Um modelo para seleção automática de algoritmos de extração de eventos de trânsito para aplicações ITS

Alexandra S. Pereira[1], **Thais R.M.B. Silva** [2], **Fabrício A. Silva**[2], **Luiz H.A. Correia**[3], **Antonio A.F. Loureiro**[1]

[1] Departamento de Ciência da Computação
Universidade Federal de Minas Gerais (UFMG) – Belo Horizonte, MG – Brasil

[2]Instituto de Ciências Exatas e Tecnológicas
Universidade Federal de Viçosa (UFV) – Florestal, MG – Brasil

[3]Departamento de Ciência da Computação
Universidade Federal de Lavras (UFLA) – Lavras, MG – Brasil

alesilva241@gmail.com, loureiro@dcc.ufmg.br, lcorreia@ufla.br
{thais.braga,fabricio.asilva}@ufv.br

Abstract. *Traffic events can be useful to a variety of Intelligent Transportation System (ITS) applications. This work presents a model that can correlate features of multiple data sources with demands from ITS applications interested in consuming traffic events in order to establish the best strategy for extracting them. Once used, the model yields to a list of events, each of them reporting what happened, besides where and when. An instance of the proposed model using two social networks as data sources and four machine learning algorithms was implemented as a case study. The results have shown that it was possible to extract a great part of the expected events, all of them with complete information.*

Resumo. *Eventos de trânsito podem ser úteis para uma variedade de aplicações de sistemas de transporte inteligente (Intelligent Transportation Systems (ITS)). Este trabalho apresenta um modelo capaz de correlacionar características de múltiplas fontes de dados com demandas de aplicações ITS interessadas em consumir eventos de trânsito para estabelecer a melhor estratégia para extraí-los. Uma vez utilizado, o modelo leva a uma lista de eventos, cada um deles reportando o que aconteceu, além de onde e quando. Uma instância do modelo proposto usando duas redes sociais como fontes de dados e quatro algoritmos de aprendizado de máquina foi implementada como estudo de caso. Os resultados mostraram que foi possível detectar uma grande parte dos eventos esperados, todos com suas informações completas sobre o que, onde e quando.*

1. Introdução

Um evento pode ser definido como um acontecimento ligado a alguma anormalidade que ocorre quando, em um determinado contexto, algo foge do padrão natural ou esperado [Allan et al. 1998]. Um evento também é definido como um acontecimento específico em um intervalo de tempo e no espaço, envolvendo relacionamento e ações entre entidades [Brants et al. 2003, Galton and Augusto 2002, Kumaran et al. 2004]. Quando o espaço de ocorrência de situações de interesse se dá em vias destinadas ao uso por veículos, classificamos as mesmas como eventos de trânsito [Albuquerque et al. 2016].

Detectar eventos de trânsito pode ser considerada uma tarefa complexa. A grande quantidade de entidades envolvidas em alguns casos, a dificuldade na obtenção de dados que tem o potencial de fornecer informações necessárias à caracterização dos eventos e até mesmo a demanda pela rapidez na detecção podem ser mencionados como alguns dos principais desafios. Por outro lado, há uma enorme demanda por soluções desse tipo por Sistemas de Transporte Inteligente (*Intelligent Transportation Systems (ITS)*), uma vez que os mesmos utilizam eventos de trânsito na construção de aplicações para cidades inteligentes. Neste trabalho, consideramos que um evento de trânsito T deve ser caracterizado pelo seu tipo (*o que*), lugar (*onde*) e momento no tempo (*quando*) em que ocorreu. Em outras palavras, utilizamos uma tupla composta por estes três elementos para caracterizar um evento de trânsito ($T = \{\{o\_que\}, \{onde\}, \{quando\}\}$).

Uma das formas mais exploradas atualmente para identificação de eventos, em particular aqueles ligados ao trânsito, têm sido a aplicação de algoritmos de aprendizado de máquina sobre bases de dados coletadas em redes sociais. Até o momento, tais trabalhos apresentam soluções que normalmente consideram uma ou mais fontes de dados específicas, aplicando sempre a mesma estratégia ou algoritmo de extração às mesmas. Neste trabalho seguimos com pilares semelhantes aos dos trabalhos anteriores. Entretanto, argumentamos que escolher as estratégias de extração de eventos a partir de características das bases de dados disponíveis, bem como de requisitos das aplicações que desejam consumir os eventos extraídos pode levar à identificação não só de um número maior de eventos como aumentar a qualidade da descrição dos mesmos. Dessa forma, o objetivo geral deste trabalho é propor um modelo que auxilia o projetista de aplicações ITS a utilizar as características mencionadas acima em conjunto para, com base em um repositório de opções de técnicas e algoritmos, escolher quais delas possuem o potencial de tornar a detecção de eventos de trânsito mais produtiva em quantidade e qualidade.

