

Detecção de Anomalias em Cenários de Cidades Inteligentes Baseados em Internet das Coisas

Ivan Zyrianoff, Fabrício O. de França, Carlos Kamienski

Universidade Federal do ABC (UFABC)
ivan.dimitry@aluno.ufabc.edu.br,
{folivetti,cak}@ufabc.edu.br

Resumo. *A implantação de aplicações baseadas em Internet das Coisas (IoT) para Cidades Inteligentes enfatiza a necessidade de soluções para detectar automaticamente eventos de interesse nesses sistemas. Alguns cenários, como iluminação pública inteligente e monitoramento de usuários em praças digitais, são exemplos que geram uma quantidade massiva de dados e anomalias necessitam ser identificadas para que ações sejam tomadas em tempo hábil, uma vez que um evento atípico impacta diretamente a população que trafega no local. Foi proposta uma arquitetura baseada em computação em névoa para viabilizar um modelo de detecção de anomalias em tempo real. O modelo foi avaliado com dados reais de praças públicas na cidade de São Paulo e com dados de sensores sinteticamente gerados, obtendo resultados satisfatórios em ambos os casos.*

Abstract. *The deployment of Internet of Things (IoT) based applications for Smart Cities emphasize the need for solutions that can automatically detect events of interest in these systems. Some scenarios, such as smart public lighting and user monitoring in the so-called digital squares, are examples that generate a massive amount of data. In these scenarios, anomalies need to be identified as fast as possible, since corresponding actions should be executed in a timely manner. An atypical event can have huge impacts on the nearby population. A fog computing based architecture has been proposed to make real-time anomaly detection feasible. The model was evaluated with real data from digital squares from the city of São Paulo and with data from synthetically generated sensors, obtaining satisfactory results in both cases.*

1. Introdução

A população mundial está fazendo a transição para a vida em cidades, no entanto a aglomeração de pessoas ocasiona diversos desafios que requerem novas soluções. O conceito de Cidades Inteligentes prevê ambientes nos quais a tecnologia da informação seja uma ferramenta para resolver alguns desses problemas [Kamienski et al. 2016]. Para isso, é necessária uma rede interconectada de dispositivos que monitorem o ambiente urbano, conhecida como de sensoriamento urbano. A Internet das Coisas (IoT) desempenha um papel fundamental na implantação de Cidades Inteligentes, viabilizando a coleta de dados de uma enorme quantidade de sensores, a análise de dados e a tomada de decisões que alteram o comportamento de sistemas em tempo real [Zanella et al. 2014]. O uso da inteligência computacional - em conjunto com IoT e Big Data - permite a alteração de comportamentos na velocidade necessária sem intervenção humana direta [Perera et al. 2014].

As inúmeras aplicações de Internet das Coisas, especialmente em ambientes de Cidades Inteligentes, geram uma quantidade massiva de dados brutos, sendo que a análise e o processamento dessas amostras são grandes desafios [Al Nuaimi et al. 2015]. Falhas nesses sistemas têm potencial de gerarem grandes perdas financeiras bem como impactar negativamente a população que circula no local. Uma potencial solução é o uso de técnicas de inteligência computacional para detectar possíveis erros e anomalias. Tradicionalmente, o processamento desses algoritmos é realizado de modo centralizado - em algum *datacenter* de nuvem - gerando um contraste com o modelo inerentemente distribuído de IoT e ocasionando diversos problemas: 1) a latência, que inviabiliza o processamento em tempo real; 2) grande ocupação da largura de banda para o tráfego de todo volume de dados; 3) alto custo de se manter um servidor que necessita de muitos recursos computacionais [Bonomi et al. 2012]. Uma potencial solução para esse problema é utilizar uma arquitetura baseada em névoa computacional, na qual a inferência é realizada em dispositivos localizados na borda da rede, e somente uma parte do processamento é realizado de maneira centralizada.

Esse artigo propõe uma arquitetura que alia computação em névoa à computação em nuvem para detecção de anomalias em Cidades Inteligentes baseadas em IoT. A inferência é realizada na borda da rede por nós da névoa computacional, enquanto o cálculo e a atualização dos parâmetros são processados de modo centralizado na nuvem. A validação do modelo foi feita em dois diferentes cenários de Cidades Inteligentes: a) Iluminação pública inteligente: cenário onde a iluminação pública é controlada via software e anomalias são erros lógicos na iluminação; b) Praças digitais na cidade de São Paulo: praças públicas que oferecem WiFi gratuito em um projeto implementado pela prefeitura, nas quais uma anomalia é mapeada como um número atípico de usuários conectados, caracterizando um evento social (e.g. manifestação política). Os dados do cenário a) foram simulados, enquanto os dados do b) são reais.

Uma anomalia em qualquer um desses cenários possui potencial de prejudicar indivíduos que estão no local, a falta de iluminação pode provocar acidentes enquanto um evento de proporções inesperadas em alguma praça pública pode trazer consequências para todos que estão próximos se o poder público não garantir a ordem e a segurança. A análise das praças digitais teve o objetivo de identificar a possibilidade de detectar eventos que geram uma grande aglomeração de pessoas, baseando-se no número de usuários conectados no ponto de acesso WiFi. Os resultados em ambos cenários foram positivos, conseguindo identificar pontos de iluminação com problemas e eventos acontecendo em tempo real na praça digital estudada, mostrando que a arquitetura e modelo proposto são genéricos suficientes para serem executados em cenários heterogêneos.

O restante do artigo está organizado como segue: na Seção 2 estão presentes os trabalhos relacionados; a Seção 3 apresenta a arquitetura proposta e o algoritmo utilizado nos experimentos; na Seção 4 são descritos os dois cenários dos experimentos; a seção 5 trata da metodologia adotada no projeto enquanto os resultados são apresentados na seção 6 e discutidos na seção 7 e por fim são apresentadas as conclusões do trabalho.

