porte. A avaliação do método é apresentada na Seção 4.2.

4.1. *Dataset*

Nosso dataset é formado por medições "ativas" feitas a partir de roteadores domésticos do ISP parceiro. Esses roteadores tem *firmware* customizado e, a cada minuto, um trem de 100 pacotes ICMP é enviado a um servidor de coleta localizado dentro da rede do ISP. Cabe ressaltar que o ISP não aplica nem filtragem nem priorização de pacotes ICMP. O intervalo entre pacotes de um mesmo trem é de 10 milissegundos de forma a capturar períodos curtos de congestionamento, e ainda sem onerar recursos da rede mesmo em *links* de menor capacidade. O trem de pacotes permite medir rajadas de perda a cada coleta. Note que a aplicabilidade do método não se limita ao uso de pacotes ICMP para a coleta. Outros métodos de coleta também podem ser utilizados. Além disso, de acordo com [Wenwei et al. 2007], medições de perda realizadas por ICMP e por TCP possuem desempenho similar. O dataset utilizado inclui o resultado de coleta de 6369 roteadores residenciais durante mais de 1 ano. Dividimos os resultados de medições de cada residência em séries temporais diárias, denotadas *pares Residência-Dia* (*par RD*), onde cada par RD possui até $T = 1440$ amostras.

Através da parceria com o ISP tivemos acesso a uma base de dados com a topologia do ISP e a localização dos roteadores realizando medições. Utilizamos estas informações durante a aplicação do algoritmo de correlação espacial proposto (descrito na Seção 3.2). Também tivemos acesso a uma base de dados de chamados técnicos realizados ao *call center*. Utilizamos a informação desta base para associar a cada par RD um rótulo indicando a presença (ou ausência) de um chamado técnico realizado ao *call center* (como descrito na Seção 3.1). Ressaltamos que a nossa análise não é baseada em nenhuma informação pessoal sobre usuários do ISP: realizamos a correlação de IDs de roteadores residenciais com um ID anônimo de usuário para identificar quais pares RD

são associados a chamados ao *callcenter*. Ressalta-se que as informações obtidas a partir da parceria com o ISP são sensíveis, o que dificulta a divulgação deste conjunto de dados.

Consideramos um conjunto de dados coletado por 18 meses entre 16/01/2020 e 30/06/2021. Dividimos o *dataset* em três partes: um conjunto de treinamento, contendo todos os pares RD coletados entre 16/01/2020 e 31/03/2020, um conjunto de validação, com dados entre 01/04/2020 e 30/06/2020, e um conjunto de teste, com dados entre 01/07/2020 e 30/06/2021. Utilizamos o conjunto de treinamento para o ajuste do modelo (Seção 4.2.1), o de validação para comparar o modelo proposto com um modelo da literatura e avaliamos os resultados do nosso modelo utilizando o conjunto de teste (Seção 4.2.2).

Para reduzir o ruído durante o processo de treinamento do modelo, consideramos apenas pares RD com ao menos 1000 amostras válidas no conjunto de treinamento. Não filtramos séries temporais dos conjuntos de validação e teste com o objetivo de deixar a avaliação de resultados mais próxima da aplicação real. Após a filtragem, o conjunto de treinamento contém 282252 pares RD, o conjunto de validação contém 444341 pares RD e o conjunto de teste contém 1727574 pares RD. O tráfego residencial pode afetar o resultado de medições de perda realizada nos roteadores [Sundaresan et al. 2011]. Desta forma, filtramos as medições em *bins* de tempo com tráfego residencial maior do que $\theta = 2.5$ Mbps, onde θ foi escolhido usando como base a menor capacidade nominal dentre as residências do *dataset*. Para reduzir a quantidade de parâmetros do modelo, utilizamos um único símbolo para codificar amostras de perda maiores do que 20 pacotes. E como períodos de indisponibilidade de rede levam a amostras faltantes, codificamos *bins* de tempo sem medições com um símbolo especial.