O restante deste trabalho está organizado da seguinte forma: A seção 2 apresenta e discute alguns dos trabalhos relacionados encontrados na literatura. A seção 3 descreve o modelo proposto para detecção de eventos de trânsito. Um estudo de caso, realizado para avaliação do modelo, está descrito na seção 4, e os resultados encontrados com sua implementação estão apresentados e discutidos na seção 5. Por fim, a seção 6 contém uma breve conclusão e algumas propostas de trabalhos futuros.

2. Trabalhos Relacionados

Grande parte dos trabalhos sobre detecção de eventos de trânsito disponíveis na literatura exploram redes sociais como fontes de dados. Esses trabalhos propõem o uso de uma única técnica de extração, aplicando-a sobre os dados coletados e avaliando a eficácia obtida com base em métricas tais como precisão, revocação e medida F.

Em maior número, encontra-se na literatura trabalhos que utilizam o *Twitter* como única fonte de dados, diferindo-se apenas quanto à técnica de extração e tipos de eventos detectados. [Anantharam et al. 2015] discutem uma abordagem que aplica processamento de linguagem natural, dentre outras técnicas, para detectar eventos que podem ser utilizados como notificações de trânsito. Os tuítes são coletados com o objetivo de determinar quando existem acidentes, obstruções ou situações que impactam grandes áreas. [Kokkinogenis et al. 2015] apresentam uma metodologia utilizando o *Twitter* como uma abstração de sensor artificial. O estudo de caso segue quatro passos princi-

pais: obter tuítes, determinar se eles estão relacionados a trânsito, extrair informações de localização e analisar o sentimento do texto. [Ribeiro Jr et al. 2012] propõem uma aplicação para detecção de eventos e condições do trânsito na cidade de Belo Horizonte. Inicialmente foi construído um dicionário previamente populado, chamado de *gazetteer* [Ratinov and Roth 2009], com o qual é possível identificar ruas, avenidas e vizinhanças da cidade. Para avaliação da aplicação foram utilizados 505 tuítes rotulados. [Sakaki et al. 2012] utilizam um método para filtrar tuítes de acordo com uma localização. O modelo foi construído com base em um *Support Vector Machine* (SVM) que possui a capacidade de classificar o relacionamento entre a ocorrência reportada e a atual localização daquela informação. *Tweet-LDA* (*Latent Dirichlet Allocation*), proposto por [Wang et al. 2014], é uma variação do método LDA para detecção de eventos de trânsito. Ele apresenta uma abordagem semi-supervisionada que trabalha com amostras de palavras contidas em um subconjunto de uma bag of words [Zhang et al. 2010]. Para validação comparou-se o método ao *SVM* utilizando acurácia como métrica.

Os trabalhos de [Petalas et al. 2016] e [Tejaswin et al. 2015] fazem uso do *Twitter* em conjunto com outra fonte de dados. Os primeiros propõem um sistema de transporte inteligente que utiliza como fontes de dados o *Twitter* e dois outros serviços: o *Highways England* e o *Birmingham City Council*, os quais fornecem uma série de informações de trânsito (e.g. fluxo de tráfego e velocidade média dos veículos). Já os autores do segundo trabalho, apresentam uma metodologia para detecção de eventos de trânsito que utiliza *Twitter* e dados meteorológicos. Técnicas de processamento de linguagem natural foram aplicadas na obtenção da localização de eventos no texto de tuítes.