2. Trabalhos Relacionados

O uso de computação em névoa como uma tecnologia essencial para viabilizar a Internet das Coisas é um tema constante na literatura da área. No entanto, grande parte da literatura

disponível levanta questões conceituais, arquiteturas, arcabouços e cenários para a combinação destas tecnologias e apresenta poucos resultados práticos advindos de protótipos, implementações e avaliações de desempenho. Artigos publicados recentemente possuem tais características como: Dastjerdi & Buyya (2016) e Chiang & Zhang (2016), que discorrem sobre objetivos, desafios, requisitos, arquiteturas, tecnologias e cenários, porém há uma lacuna de resultados práticos. Por outro lado, Bonomi et al. (2014) apresenta casos de uso e requisitos para aplicações que combinam computação em névoa com IoT e análise de dados (*analytics*) em cenários como semáforos inteligentes e parques eólicos. Além disso, discorre sobre a interação entre a névoa e a nuvem. No entanto, não apresenta implementações nem resultados de tais casos de uso.

Yannuzzi et al. advogam que a névoa computacional é um modelo mais adequado do que a nuvem para a implementação de IoT e por isso devem ser usadas em conjunto. É apresentado um cenário de mobilidade de usuários que se beneficia da névoa e resultados de avaliação. No entanto, não são abordadas questões relacionadas à imensa quantidade de dados a serem gerados por aplicações IoT e o seu processamento distribuído e dividido entre névoa e nuvem, como propomos nesse trabalho. FogFlow [Cheng et al. 2017] é um arcabouço de execução distribuída para processar fluxos de informação entre a nuvem e a névoa incluindo a orquestração de serviços. O trabalho se concentra na arquitetura do sistema e apresenta resultados adequados de avaliação de desempenho. No entanto, não é abordado o assunto de inteligência computacional para tratar de uma massa imensa de dados e como isso se alia a arquitetura FogFlow.

Neste artigo é avaliado o processamento distribuído de dados de IoT em cidades inteligentes para duas aplicações reais e factíveis na cidade de São Paulo para identificar comportamentos normais e anômalos. Com base num projeto da prefeitura para converter 700.000 postes de iluminação em lâmpadas de LED equipadas com sensores e atuadores IoT, são gerados dados sintéticos através da ferramenta SenSE [Zyrianoff et al. 2017] para validar a técnica e a arquitetura proposta nesse artigo. Além disso, são usados dados reais de usuários conectados em praças digitais do programa WiFiLivreSP [Ratusznei et al. 2015, Ratusznei et al. 2016] para detectar aglomerações de pessoas em determinadas regiões da cidade servidas pela rede pública de WiFi Municipal.

3. Uma Arquitetura de Névoa e Nuvem para IoT

O foco desse artigo é propor uma arquitetura baseada em um modelo híbrido de computação em nuvem e névoa para a detecção de anomalias em tempo real em cenários de Cidades Inteligentes baseados em IoT. Essa arquitetura será validada em dois cenários diferentes: 1) detecção de falhas na iluminação pública inteligente; 2) identificação de eventos em praças digitais.

A Figura 1 ilustra o fluxo dados para o funcionamento básico e tradicional do modelo de detecção de anomalias. Sensores enviam informações para um ponto de acesso (*gateway*), que transmite a informação até um servidor localizado em um *datacenter* – normalmente localizado em uma nuvem computacional - onde será executado o modelo de detecção de anomalias para rotular cada nova amostra como regular ou anômala. Uma vez isso processado e enviado para o sistema competente – como por exemplo, alguma central de monitoramento - os dados são armazenados em lotes e periodicamente serão

feitas atualizações a esse modelo, com base em otimizações processadas com os dados históricos armazenados.

No entanto, essa abordagem implementada em cenário complexos de IoT em Cidades Inteligentes possui alguns problemas: 1) Centralizar todo o processamento de dados de vários dispositivos IoT gerará uma grande carga, necessitando de muitos recursos computacionais; 2) A latência para enviar os dados até um ponto centralizado fisicamente distante do lugar da geração do dado pode significar perda da resposta em tempo real; 3) Caso todas as mensagens de dispositivos IoT sejam enviadas para um único ponto para ser processado, boa parte da banda será ocupada por esses dados.

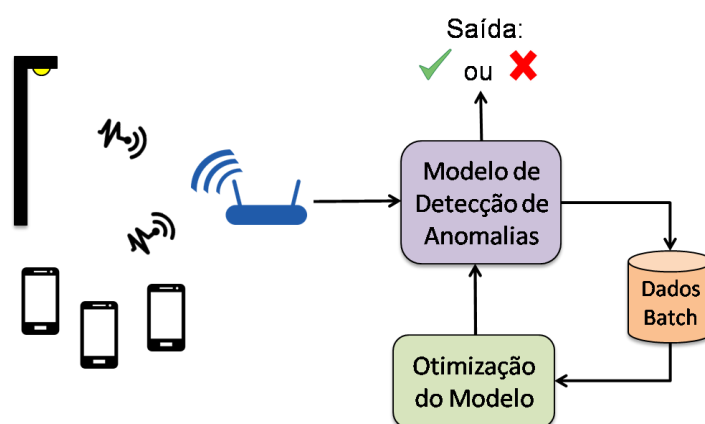


Figura 1. Fluxo de dados no modelo proposto de detecção de anomalias

Uma potencial solução para as questões levantadas é uma arquitetura baseada em computação em névoa, ilustrada na Figura 2 pelo esquema de processamento de dados nas praças públicas de São Paulo. Ao invés de centralizar todo o processamento, cada *gateway* irá atuar como um nó da névoa computacional que fará a inferência se um dado é anômalo ou não, dessa forma, garante-se o processamento da informação em tempo real, já que o dado é processado na borda da rede. Esses dispositivos não possuem muitos recursos computacionais, porém existe uma gama de algoritmos de detecção de anomalias cujo processo de rotular uma nova amostra é rápido e computacionalmente barato, no entanto, a definição dos parâmetros do modelo computacional consome tempo e recursos computacionais inviáveis para um *gateway*.