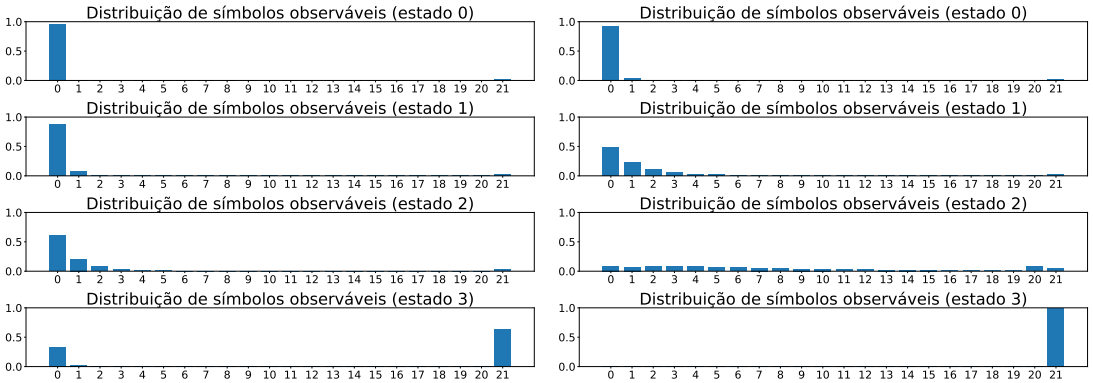
4.2. Avaliação do Método Proposto

Apresentamos a seguir os resultados obtidos utilizando nosso *dataset* real de longa duração. Primeiro, avaliamos se a probabilidade de um usuário fazer uma reclamação ao *call center* é maior quando sua qualidade de experiência é ruim (Seção 4.2.1). Em seguida, usamos nosso modelo de correlação espacial para verificar se a fração de RDs em que há chamados aumenta quando uma anomalia de rede afeta a qualidade de experiência de um grande número de residências e comparamos nosso método com [Santos et al. 2019] (Seção 4.2.2). Por fim, mostramos exemplos de anomalias reais detectadas pelo nosso método (Seção 4.2.3). A aplicação-exemplo deste trabalho realiza detecção de problemas a cada dia porque os rótulos com ruído obtidos do ISP tem granularidade diária. Entretanto, ressaltamos que o método poderia detectar problemas em escalas de tempo menores, desde que a escala de tempo dos rótulos obtidos fosse inferior a um dia.

4.2.1. Modelo de Detecção

Conforme descrito na Seção 3.1.1, as observações de perda de pacotes e de chamados ao *call center* são correlacionadas em um único modelo, e as observações de perda condicionadas ao valor da variável latente Y podem ser representadas por uma HMM. A Figura 3 mostra as distribuições das variáveis observáveis de perda de pacotes condicionadas ao valor da variável latente (não observável) Y para o modelo obtido após o treinamento

(relembramos que Y é a variável do modelo associada à QoE percebida pelos usuários, isto é, boa ou má qualidade). Ressaltamos que as distribuições mostradas na Figura 3 foram encontradas pelo modelo de maneira automática. É possível observar que perdas são mais raras para o modelo associado a boa QoE (Figura 3(a)). Além disso, eventos de indisponibilidade são mais prováveis no modelo associado a uma QoE ruim (Figura 3(b)). A partir do modelo é possível observar que a maioria dos pares RDs observam uma rede de boa qualidade (isto é, $P[Y = 0] > P[Y = 1]$). Note que a probabilidade de um usuário realizar um chamado é maior quando a qualidade de rede é pior, como esperado (isto é, $P[\tilde{Y} = 1|Y = 1] > P[\tilde{Y} = 1|Y = 0]$).