Este trabalho se diferencia dos demais da literatura por apresentar um modelo que visa combinar as características de possíveis múltiplas bases de dados com os requisitos de uma aplicação ITS quanto ao processo de detecção, de maneira a determinar, dentre as opções disponíveis, a técnica de extração mais adequada. Com isso, espera-se que cada aplicação tenha o potencial de ampliar a quantidade de eventos e a qualidade dos resultados. Uma instância do modelo foi implementada em um estudo de caso com alto volume de dados reais. Até onde os autores puderam pesquisar, os trabalhos relacionados disponíveis atualmente na literatura não apresentam uma estratégia genérica de indicação automática de algoritmos para extração de eventos de trânsito, mas sim análises e resultados pensados para recortes específicos. O modelo proposto é, portanto, inovador neste sentido, permitindo que os projetistas de diversas e diferentes aplicações encontrem, de maneira simplificada e objetiva, indicações capazes de levar à resultados satisfatórios em quantidade e qualidade de eventos identificados.

3. Modelo para Detecção de Eventos de Trânsito

O modelo proposto neste trabalho foi projetado para auxiliar na detecção de eventos de trânsito utilizando múltiplas fontes de dados, que não são pré-definidas e sim inerentes ao contexto de cada aplicação. Nesse caso, o projetista de uma aplicação ITS providenciará uma ou mais bases de dados (e.g., redes sociais, GPS de veículos, dados meteorológicos), e informará ao modelo que deseja utilizá-las. Isso deve ser feito por meio de arquivos de metadados, conforme será explicado a seguir, mantendo o modelo genérico o suficiente para trabalhar com diversos e diferentes tipos de dados. Uma das vantagens de ter acesso à múltiplas bases de dados é a possibilidade de aumentar o volume de eventos detectados, bem como de melhorar o grau de caracterização de cada um deles.

Além de múltiplas fontes de dados, o modelo também considera uma variedade de técnicas e ferramentas para detecção de eventos. Técnicas diferentes possuem requisitos e características diversas, resultando em extrações com diferentes níveis de qualidade, dependendo muitas vezes do formato e quantidade de dados, bem como apresentando demandas computacionais distintas, que levarão a tempos de resposta maiores ou menores. Dessa forma, fica claro que diferentes combinações de fontes de dados e requisitos de aplicações quanto ao processo de detecção se beneficiam mais ou menos de acordo com a estratégia de extração utilizada. Diferentemente das bases de dados, que são ofertadas pelo projetista da aplicação, os algoritmos de extração a serem sugeridos pelo modelo foram pré-definidos. É possível realizar modificações, efetuando-se mudanças na implementação proposta e descrita neste trabalho.