Periodicamente, os nós da névoa devem enviar informações para um controlador centralizado localizado em algum *datacenter*. Esse controlador deve armazenar os dados em *batch* de cada nó da névoa e periodicamente processar otimizações, utilizando alguma ferramenta para processamento de Big Data, como o Spark¹, e informar os novos parâmetros para o modelo de detecção de anomalias de cada nó. Portanto, os parâmetros de cada *gateway* serão automaticamente atualizados para refinar a detecção de anomalias, empregando um modelo distribuído e hierárquico de processamento da informação. O intuito é que essa arquitetura proposta seja genérica suficiente para que possa ser aplicada em diferentes cenários dentro do escopo de Cidades Inteligentes, por isso a validação da mesma foi realizada utilizando cenários que se diferem muito, apesar de ambos necessitarem de processamento em tempo real.

¹ <https://spark.apache.org/>



Figura 2. Arquitetura nuvem/névoa para detecção de anomalias, ilustrada na cidade de São Paulo

3.1. Algoritmo de detecção de anomalias

O método de detecção de anomalias utilizado nesse artigo foi o algoritmo gaussiano. O algoritmo traz bons resultados apesar de sua simplicidade, além disso o processo de rotular uma nova amostra é extremamente rápido, característica imprescindível para a arquitetura proposta. O princípio básico do método é que os atributos considerados normais se distribuem conforme uma curva normal, dessa forma, os valores que estão distantes da média por certo limiar são considerados anômalos.

A etapa de aprendizado do algoritmo utiliza apenas os dados considerados normais para definir uma região em que o comportamento está dentro do esperado. Após o treinamento, todo e qualquer ponto fora do padrão estabelecido pelo modelo é considerado anômalo. Dessa forma também não ocorre sobreajuste no modelo por falta de amostras anômalas (e esse foi um dos motivos da escolha desse algoritmo). As amostras de teste apresentam anomalias para validar o modelo gerado. O algoritmo é apresentado na Figura 3.

No cenário de computação em névoa, apenas o processo indicado como processamento em tempo real que seria computado no nó da névoa, no caso, no ponto de acesso. Esse é um processamento leve e rápido, não necessitando muitos recursos computacionais. Para o processamento em *batch* é necessário mais tempo e recursos computacionais, no entanto, esses passos são necessários somente para calcular os parâmetros utilizados no processamento em tempo real. Portanto podem ser processados em um *datacenter* localizado em alguma nuvem computacional.

Como mostra a arquitetura na Figura 2 cada nó da névoa envia dados ao banco de dados centralizado, que periodicamente atualiza o modelo de análise de anomalias com novos parâmetros, otimizando a detecção automaticamente. Essa otimização pode ser apenas o refinamento do limiar, desde a atualização da média e variância, ambos processos executados em *batch*. Portanto, apesar do algoritmo ser o mesmo para todos os pontos da cidade, cada nó da névoa terá seus próprios parâmetros únicos e otimizados para aquele local, calculados por uma aplicação centralizada, localizada em alguma nuvem computacional.

Processamento em batch :	
1	treinamento <- 80% dos dados sem anomalias
2	validaçãoCruzada <- 20% dos dados
3	média, variância <- calculaParâmetros(treinamento)
4	epsilon <- validacaoCruzada(validaçãoCruzada, média, variância)
Processamento em tempo real:	
5	para cada nova amostra x:
6	detectaAnomalia(x, media, variância)
7	calculaParâmetros (valores):
8	média <- média(valores)
9	variância <- variância(valores)
10	retorna média,variância
11	validacaoCruzada(valores, média, variância):
12	épsilon <- [10e-300 até 10e-1]
13	melhorF1 <- 0
14	para cada épsilon em epilons:
15	para cada x em valores:
16	s <- detectaAnomalia(x, média, variância, epsilon)
17	f1 <- calculaF1-score(s)
18	se f1 > melhorF1
19	melhorF1 <- f1
20	retorna melhorF1
21	detectaAnomalia(x, média, variância, épsilon):
22	calc <- Gaussiana(x, média, variância)
23	se calc > épsilon
24	retorna Anomalia Detectada
25	senão
26	retorna Amostra Normal

Figura 3. Algoritmo de detecção de anomalias

4. Cenários

Ambos os cenários escolhidos para os experimentos do algoritmo de detecção de anomalias estão dentro do escopo de Cidades Inteligentes e IoT. O cenário da iluminação pública inteligente é mais futurista, portanto, os dados foram simulados, enquanto o cenário das praças digitais já existe e os dados utilizados foram dados reais.

4.1. Iluminação Pública Inteligente

Em 2015, a prefeitura paulistana abriu uma licitação da forma de parceria público-privada (PPP) na modalidade de concessão administrativa para modernização, otimização, expansão, operação, manutenção e controle remoto e em tempo real da infraestrutura da rede de iluminação pública do município de São Paulo². Nessa licitação é descrito como um dos requisitos a detecção de falhas, a priorização de atendimentos e intervenções em tempo real. Os pontos de iluminação pública seriam equipados com sensores que enviariam informações periodicamente até uma central e também seriam atuadores, já que a sua luminosidade deveria ser controlada via software, de maneira centralizada.