(a) Modelo obtido quando $Y = 0$. $P(Y = 0) = 0.8208$ e $P(\tilde{Y} = 1|Y = 0) = 0.0026$

(b) Modelo obtido quando $Y = 1$. $P(Y = 1) = 0.1792$ e $P(\tilde{Y} = 1|Y = 1) = 0.0116$

Figura 3. Modelo de detecção obtido a partir do conjunto de treinamento. Minutos de indisponibilidade são representados pelo símbolo 21

4.2.2. Resultados da Correlação Espacial

A partir do modelo temporal que foi treinado, é possível estimar Y_r, isto é, a QoE de cada residência em um dia, utilizando as medições observáveis naquele dia. A estimativa de Y_r em um dia e para cada residência pode então ser correlacionada espacialmente com o objetivo de localizar quais regiões e/ou equipamentos de rede apresentam problemas que afetam a QoE de usuários. Os resultados do método de correlação espacial da Seção 3.2 são apresentados a seguir.

Um equipamento com problemas (por exemplo, problemas de congestionamento ou de *hardware*) pode afetar os usuários localizados na subárvore ligada ao equipamento. Esses usuários, por sua vez, podem ou não perceber o problema. Desta forma, definimos todas as residências na subárvore de um nó marcado como problemático pela correlação espacial da Seção 3.2 como *potencialmente afetadas* (PA) por um problema. Relembramos que um nó é marcado como problemático quando a maioria dos votos de roteadores em sua subárvore detecta degradação de QoE, como explicado na Seção 3.2. O conjunto complementar a PA, considerando como universo todas as residências da árvore em observação, é definido como $\overline{\text{PA}}$.

Seja p_g (p_b) a probabilidade de um RD pertencente ao conjunto $\overline{\text{PA}}$ (PA), fazer um chamado técnico ao *call center* e $\hat{p}_g$ ($\hat{p}_b$) os respectivos estimadores dessas probabilida-

des. Caso o nosso algoritmo tenha um bom desempenho para detectar e localizar falhas, esperamos que $\hat{p}_b$ seja maior do que $\hat{p}_g$, ou seja, que as reclamações ao *call center* sejam mais frequentes para os pares *Residência-Dia* pertencentes ao conjunto PA.

Realizamos a estimativa de $\hat{p}_g$ e $\hat{p}_b$ para cada mês do período de Julho de 2020 a Junho de 2021. Para todos os meses considerados, o estimador $\hat{p}_b$ associado a RD's potencialmente afetados por um problema de rede é uma ordem de grandeza maior do que o estimador $\hat{p}_g$ associado a RD's não afetados ($\overline{\text{PA}}$). Para comparar os estimadores $\hat{p}_b$ e $\hat{p}_g$ aplicamos um teste de hipótese para comparação de proporções de duas populações [Montgomery and Runger 2010] com nível de significância $\alpha = 0.05$. Em todos os casos, podemos rejeitar a hipótese nula $p_b = p_g$ em favor da hipótese alternativa $p_b > p_g$. Desta forma, nossa evidência indica que a probabilidade de um usuário realizar um chamado é maior quando nosso método detecta uma piora na qualidade da rede.

Uma métrica de interesse é a diferença entre as probabilidades p_b e p_g, que indica o quanto aumenta a probabilidade de realizar um chamado quando o algoritmo detecta uma anomalia. Para estimar a distribuição de $(p_b - p_g)$, usamos inferência Bayesiana. Modelamos o número de chamados em regiões com qualidade de rede boa / ruim usando uma distribuição binomial com parâmetro p_g (rede boa) / p_b (rede ruim). Adotamos para os 2 parâmetros uma distribuição *a priori* Beta(1, 1) (distribuição uniforme). Dado o número de RDs com rede boa / ruim em que houve chamado, a distribuição *a posteriori* de p_g e p_b também segue uma distribuição Beta (distribuição *a priori* conjugada). Portanto, podemos estimar diretamente a distribuição *a posteriori* de $(p_b - p_g)$ a partir de amostras da distribuição *a posteriori* de p_b e p_g utilizando o método de Monte Carlo. Mostramos os resultados para cada mês do conjunto de teste na Figura 4. É possível observar que a *Highest Posterior Density (HPD)* da distribuição *a posteriori* de $p_b - p_g$ nunca inclui zero, isto é, existe uma forte evidência de que a probabilidade de haver um chamado aumenta quando nosso método detecta uma degradação de qualidade na rede. Além disso, o aumento na probabilidade pode ser de até 0.018, como observado em Fevereiro de 2021. **Observação:** é importante notar que um valor de probabilidade da ordem de 10^{-2} não é insignificante, pois pode representar centenas de chamados extras diários. Por exemplo, uma região de um ISP com 10000 usuários em PA e $p_b - p_g = 0.018$ implicam numa média de 180 novos chamados em um dia!