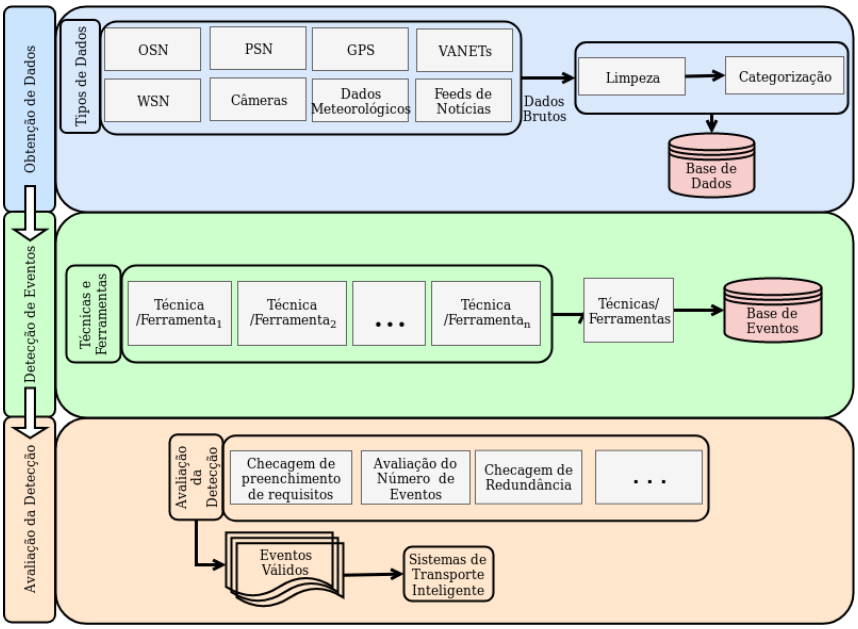

Figura 1. Visão geral do modelo proposto para detecção de eventos de trânsito

A figura 1 apresenta o modelo para extração de eventos proposto, composto de 3 fases: (1) Obtenção de Dados, (2) Detecção de Eventos e (3) Avaliação. Na fase (1) o projetista da aplicação seleciona as múltiplas fontes de dados a serem utilizadas. Ele executa atividades que processam esses dados no sentido de limpá-los e prepará-los para serem usados por técnicas e ferramentas de análise de dados, armazenando-os, ao final, em bases de dados que servirão de entrada para a próxima fase. O projetista ainda informa os metadados solicitados tanto para as bases de dados como para requisitos da aplicação que utilizará os eventos extraídos. Os metadados são especificados em arquivos no formato JSON pré-determinados pelo modelo. Na fase (2), os arquivos de metadados são passados para uma Árvore de Decisão (AD), que representa um método simples e direto, de baixo custo computacional, para a escolha dos possíveis algoritmos que farão a detecção dos eventos. Com a AD é possível determinar, dentre as técnicas e ferramentas já disponíveis no modelo, uma ou mais opções com melhor potencial de detecção para cada combinação particular de bases de dados e aplicação. Neste ponto, as estratégias retornadas pela AD já podem ser usadas junto aos dados, os quais já devem se encontrar limpos e categorizados. A fase (3) valida o que foi detectado, checando critérios de qualidade especificados para

o contexto corrente de utilização do modelo.

No modelo, a AD proposta combina os campos dos metadados das bases de dados e dos requisitos da aplicação. A tabela 1 mostra quais são os metadados propostos para caracterização das fontes de dados e alguns dos valores que podem ser usados para preenchimento dos mesmos. Ela também mostra os metadados que indicam as expectativas de uma aplicação quanto ao processo de extração de eventos de trânsito. A ideia é permitir que cada aplicação expresse informações tais como a importância dada a cada uma das três partes da tupla que caracteriza um evento, a necessidade ou não de rapidez na detecção, a priorização por dados coletados em tempo real, dentre outros aspectos. Ao final, o objetivo da AD é fornecer, de maneira automatizada, uma ou mais técnicas que sejam mais adequadas, isto é, com melhor potencial de detecção para o contexto da aplicação. A AD foi treinada com um conjunto de possíveis combinações entre os tipos de metadados considerados (1), tendo sido alimentada com as seguintes técnicas e ferramentas para análise de dados: *K-Nearest Neighbors(k-NN)*, AD, *random forests* e *SVM*. A obtenção de uma precisão de 72.56% para a AD indica que a mesma possui uma taxa reduzida de falsos positivos. Neste trabalho considera-se a extração dos seguintes tipos de eventos de trânsito: acidente, construção, reforma na via e engarrafamento usual.

Tabela 1. Especificação dos metadados referentes às fontes de dados

Metadado/Base de dados	Exemplos	**Metadado/Aplicação**	Exemplos
Tipo de Dados	Texto, Numérico	**O que**	Sim, Não
Categoria Técnica	Tempo Real, Download	**Quando**	Sim, Não
Volume	Alto, Médio, Baixo	**Onde**	Sim, Não
Qualidade	Alta, Média, Baixa	**Aplicação**	Alertas, Emergência
Tipo Arquivo	Com/Sem Estrutura	**Detecção Rápida**	Sim, Não
Organização	Plano, Hierárquico	**Tempo Real**	Sim, Não
Velocidade Detecção	Alta, Baixa	**Qualidade**	Alta, Baixa
Fonte	Twitter, Waze	**Volume Dados**	Alto, Baixo

4. Estudo de Caso: Detectando Eventos de Trânsito em uma Área de Alta Concentração Urbana

Para validação do modelo apresentado e discutido na seção 3, apresentamos um estudo de caso desenvolvido considerando uma aplicação ITS que consome eventos de trânsito em uma região de alta densidade urbana. Foram coletados e utilizados dados de trânsito da cidade de São Francisco, na Califórnia/EUA, especialmente na região da baía, uma vez que este local é conhecido pelo tráfego normalmente sobrecarregado. A aplicação considerada neste estudo de caso tem como principal objetivo fornecer informações relevantes em forma de mensagens para um serviço de carros conectados de uma cidade inteligente. Cada mensagem pode ser importante no papel do rebalanceamento do tráfego em caso de áreas impactadas por acidentes e interdição de ruas e avenidas em decorrência de alguma reforma ou melhoria.