Com a implementação da iluminação pública inteligente espera-se que problemas mecânicos (e.g. uma lâmpada queimada ou um poste derrubado) sejam triviais de serem identificados por meio de sensores, sem a necessidade de algoritmos sofisticados. No

²http://www.prefeitura.sp.gov.br/cidade/secretarias/upload/chamadas/edital_de_licitaco_1447363708.pdf

entanto, sistemas controlados por software estão sujeitos a falhas lógicas, desencadeadas tanto por *bugs* na programação quanto por falhas e erros de comunicação, essas anomalias são mais desafiadoras de serem detectadas e por isso que o cenário analisado pretende identificar justamente esse tipo de falha.

Baseando-se no projeto dessa licitação, modelamos um cenário utilizando dados sinteticamente gerados para detecção de falhas em pontos de iluminação em tempo real. A licitação previa o mínimo de 250.000 pontos de iluminação nesse modelo, utilizando o SenSE [Zyrianoff et al. 2017] (um gerador de tráfego IoT), foram simulados 700.000 dispositivos inteligentes de iluminação pública. A modelagem e os experimentos têm como objetivo serem uma prova de conceito que é possível utilizar algoritmos de detecção de anomalias nesse tipo de cenário, em uma arquitetura de computação em névoa e obter resultados satisfatórios.

Foi definido que cada dispositivo de iluminação pública envia mensagens utilizando o protocolo MQTT (típico de IoT) com periodicidade de 10 minutos ao *broker* MQTT fisicamente mais próximo, localizado no mesmo nó da névoa onde a mensagem será rotulada. A carga útil de cada mensagem contém um número que é o identificador único de cada dispositivo e um valor que varia de 0 até 1 que indica a luminosidade emitida por aquele ponto de iluminação.

No caso de uma falha (anomalia), a iluminação era feita de maneira incorreta, que varia desde o ponto de iluminação apagado durante a noite, até emitindo luz intensa durante o dia. Nem toda falha era extrema, ou seja, o dispositivo ficar completamente apagado ou completamente aceso, e.g. o dispositivo parcialmente aceso quando deveria estar completamente. Acreditamos que erros desse tipo são factíveis de ocorrerem em sistemas IoT complexos, como o cenário proposto.

4.2. Praças Digitais em São Paulo

Para incentivar a participação dos cidadãos e promover a inclusão digital a prefeitura de São Paulo iniciou o programa WiFiLivreSP³ em 2014, que implantou 120 praças digitais, nas quais o acesso a internet WiFi é irrestrito e gratuito a qualquer cidadão.

Os dados de conexão de usuários foram coletados durante vários anos e os principais resultados de desempenho são apresentados em Ratusznei et al. (2015) Ratusznei et al. (2016). No entanto, os dados provindos dos pontos de acesso nas praças digitais possuem potencial para trazer uma série de outras informações relevantes que os dados brutos não mostram. O proposto neste artigo é identificar a ocorrência de eventos com grande circulação de pessoas em alguma praça digital através do número de usuários conectados ao ponto de WiFi.

O algoritmo de detecção de anomalias atua no número de usuários conectados, caso essa métrica esteja fora do comportamento estabelecido pelo algoritmo como padrão, é detectada uma anomalia. Esperamos que essas anomalias sejam devido a eventos inesperados ocorrendo nesses locais, como manifestações políticas ou eventos culturais. Cada uma das praças possui um ponto de acesso, esses dispositivos possuem poucos recursos computacionais e não conseguiram executar todo o modelo. No entanto, com a arquitetura apresentada na Figura 2, o processamento que demanda mais recursos

³ http://www.prefeitura.sp.gov.br/cidade/secretarias/inovacao/inclusao_digital/index.php?p=152219

computacionais é realizado na nuvem, enquanto o ponto de acesso age como um nó da névoa computacional, que dado os parâmetros - calculados em um *datacenter* - conseguirá inferir se uma nova amostra é anômala ou não.

Para a realização dos experimentos foi escolhida apenas uma praça digital: o vão livre do MASP. Essa escolha possui duas principais motivações: 1) O vão livre do MASP é icônico para cidade de São Paulo e conhecido por ser palco de diversos eventos durante o ano, desde manifestações populares até eventos culturais; 2) Como é necessário realizar a validação do modelo, é preciso confirmar se realmente ocorreu um evento no dia que o algoritmo classifica como anômalo e o MASP recebe constante atenção da mídia, enquanto em outras praças seria mais difícil realizar tal verificação.

5. Metodologia

Em ambos os cenários, o componente da aplicação localizado na nuvem computacional foi programado e executado utilizando o Apache Spark 2.2.0, programado em Python. Enquanto, a análise de cada amostra para testes que é processada no nó da névoa foi executada por um leve *script* em Python.

5.1. Análise dos Dados

5.1.1. Cenário 1: Iluminação Pública

Foi utilizado o SenSE para simular 700.000 pontos de iluminação inteligente enviando dados, inicialmente, nenhum erro foi injetado, a fim de compor a etapa de treinamento. O treinamento do modelo foi feito com 21.399.394 mensagens de sensores de iluminação pública e tanto a etapa de validação cruzada quanto a de testes possui aproximadamente 20% do conjunto de dados usado na etapa de treinamento. Em cada conjunto de dados com anomalias, elas representavam cerca de 2% do total de dados.

Para que os dados gerados tivessem significado foi computacionalmente definido pelo simulador dois períodos, um representando o dia, em que haveria iluminação ambiente e o outro representando à noite, quando é necessária a iluminação artificial. Os histogramas para o conjunto de dados do dia e da noite são ilustrados pela Figura 4, observando-os conseguimos identificar que durante o dia os dados se aproximam mais do 0 (completamente apagado) e durante a noite aproximam-se do 1 (completamente aceso), no entanto, os dados estão distribuídos sendo que durante o período do dia, a intensidade luminosa emitida concentra-se em valores menores do que 0,4 e durante a noite observa-se que a maioria dos valores em torno de 0,7.