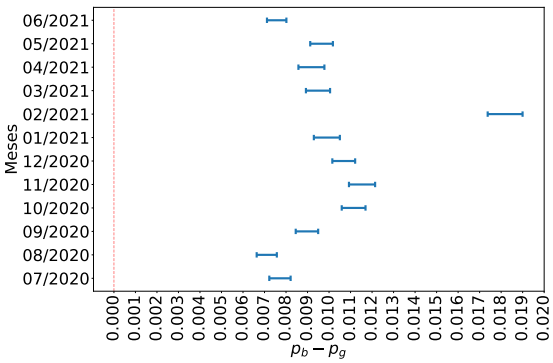


Figura 4. 95% HPD da diferença entre p_b e p_g para cada mês

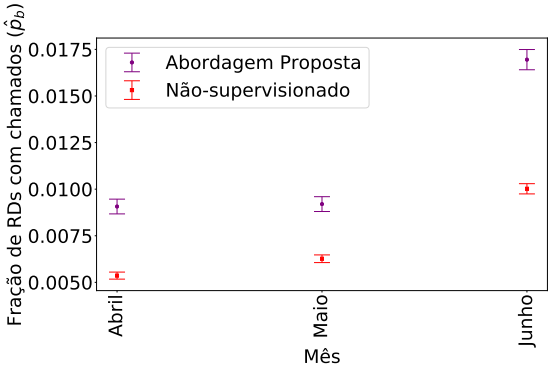


Figura 5. Comparação entre abordagem proposta e método não-supervisionado (Intervalo de Confiança de 95%)

Por fim, comparamos os resultados do modelo proposto neste trabalho com o nosso método de detecção de pontos de mudança não-supervisionado [Santos et al. 2019]. Reajustamos o modelo não-supervisionado usando os dados do conjunto de treinamento deste artigo. Em seguida, utilizamos o conjunto de validação para obter a fração de chamados em regiões da rede com degradação de QoS detectada por cada um dos métodos, isto é, comparamos o valor de $\hat{p}_b$ obtido por cada abordagem. A Figura 5 mostra que o método proposto neste trabalho apresenta desempenho superior ao método não-supervisionado, uma vez que regiões marcadas como potencialmente afetadas pelo método proposto estão associadas a uma fração de chamados ao *call center* maior do que regiões marcadas pelo método não-supervisionado em todos os meses considerados.

4.2.3. Anomalias Detectadas

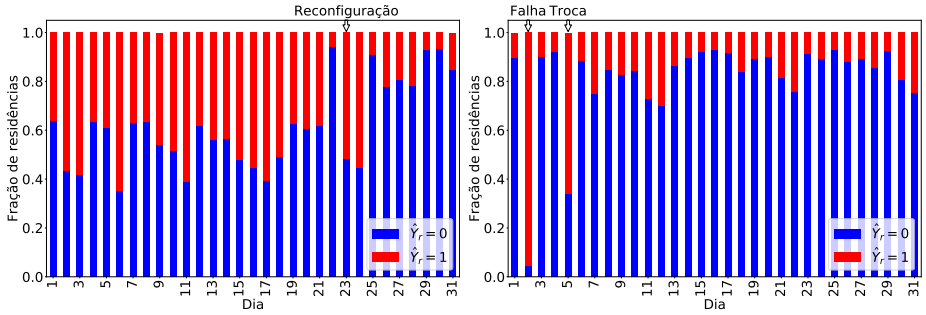
Além de detectar períodos em que a QoE dos usuários está ruim com alta probabilidade, nosso método também é capaz de identificar eventos que afetam significativamente a qualidade da rede. Apresentamos nesta seção exemplos de anomalias reais detectadas pela nossa abordagem, onde a causa raiz de cada evento foi confirmada pelo ISP.