4.1. Materiais

O estudo de caso utilizou dados oriundos das redes sociais *Waze*, *Twitter* e *HEREMaps*[1], sendo esta última utilizada apenas para validação dos eventos extraídos (*ground truth*). Os

[1]https://www.waze.com, https://www.twitter.com, https://developer.here.com/develop/rest-api

dados coletados continham, em sua maioria, relatos de problemas de trânsito na região da baía de São Francisco, estado da Califórnia/EUA. Para as implementações, foram utilizados os seguintes recursos de software: *numpy*, *scikit-learn*, *jupyter notebook* e *python 3*. Além disso, os equipamentos de hardware foram: *MacBook Pro*, com *OSX - Mojave*, processador 2,8 GHz Intel Core i7, 16GB de memória RAM.

4.2. Métodos

Todas as fases descritas na seção 3 foram instanciadas. Na fase (1), para a aplicação do estudo de caso, considerou-se duas fontes de dados: *Twitter* e *Waze*. Os dados foram obtidos por meio de *web crawlers* e *REST APIs*. Uma vez que o Waze não disponibiliza APIs para coleta de dados diretamente em seu sistema, foram selecionadas contas do aplicativo no *Twitter*. Em um período de 3 meses de postagem foi possível coletar 7.217 tuítes dessas contas, bem como outros 300 mil (reduzidos para 67.489 após o processo de rotulação) de contas diversas do *Twitter* que faziam menção à eventos de trânsito na região da baía de São Francisco, na Califórina/EUA. Já os dados coletados do serviço de geolocalização *HEREMaps*, obtidos via uso de APIs diretas do sistema, foram utilizados para a validação dos eventos detectados após o uso do modelo (*ground truth*). Assim, neste trabalho, um evento extraído é considerado válido foi notificado pelo *HEREMaps*.

Para limpeza dos dados foi utilizado um *script* Python. Para o texto de cada um dos tuítes, buscamos e eliminamos conteúdos iniciados com símbolos marcadores, tais como #, além de links e outras informações ruidosas. As abreviações também foram substituídas pelas palavras completas, em um processo de tentativa e erro com uso de expressões regulares, visto que elas também podem causar viés na criação de um modelo de aprendizado de máquina para extração de informações relevantes no contexto deste trabalho. Após a limpeza, cada uma das fonte de dados foi categorizada e armazenada novamente em banco de dados não relacional. Todos os dados utilizados foram manualmente rotulados. Neste processo, observou-se para cada texto, se o mesmo possuia informações sobre os atributos o que e onde de um evento.

Na fase (2), utilizando os metadados construídos para as bases de dados usadas e para a aplicação considerada (ver Tabela 1), bem como a lista de técnicas e ferramentas, conforme explicação em 3, a AD determinou quais técnicas seriam capazes de melhor extrair eventos. Em seguida, os mesmos foram armazenados em uma base de eventos como tuplas ($T = \{\{o\_que\}, \{onde\}, \{quando\}\}$). Neste estudo de caso, a AD levou a uma folha com os algoritmos *SVM* e *random forests*, os quais são conhecidos na literatura por apresentarem bom comportamento em detecção de maneira generalista, em variados cenários e contextos. Com os dados limpos, categorizados e de posse das técnicas e ferramentas estabelecidas pela AD, é possível começar a detecção dos eventos. Embora a AD tenha indicado o uso de 2 algoritmos, para fins de comparação foram implementadas as 4 técnicas disponíveis no modelo para obtenção do atributo o que do evento. Para a etapa de definição do local onde está ocorrendo o evento (onde) foi adotado um modelo importante da literatura, o extrator de entidades de localização de *Stanford* [Finkel and Manning 2009]. Em conjunto com este extrator, foi adicionado um *gazetteer* com referências de latitude e longitude de pontos específicos da cidade de São Francisco. O objetivo é reduzir o tempo e o consumo de recursos com processamento de geocódigo. Para a definição do momento quando ocorre o evento (quando) foi extraída a informação diretamente da fonte de dado, ou seja, quando a mesma foi publicada na rede social.