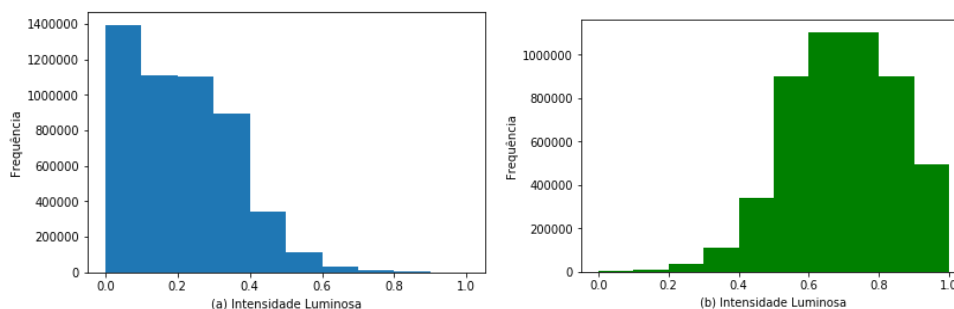


Figura 4. Intensidade luminosa; (a) período do dia (b) período da noite.

5.1.2. Cenário 2: Praças Digitais

Os dados utilizados são dados históricos que trazem informações do uso de internet no vão livre do MASP, o banco de dados utilizado está descrito no relatório da prefeitura⁴. Os dados analisados vão do dia 15/12/2014 até o dia 03/09/2016, sendo que o tempo entre duas amostras é de 10 minutos. Da massa total de dados, cerca de 60% foi utilizada para o treinamento, 20% para a validação cruzada e os outros 20% para testes.

O pré-processamento dos dados revelou que o comportamento do número de usuários conectados a praça se diferencia nos finais de semana dos dias úteis e também muda por período do dia. Portanto, não faz sentido usar os mesmos parâmetros do algoritmo de detecção de anomalias para todos os dados. Cada amostra foi classificada de acordo com o período e o dia da semana, sendo que o período foi dividido em: manhã (6h-12h), tarde (12h-17h), noite (17h-00h) e madrugada (00h-6h); e os dias da semana foram divididos em dias úteis ou finais de semana.

Também foram mapeados manualmente 81 eventos que ocorreram no período estudado no vão livre do MASP para verificar a taxa de acerto do modelo. Os histogramas das Figura 5, Figura 6 e Figura 7 ajudam a compreender a dinâmica da praça analisada.

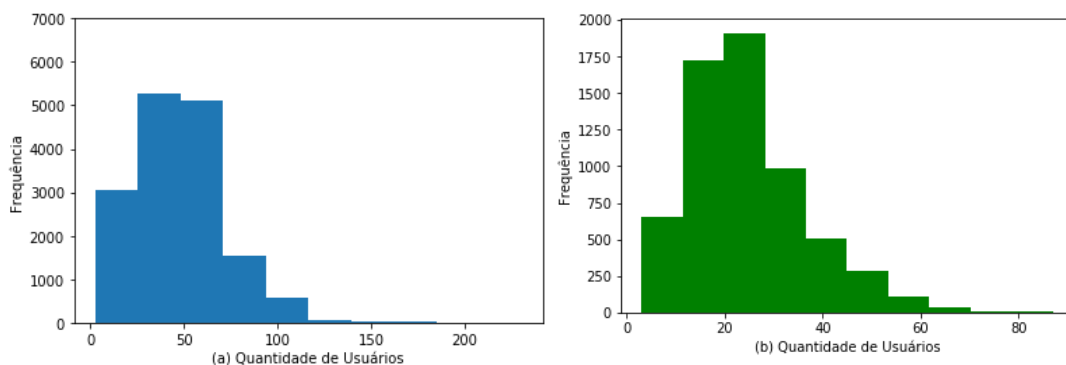


Figura 5. Número de usuários pela manhã; (a) dias úteis (b) finais de semana

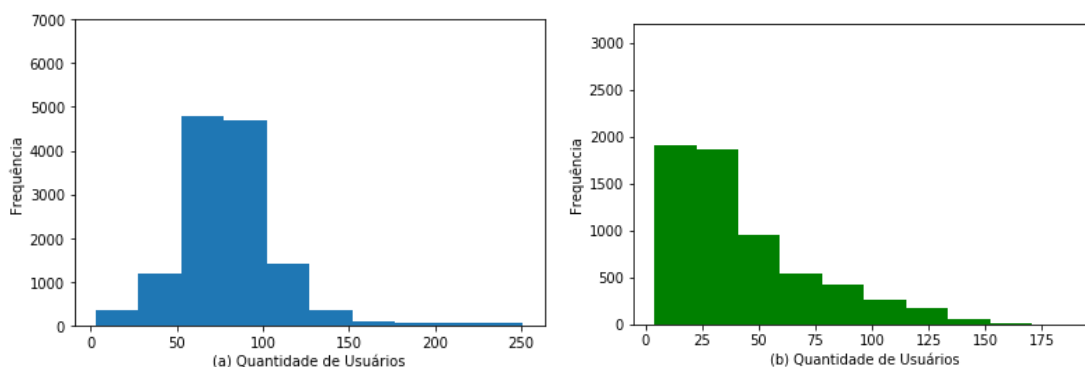


Figura 6. Número de usuários à tarde; (a) dias úteis (b) finais de semana.