Mudanças na configuração da rede: A Figura 6(a) mostra a fração de residências nas quais, a cada dia, nosso algoritmo estimou uma QoE boa ou ruim durante o mês de maio de 2020 para uma determinada região da rede. Os resultados mostram que o nosso método detectou uma QoE ruim para uma quantidade considerável de residências até o dia 24 (entre 40% e 60% das residências aproximadamente). Após esta detecção entramos em contato com o ISP parceiro, que reportou que um processo de reconfiguração tinha sido iniciado nesta região no dia 23 visando melhorar a QoS da rede. Nosso método foi capaz de detectar este evento automaticamente: é possível observar que o percentual de residências com QoE ruim diminui para a faixa de 10% a 20%, aproximadamente, a partir do dia 23.

Falha de equipamento: A Figura 6(b) mostra os resultados obtidos pelo nosso método em uma outra região da rede durante o mês de Agosto de 2020. É possível perceber que a fração de usuários cujo algoritmo estimou boa QoE é em geral alta, acima de 80% aproximadamente, durante a maior parte do tempo. No entanto, para uma fração significativa de usuários, nosso algoritmo detectou uma QoE ruim nos dias 2 e 5. O ISP reportou que um equipamento de rede apresentou falha elétrica no dia 2, causando indisponibilidade na rede durante algumas horas. Este equipamento foi trocado no dia 5, e a rede também apresentou indisponibilidade durante o período de manutenção. Ambos os eventos foram identificados automaticamente pelo nosso método.

5. Conclusão

Propomos neste trabalho um novo método de detecção de anomalia utilizando medições de QoS e rótulos de QoE com ruído. O nosso método é baseado em um modelo estatístico que relaciona medições de QoS com rótulos de QoE observados para inferir a distribuição da qualidade de rede latente. Propomos um modelo cujos parâmetros são estimados pelo método Expectation-Maximization e utilizamos este modelo para detectar anomalias a partir das medições de QoS realizadas por roteadores domésticos. Com base neste modelo, elaboramos um método de correlação espacial para identificar quais equipamentos



(a) Evento 1: Reconfiguração da rede iniciada no dia 23

(b) Evento 2: Falha em equipamento no dia 2. Equipamento é trocado no dia 5

Figura 6. Exemplos de anomalias reais detectadas pelo nosso método

de rede são os prováveis causadores da anomalia. Os resultados obtidos a partir de um dataset real de medições coletadas por milhares de roteadores residenciais durante 18 meses mostram que o método é capaz de detectar e localizar eventos que impactam negativamente a QoE dos usuários e que é ainda capaz de estimar a probabilidade de ocorrerem chamadas técnicas ao *call center*. Por fim, apresentamos exemplos de anomalias reais detectadas pelo método.

Agradecimento: Este trabalho é parcialmente suportado por projeto de cooperação MCTIC-RNP/NSF, MCTIC/FAPESP, e projetos do CNPq e FAPERJ, além de bolsas CAPES.

Referências

[Bishop 2006] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg.

[Chandola et al. 2009] Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):1–58.

[de Souza e Silva et al. 2011] de Souza e Silva, E., Leão, R. M. M., and Muntz., R. R. (2011). Performance evaluation with hidden markov models. In *Performance Evaluation of Computer and Communication Systems. Milestones and Future Challenges*, pages 112–128.

[Dempster et al. 1977] Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22.