Uma vez que os dados já se encontravam previamente rotulados, foi possível começar a tarefa de treinamento dos modelos de aprendizado de máquina. Na implementação foi utilizada a biblioteca *scikit-learn*. Cada algoritmo utilizou um conjunto de parâmetros calculado por um recurso da própria biblioteca para estimar quais são as configurações mais adequadas. Para ilustrar a fase (3) do modelo, avaliamos se todos requisitos da aplicação foram satisfeitos, isto é, em quantos eventos foram retornadas as informações essenciais e solicitadas para a caracterização de um evento: o que, quando e onde. A implementação utilizada neste caso utiliza uma estratégia simples, percorrendo linearmente toda a base de eventos e verificando em quais deles existem todas as informações essenciais.

A seção a seguir apresenta resultados sobre a capacidade de detecção de eventos do modelo proposto neste trabalho para o estudo de caso apresentado. Além de mostrar a quantidade e qualidade dos eventos detectados considerando o cenário de uso das duas fontes de dados e de todos os algoritmos retornados pela AD, exatamente conforme descrito acima, serão também apresentados resultados de outros cenários com o objetivo de ressaltar a importância dos dois principais diferenciais do modelo proposto neste trabalho: o uso de múltiplas fontes de dados e a seleção de algoritmos de extração mais alinhados com as características correntes das bases de dados e da aplicação.

5. Resultados

5.1. Cenários

A seguir estão descritos os 4 cenários de experimentação criados neste trabalho. Os 3 primeiros cenários consideram o uso dos algoritmos de extração de eventos apontados pela AD do modelo proposto neste trabalho, porém com mudanças quanto às fontes de dados utilizadas. Já o quarto cenário utiliza um trabalho de referência da literatura para a extração de eventos. Para este último cenário foi utilizada a mesma base de dados do *Twitter* considerada nos cenários 1 a 3. Vale ressaltar que o HEREMaps não foi utilizado como fonte de dados em nenhum cenário, uma vez que serviu como ground truth para validação dos eventos extraídos.

- Cenário 1: modelo deste trabalho e apenas *Twitter* como fonte de dados;
- Cenário 2: modelo deste trabalho e apenas *Waze* como fonte de dados;
- Cenário 3: modelo deste trabalho e ambos, *Twitter* e *Waze*, como fontes de dados;
- Cenário 4: modelo *baseline* da literatura e *Twitter* como fonte de dados;

5.2. Resultados - Cenários 1 e 2

Nas figuras 2 e 3 estão apresentadas as taxas de acerto na detecção de eventos de trânsito extraídos com $T = \{\{o\_que\}, \{onde\}\}$ para os cenários 1 (apenas *Twitter*) e 2 (apenas *Waze*). Os resultados para o Cenário 1 mostram que utilizar somente o *Twitter* dificulta a detecção uma vez que não há garantia de referências à localização exata, ou qualquer localização, no conteúdo de um tuíte do *Twitter*. Além disso, nessa base os dados são esparsos e o vocabulário amplo. Dessa forma, de acordo com o algoritmo utilizado, a taxa de acerto varia entre 57% e 62%. Já a utilização do *Waze* como única fonte de dados apresenta por si só resultados satisfatórios, uma vez que existe maior presença de informações completas e relevantes sobre trânsito nesta base de dados. Assim, para o Cenário 2, dependendo do algoritmo utilizado, foi possível observar que entre 81% e

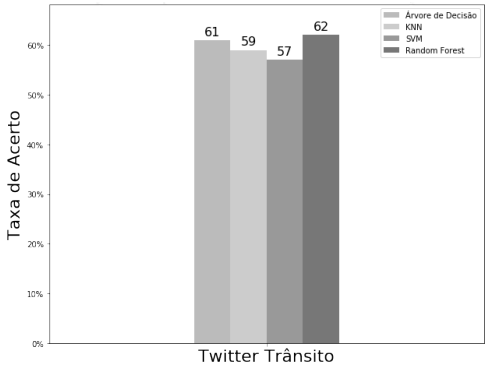

Figura 2. Cenário 1

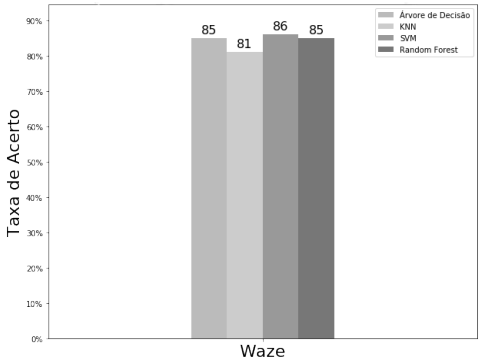

Figura 3. Cenário 2

Figura 4. Eventos válidos (o que e onde)

86% dos eventos reportados constavam no *HEREMaps* com os atributos tipo e localização corretamente preenchidos.