⁴http://www.prefeitura.sp.gov.br/cidade/secretarias/upload/servicos/inclusao_digital/wifilivresp/relatoriosufabc/WiFi-Livre_R2-1.pdf

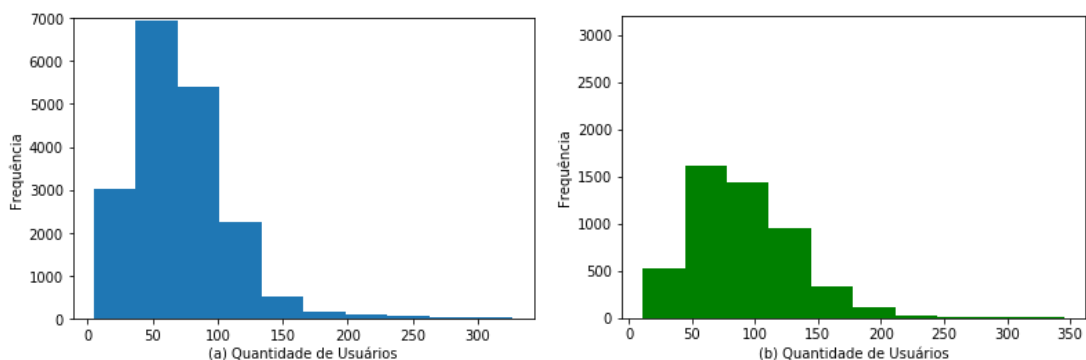


Figura 7. Número de usuários à noite; (a) dias úteis (b) finais de semana.

É possível perceber que o período com a menor quantidade de usuários são as manhãs, sendo que uso mais intenso durante os dias úteis. As tardes e noites possuem um comportamento parecido, sendo que as tardes habitualmente possuem uma maior quantidade de usuários conectados e somente durante a noite que a média de usuários aos finais de semana é maior do que durante os dias úteis.

5.2. Métricas

Foram calculadas três métricas para cada conjunto de dados para inferir a eficácia do modelo: precisão, revocação e F1-score. Elas foram escolhidas de tal forma a analisarmos o desempenho do algoritmo em relação a sua capacidade em detectar anomalias. Nesse cenário a ocorrência de falsos negativos implica em falhas que não serão corrigidas, o que é mais crítico para o sistema. A simples precisão (quantidade de acertos sobre o total de amostras) não é suficiente no contexto de detecção de anomalias, uma vez que o subconjunto de dados anômalos é muito menor do que a quantidade de amostras, portanto, um algoritmo que sempre rotula a amostra como normal teria um desempenho satisfatório. O conjunto das três métricas contribui para um entendimento completo do funcionamento do algoritmo.

A precisão mostra o desempenho geral do modelo, a revocação (quantidade de anomalias corretamente identificadas sobre o total de anomalias) é sensível a falsos negativos e avalia o quão bom o modelo é em identificar dados anômalos e o cálculo de F1-score é a média harmônica das duas métricas, calculado segundo a Equação 1.

$$F1 - score = 2 \frac{precisão * revocação}{precisão + revocação} \quad (1)$$

6. Resultados

6.1. Iluminação Pública Inteligente

Os resultados para a precisão da detecção de anomalias no cenário da iluminação pública inteligente estão presentes na Tabela 1. Como o resultados ótimo é 1, tanto o conjunto de dados referentes ao dia quanto o conjunto de dados referentes a noite obteve ótimos valores para as métricas.

Para o período em que foi simulado o dia, a revocação calculado foi 1. Isso significa que toda anomalia foi identificada, não houve falsos negativos na análise desse conjunto de dados. Ocorreram falsos positivos, já que a precisão não foi máxima, no entanto, o valor muito próximo ao melhor possível indica que a detecção de anomalias

obteve resultados satisfatórios. No período simulado como noite os resultados também são bons, porém observa-se que ocorreram falsos negativos, já que a revocação não possui o valor máximo. As precisões, bem como o F1-score, apresentam valores próximos dos melhores possíveis.

Como foram feitas simplificações na modelagem a fim de tratar esse cenário com uma maior granularidade e um nível de abstração maior, esses resultados possuem um significado maior como prova de conceito e não se espera um resultado tão bom quanto esse em um cenário real. Apesar disso, esses resultados iniciais indicam que esse algoritmo, aliado a arquitetura híbrida de computação nuvem-névoa proposta são adequados para ambientes complexos de IoT em cenários de Cidades Inteligentes.

Tabela 1. Resultados da Iluminação Pública Inteligente

Período	Precisão	Revocação	F1-Score
Iluminação natural (dia)	0,99	1,00	1,00
Iluminação artificial (noite)	0,99	0,94	0,97

6.2. Praças Digitais em São Paulo

Os resultados para o conjunto de dados referente às praças digitais, mais especificamente ao caso de estudo - o vão livre do MASP – estão presentes na Tabela 2. Esse cenário foi analisado com dados reais, apesar disso, os resultados foram extremamente positivos. Na tabela não foi incluído o resultado para as madrugadas, uma vez que não foi mapeado nenhum evento nesse período fora o tradicional ano novo na Av. Paulista.

Na maioria dos casos, todas as anomalias foram detectadas (revocação igual a 1) e mesmo assim a precisão também obteve bons resultados, mostrando que apesar da alta taxa de acerto de anomalias o algoritmo também tem uma boa tolerância a falsos positivos. A análise dos resultados indica que essa praça possui um comportamento parcialmente previsível, e que mudanças a esse comportamento usualmente significam que algum evento está acontecendo no local.