[Herodotou et al. 2014] Herodotou, H., Ding, B., Balakrishnan, S., Outhred, G., and Fitter, P. (2014). Scalable near real-time failure localization of data center networks. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1689–1698.

[Hu et al. 2020] Hu, J., Zhou, Z., Yang, X., Malone, J., and Williams, J. W. (2020). Cablemon: Improving the reliability of cable broadband networks via proactive network maintenance. In *17th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 20)*, pages 619–632.

[Jin et al. 2010] Jin, Y., Duffield, N., Gerber, A., Haffner, P., Sen, S., and Zhang, Z.-L. (2010). Nevermind, the problem is already fixed: proactively detecting and troublesho-

oting customer dsl problems. In *Proceedings of the 6th International COnference*, pages 1–12.

[Lakhina et al. 2004] Lakhina, A., Crovella, M., and Diot, C. (2004). Diagnosing network-wide traffic anomalies. *ACM SIGCOMM computer communication review*, 34(4):219–230.

[Lakhina et al. 2005] Lakhina, A., Crovella, M., and Diot, C. (2005). Mining anomalies using traffic feature distributions. *ACM SIGCOMM computer communication review*, 35(4):217–228.

[Montgomery and Runger 2010] Montgomery, D. C. and Runger, G. C. (2010). *Applied statistics and probability for engineers*. John Wiley & Sons.

[Natarajan et al. 2013] Natarajan, N., Dhillon, I. S., Ravikumar, P., and Tewari, A. (2013). Learning with noisy labels. In *NIPS*, volume 26, pages 1196–1204.

[Parhami 1994] Parhami, B. (1994). Voting algorithms. *IEEE transactions on reliability*, 43(4):617–629.

[Peng et al. 2017] Peng, Y., Yang, J., Wu, C., Guo, C., Hu, C., and Li, Z. (2017). detector: a topology-aware monitoring system for data center networks. In *2017 {USENIX} Annual Technical Conference ({USENIX}{ATC} 17)*, pages 55–68.

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

[Santos et al. 2019] Santos, G. H., Mendonça, G., de Souza e Silva, E., Leão, R. M. M., Menasche, D. S., et al. (2019). Análise não supervisionada para inferência de qualidade de experiência de usuários residenciais. In *Anais do XXXVII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 958–971. SBC.

[Silveira and Diot 2010] Silveira, F. and Diot, C. (2010). Urca: Pulling out anomalies by their root causes. In *2010 Proceedings IEEE INFOCOM*, pages 1–9. IEEE.

[Song et al. 2011] Song, H. H., Ge, Z., Mahimkar, A., Wang, J., Yates, J., Zhang, Y., Basso, A., and Chen, M. (2011). Q-score: Proactive service quality assessment in a large iptv system. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pages 195–208.

[Streit et al. 2021] Streit, A., Santos, G. H., Leão, R. M., de Souza e Silva, E., Menasché, D., and Towsley, D. (2021). Network anomaly detection based on tensor decomposition. *Computer Networks*, 200:108503.

[Sundaresan et al. 2011] Sundaresan, S., de Donato, W., N.Feamster, Teixeira, R., Crawford, S., and Pescapè, A. (2011). Broadband internet performance: A view from the gateway. In *ACM SIGCOMM 2011*.

[Tan et al. 2019] Tan, C., Jin, Z., Guo, C., Zhang, T., Wu, H., Deng, K., Bi, D., and Xiang, D. (2019). Netbouncer: Active device and link failure localization in data center networks. In *16th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 19)*, pages 599–614.

[Wenwei et al. 2007] Wenwei, L., Dafang, Z., Jinmin, Y., and Gaogang, X. (2007). On evaluating the differences of tcp and icmp in network measurement. *Computer Communications*, 30(2):428–439.

[Xie et al. 2018] Xie, K., Li, X., Wang, X., Xie, G., Wen, J., and Zhang, D. (2018). Graph based tensor recovery for accurate internet anomaly detection. In *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, pages 1502–1510. IEEE.