5.3. Resultados - Cenário 3

Com o Cenário 3 é possível observar que o uso de múltiplas fonte de dados neste estudo de caso se mostrou vantajosa uma vez que foram foram detectados 21.22% e 30.14% a mais de eventos do que utilizando, respectivamente, apenas *Waze* (Cenário 2) e *Twitter* (Cenário 1). Considerando $T = \{\{o\_que\}\}$, foi possível encontrar 92,782% dos eventos detectados em conjunto pelo *Twitter* e *Waze* nos dados do *HEREMaps*. Já quando $T = \{\{o\_que\}, \{onde\}\,\{quando\}\}$, 86,8% dos eventos extraídos foram validados. Na tabela 2 é possível encontrar os principais resultados obtidos para o Cenário 3 no que diz respeito à sua capacidade de reportar os eventos da lista do *HEREMaps* (porcentagem dos eventos do *HEREMaps* também identificados pelo Cenário 3).

Eventos	**Resultado**
Quantidade Total	6135
% Eventos (o que)	87%
% Eventos (o que, onde)	70%

Tabela 2. Cenário 3

5.4. Resultados - Cenário 4

Em [Anantharam et al. 2015], os autores propõem uma abordagem de detecção de eventos de trânsito que usa o *Twitter* como base de dados e uma única estratégia de extração. Este trabalho foi escolhido para ser o *Baseline* de comparação com a literatura e sua implementação representa o Cenário 4 de avaliação. Como critério de avaliação, serão comparados os volumes de eventos detectados pelos Cenários 3 e 4. Nas implementações foram utilizados os mesmos dados do *Twitter* considerados para os Cenários 1 e 3. Vale ressaltar que a detecção da localização no Cenário 4 é feita diretamente pela latitude e pela longitude, exatamente como proposto pelo trabalho relacionado.

Os autores em[Anantharam et al. 2015] utilizam uma tupla contendo 5 elementos para caracterizar um evento. Os atributos desta tupla que correspondem àqueles adotados neste trabalho são: Eventos (conjunto de termos que representem eventos citados no

texto), Dia da Semana (data associada ao dado) e Localização(identificador textual que corresponde à localização associada ao texto, de acordo com uma estrutura de grades aninhadas (*geohash*)). Essas tuplas são processadas em conjuntos agrupados por intervalos de tempo regulares. Em seguida estes conjuntos são separados em subconjuntos de acordo com os termos que caracterizam o evento ocorrido, agrupando dados categorizados com o mesmo tipo de evento. Os elementos de cada um desses subconjuntos são filtrados, mantendo-se apenas a localização que ocorre com mais frequência. Para que seja considerado um evento válido, o *Baseline* propõe o uso de redundância (número de vezes que o evento ocorre na base de eventos). Para o Cenário 4 implementado neste trabalho, o evento precisa aparecer pelo menos três vezes, para que seja legítimo. Este valor de redundância foi proposto e avaliado pelo próprio *Baseline*.

Na detecção de $T = \{\{o\_que\}\}$, o Cenário 4 alcançou 68% de taxa de acerto. O Cenário 3, que usa o modelo proposto neste artigo e para o qual foram utilizadas duas fontes de dados, *twitter* e waze, chegou a 87% de taxa de acerto. No geral, para $T = \{\{o\_que\}, \{onde\}\, \{quando\}\}$, foi possível obter para o Cenário 4 uma taxa de acerto máxima de 12.9%, o que está abaixo do valor de 70% alcançado pelo Cenário 3.