O único caso que o F1-score foi abaixo de 0,9 é para as manhãs dos finais de semana, analisando os dados e o que sabemos do comportamento dessa praça em específico, acreditamos que isso tenha acontecido devido que a partir de 28/08/2015, a av. Paulista – onde o MASP está localizado – tem o tráfego fechado para veículos e aberto para o trânsito de pedestres em alguns domingos. Nesses dias, normalmente, ocorre uma grande movimentação de pessoas pela avenida, o que provavelmente impactou também no número de usuários conectados no ponto de acesso do vão livre do MASP. O algoritmo identifica esses dias atípicos como se houvesse algum evento ocorrendo, aumentando assim a quantidade de falsos positivos.

O algoritmo identificou diversos dias que ocorreram eventos, alguns acredita-se que o poder público já tinha conhecimento que iriam ocorrer. No entanto, foram identificados eventos com grande adesão de pessoas que talvez tenham sido completamente inesperados e a identificação em tempo real desses contribuiria para que ações fossem tomadas para garantir que o evento acontecesse da melhor forma possível. Exemplos de eventos inesperados que o algoritmo identificou são: encontros promovidos por influenciadores digitais, que levaram uma grande aglomeração de pessoas para o vão livre do MASP e arredores; e diversas manifestações que, infelizmente, acabaram em

confrontos com a polícia. Caso essas manifestações fossem identificadas antes de tomarem grandes proporções, talvez confrontos poderiam ter sido evitados.

Tabela 2. Resultados da análise na praça digital no vão livre do MASP

Dia da Semana	Período	Precisão	Revocação	F1-score
Dias úteis	Manhã	0,99	1,00	1,00
Dias úteis	Tarde	0,95	1,00	0,97
Dias úteis	Noite	0,96	0,93	0,95
Finais de Semana	Manhã	0,89	0,72	0,80
Finais de Semana	Tarde	0,94	1,00	0,97
Finais de Semana	Noite	0,89	1,00	0,94

7. Discussão

A arquitetura combina o poder de processamento da nuvem com a rapidez e eficiência da névoa e foi proposta para ser implementada em ambientes baseados em IoT em Cidades Inteligentes, a arquitetura foi validada em dois cenários, um com dados reais provindos de praças digitais da cidade de São Paulo e outro com dados simulados por um gerador de tráfego para IoT. Em ambos os casos, a detecção de anomalias obteve ótimos resultados apesar da simplicidade do modelo gaussiano. A boa taxa de acerto está atrelada com o refinamento dos parâmetros, indicando que em um cenário real a atualização deve ocorrer periodicamente. Os cenários propostos e implementados são provas de conceito para verificarem que o modelo proposto de detecção de anomalias para Cidade Inteligentes possui potencial para ser explorado de modo útil.

Os resultados positivos da análise realizada no cenário da iluminação pública inteligente indicam que o algoritmo e a arquitetura proposta possuem potencial para serem implementadas em ambientes reais, porém há algumas considerações a serem feitas: 1) a granularidade trabalhada foi alta, já que se trata de uma prova de conceito. A iluminação solar não é uniforme para toda cidade, diferentes áreas recebem incidência solar diferente, há sombras e nuvens; 2) para obter maior precisão seria interessante incrementar o modelo com mais atributos, levando em consideração a posição geográfica na cidade do ponto de iluminação e a luminosidade ambiente por exemplo.

Considerando os experimentos realizados, foi possível identificar com sucesso algumas flutuações atípicas na luminosidade emitida. Análises da classificação dos dados revelaram que alguns falsos negativos na realidade são flutuações leves da intensidade luminosa, sendo que a amostra considerada anômala foi rotulada como regular. Considera-se que essa classificação seja aceitável, uma vez que o erro não é significativo.

O cenário das praças públicas, diferentemente da iluminação inteligente, fisicamente existe. Os dados analisados são reais e mesmo assim os resultados foram positivos. Indicando que é possível traçar um padrão de uso para praças e identificar em tempo real se algum evento está acontecendo. Apesar de somente o vão livre do MASP ter sido analisado, cada praça possui um padrão diferente de uso [Ratusznei et al. 2016] e possuirá parâmetros únicos para inferir anomalias. A análise dos resultados da detecção de anomalias no vão livre do MASP são referentes a época de alta instabilidade política no país, devido ao processo de impeachment da ex-presidente Dilma Rousseff, que gerou

muitas manifestações políticas na Av. Paulista Algumas dessas manifestações tiveram concentração no vão livre do MASP e foram identificadas pelo algoritmo. Alguns atos políticos são informados com antecedência e o poder público já tem o conhecimento onde e quanto irá ocorrer, mas por vezes, uma manifestação surge em resposta a fato ocorrido no dia, ou até mesmo era um ato previsto para ser pequeno que toma grandes proporções. A identificação rápida desses eventos é importante para que o poder público possa adotar medidas para garantir a segurança dos participantes. Além disso, dependendo da dimensão do evento, é possível fazer alterações no trânsito local.

Para o cálculo das métricas, foi necessário mapear os eventos que aconteceram no vão livre do MASP. No entanto, o processo de detecção de anomalias identificou eventos que a verificação manual não encontrou, pois não foram divulgados pela mídia tradicional. O exemplo mais icônico foi um desvio acentuado no dia 22/05/2015, sempre identificado como anômalo. Depois de pesquisas mais aprofundadas, descobriu-se que um grande influenciador digital marcou um encontro com seus fãs no vão livre do MASP esse dia, o que gerou uma grande aglomeração de indivíduos no local.

A análise dos dados do vão livre do MASP gerou informações relevantes a possibilidade de identificar eventos inesperados em tempo real. Os resultados dessa prova de conceito podem ser ampliados para as demais praças, uma vez que eventos sejam mapeados para ajustar os parâmetros. Outro ponto interessante foi a desclassificação de eventos, o interesse da pesquisa é identificar eventos que gerem uma grande aglomeração de pessoas, inicialmente aberturas de exposições no MASP foram considerados eventos, mas logo descartados, visto que houve pouco movimento de indivíduos no local. Também é necessário considerar que como a inferência de eventos é feita com base no número de usuários do WiFi, caso aconteça algum evento cujo os participantes não se conectem a rede, ele passará despercebido.

É importante enfatizar a necessidade de uma arquitetura híbrida nuvem-névoa para o processamento de dados em *batch* e tomada de decisão em tempo real em um ambiente geograficamente distribuído típico de uma aplicação em tempo real para Cidades Inteligentes. Decisões relacionadas com ocorrências contextuais devem ser tomadas localmente pelo nó da névoa que está próximo aos sensores, atuadores e usuários. Por outro lado, esses componentes não possuem capacidade computacional suficiente para realizar o processamento de larga escala da grande quantidade de dados necessária para gerar os parâmetros necessários para a tomada da decisão. Neste caso, temos dados históricos para os quais devem ser calculados: média, desvio padrão e limiar (épsilon); para cada uma das 120 praças digitais. Para que a cidade se torne totalmente conectada num cenário futurista serão necessárias dezenas de milhares de locais cobertos por sinal WiFi e será também necessária uma infraestrutura considerável envolvendo processamento em *batch* na nuvem computacional e análise e detecção de anomalias de maneira centralizada na névoa para gerar alarmes para agências públicas responsáveis pela vigilância e segurança.

8. Agradecimentos

Essa pesquisa foi parcialmente financiada pelo projeto SWAMP [Kamiencki et. al. 2018], uma colaboração entre Brasil e União Europeia.

9. Conclusão

O artigo apresenta uma arquitetura que alia computação em névoa e em nuvem para detectar anomalias em tempo real em ambientes de Cidades Inteligentes baseados em Internet das Coisas. A arquitetura foi validada em dois cenários, sendo um mais futurista, a iluminação pública inteligente, e o outro já existente, as praças digitais em São Paulo, em ambos os casos os experimentos obtiveram alta taxa de acertos e se mostraram adequados para a arquitetura proposta. Como trabalhos futuros deseja-se realizar experimentos com outras técnicas inteligentes, explorando o potencial de uma arquitetura distribuída na névoa computacional, além de realizar um número maior de experimentos explorando outras praças digitais além do vão livre do MASP.

Referências

- Al Nuaimi, E., Al Neyadi, H., Mohamed, N., Al-Jaroodi, J. (2015), “Applications of Big Data to Smart Cities”, *Journal of Internet Services and Applications*, 6(1), 2015.
- Bonomi, F., Milito, R., Zhu, J., Addepalli, S. (2012), "Fog Computing and its role in the Internet of Things", *IEEE Workshop on Mobile Cloud Computing (MCC)*, 2012.
- Bonomi, F., Milito, R., Natarajan, P., Zhu, J., (2014), “Fog Computing: A Platform for Internet of Things and Analytics”. *Big Data and Internet of Things: A Roadmap for Smart Environments*, Springer, 546:169–186, March 2014.
- Cheng, B., Solmaz, G., Cirillo, F., Kovacs, E., Terasawa, K., Kitazawa, A. (2017), "FogFlow: Easy Programming of IoT Services Over Cloud and Edges for Smart Cities", *IEEE Internet of Things Journal*, August 2017.
- Chiang, M., Zhang, T., (2016), "Fog and IoT: An Overview of Research Opportunities", *IEEE Internet of Things Journal*, 3(6), Dec. 2016.
- Dastjerdi, A., Buyya, R. (2016), "Fog Computing: Helping the Internet of Things Realize Its Potential", *IEEE Computer*.
- Kamienski, C., Biondi, G., Borelli, F., Heideker, A., Ratusznei, J., Kleinschmidt, J., “Computação Urbana: Tecnologias e Aplicações para Cidades Inteligentes”, *Minicursos SBRC 2016*, Maio de 2016.
- Kamienski, C., Soininen, J.P., Taumberger, M., Fernandes, S., Toscano, A., Salmon Cinotti, T., Filev Maia, R., Torre Neto, A., "SWAMP: an IoT-based Smart Water Management Platform for Precision Irrigation in Agriculture", *aceito para o Global IoT Summit 2018 (GIoTS'18)*, Junho de 2018.
- Perera, C., et. al. (2014), "Context Aware Computing for The Internet of Things: A Survey", *IEEE Comm. Surveys & Tutorials*, First Quarter 2014.
- Ratusznei, J., Silva, W., Pinheiro, N., Melo, R., Kamienski, C. (2015). *Uma Rede WiFi Aberta de Larga Escala como Infraestrutura para Cidades Inteligentes*, SEMISH 2015.
- Ratusznei, J., Zannotto, W., Kamienski, C. (2016). *Qualidade de Experiência do Usuário em uma Rede WiFi Municipal usando Streaming de Vídeo*, SEMISH 2016.
- Yannuzzi, M., Milito, R., Serral-Gracià, R., Montero, D., Nemirovsky, M. (2014), “Key ingredients in an IoT recipe: Fog Computing, Cloud Computing, and more Fog Computing”, *IEEE CAMAD Workshop 2014*.
- Zanella, A., Bui, N., Castellani, A., Vangelista, L., Zorzi, M. (2014), “Internet of Things for Smart Cities”, *IEEE Internet of Things Journal*, 01(1), Feb. 2014.
- Zyrianoff, I.; Borelli, F.; Kamienski, C., (2017), “SenSE – Sensor Simulation Environment: Uma ferramenta para geração de tráfego IoT em larga escala”, *SBRC 2017. Salão de Ferramentas*.