6. Conclusão e Trabalhos Futuros

Este trabalho propôs um modelo para extrair e caracterizar um evento, fornecendo as informações sobre tipo, local e momento de ocorrência, para detectar ocorrências que causam impactos no trânsito. Este modelo explora metadados das fontes de dados utilizadas junto com alguns requisitos da aplicação quanto ao processo de detecção para determinar, usando uma AD, quais técnicas seriam mais adequadas para extração de eventos. Uma instância desse modelo foi implementada em um estudo de caso. Os resultados indicaram que utilizar a AD torna simples a tarefa de determinar algoritmos mais adequado em situações específicas e que mais de uma fonte de dados pode melhorar a detecção devido à interferência complementar que tais fontes podem exercem umas sobre as outras. A mesma AD pode ser utilizada em outras situações (e.g., outras cidades, aplicações ITS, fontes de dados) desde que se produza os arquivos de metadados necessários. Alguns dos trabalhos futuros ligados ao modelo proposto neste artigo são: identificação de novas parametrizações para metadados e requisitos das aplicações, avaliação de outros mecanismos para a escolha das melhores técnicas de extração de eventos, adoção de novas estratégias de avaliação dos eventos detectados e uso de algoritmos mais diversificados com potencial de detectar novos eventos utilizando abordagens de aprendizagem por reforço.

Agradecimentos

Este trabalho contou com o apoio da Fapemig, CNPq, CAPES e FAPESP (projetos 15/24494-8 e & 18/23064-8).

Referências

Albuquerque, F. C., Casanova, M. A., Lopes, H., Redlich, L. R., de Macedo, J. A. F., Lemos, M., de Carvalho, M. T. M., and Renso, C. (2016). A methodology for traffic-related twitter messages interpretation. *Computers in Industry*, 78:57–69.

Allan, J., Papka, R., and Lavrenko, V. (1998). On-line new event detection and tracking. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 37–45. ACM.

Anantharam, P., Barnaghi, P., Thirunarayan, K., and Sheth, A. (2015). Extracting city traffic events from social streams. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 6(4):43.

Brants, T., Chen, F., and Farahat, A. (2003). A system for new event detection. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 330–337. ACM.

Finkel, J. R. and Manning, C. D. (2009). Nested named entity recognition. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 141–150. Association for Computational Linguistics.

Galton, A. and Augusto, J. C. (2002). Two approaches to event definition. In *International Conference on Database and Expert Systems Applications*, pages 547–556. Springer.

Kokkinogenis, Z., Filguieras, J., Carvalho, S., Sarmento, L., and Rossetti, R. (2015). Mobility network evaluation in the user perspective: Real-time sensing of traffic information in twitter messages. *Advances in Artificial Transportation Systems and Simulation*, pages 219–234.

Kumaran, G., Allan, J., and McCallum, A. (2004). Classification models for new event detection. In *International conference on information and knowledge management (CIKM2004)*.

Petalas, Y. G., Ammari, A., Georgakis, P., and Nwagboso, C. (2016). A big data architecture for traffic forecasting using multi-source information. In *International Workshop of Algorithmic Aspects of Cloud Computing*, pages 65–83. Springer.

Ratinov, L. and Roth, D. (2009). Design challenges and misconceptions in named entity recognition. In *Proceedings of the thirteenth conference on computational natural language learning*, pages 147–155. Association for Computational Linguistics.

Ribeiro Jr, S. S., Davis Jr, C. A., Oliveira, D. R. R., Meira Jr, W., Gonçalves, T. S., and Pappa, G. L. (2012). Traffic observatory: a system to detect and locate traffic events and conditions using twitter. In *Proceedings of the 5th ACM SIGSPATIAL International Workshop on Location-Based Social Networks*, pages 5–11. ACM.

Sakaki, T., Matsuo, Y., Yanagihara, T., Chandrasiri, N. P., and Nawa, K. (2012). Real-time event extraction for driving information from social sensors. In *Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), 2012 IEEE International Conference on*, pages 221–226. IEEE.

Tejaswin, P., Kumar, R., and Gupta, S. (2015). Tweeting traffic: Analyzing twitter for generating real-time city traffic insights and predictions. In *Proceedings of the 2nd IKDD Conference on Data Sciences*, page 9. ACM.

Wang, D., Al-Rubaie, A., Davies, J., and Clarke, S. S. (2014). Real time road traffic monitoring alert based on incremental learning from tweets. In *Evolving and Autonomous Learning Systems (EALS), 2014 IEEE Symposium on*, pages 50–57. IEEE.

Zhang, Y., Jin, R., and Zhou, Z.-H. (2010). Understanding bag-of-words model: a statistical framework. *International Journal of Machine Learning and Cybernetics*, 1(1-4):43–52.