Um *Framework* de Extração e Etiquetamento de Informações de Trânsito

Leonardo Tetéo; Pedro Moura; Elton F. de S. Soares; Carlos Alberto V. Campos

[1]Departamento de Informática Aplicada (DIA)
Universidade Federal do Estado do Rio de Janeiro (UNIRIO)
Avenida Pasteur, 458 – 22.290-250 – Rio de Janeiro – RJ – Brasil

Resumo. *Com o grande avanço nas tecnologias computacionais, está cada vez mais possível usar as redes sociais para a coleta e análise de informações de indivíduos, comunidades e a respeito da dinâmica das cidades em tempo real. O excesso de informação, porém, é um desafio até mesmo no contexto das cidades, onde muitos eventos ocorrem em paralelo. Neste contexto, este trabalho propõe um framework que tem como função facilitar a extração, tratamento e identificação de eventos e suas localidades em tweets escritos na Língua Portuguesa utilizando a técnica Conditional Random Fields para realizar a tarefa de Reconhecimento de Entidades Nomeadas. Os resultados obtidos demonstram o potencial da ferramenta em identificar eventos de trânsito em uma dada localidade de interesse.*

Abstract. *With the great development of computational technologies, it has been possible to use social networks to collect and analyze information of individuals, communities and with respect to cities in real time. The information overload, however, is a challenge even in the context of cities, where many events occur in parallel. In this context, this paper describes the development of a framework that seeks to ease the extraction, treatment and identification of events and their locations in tweets written in the Portuguese language using the Conditional Random Fields technique to address the Named Entity Recognition task. The obtained results demonstrate the potential of the tool in identifying important traffic events in a given location of interest.*

1. Introdução

Uma das grandes revoluções deste século foi a introdução da tecnologia da informação no cotidiano da sociedade. Desde o final da década de 90 até os dias atuais, a vida pessoal e profissional mudaram muito. A tecnologia tomou conta de cada aspecto da vida humana, tornando-se algo essencial e evoluindo rapidamente. A criação da Internet em 1969 marcou o começo da nova era conhecida como Era da Informação. Após um período de alto desenvolvimento tecnológico, em que se destacaram a invenção da Internet banda larga, os dispositivos móveis e as redes sociais, hoje observamos uma forte dependência da sociedade tanto da Internet quanto das constantes informações propocionadas por esses adventos.

A evolução desses aspectos permitiu que qualquer pessoa tenha toda a informação que necessite em tempo praticamente real. Isso proporciona muitos benefícios, mas também gera desafios causados pelo grande volume de informação recebido. Essa sobrecarga de informação pode fazer com que os usuários não cheguem no resultado que

esperam. Dessa forma, ainda há muitos avanços a serem realizados e, neste sentido, a área Extração de Conhecimento é uma das que se propõem a abordar tais desafios.

Tais avanços tecnológicos não só beneficiaram as pessoas, como também cidades inteiras. O conceito de cidades inteligentes está sendo cada vez mais adotado por gestores publicos e empresas, de modo que diversos países estão se voltando para a tecnologia da informação e comunicação (TIC) como um meio de obter dados nunca antes pensados e observados sobre a dinâmica das cidades, de maneira a realizar melhorias na vida de seus cidadãos. Muitas pesquisas têm sido desenvolvidas neste contexto, que é amplo e multidisciplinar, envolvendo diferentes áreas do conhecimento. A TIC também está presente sendo *Big Data* e Extração de Conhecimento dois campos presentes nas pesquisas sobre cidades inteligentes (*Smart Cities*). Assim sendo, muitas pesquisas pertencentes ao estado-da-arte que unem os conceitos desses campos fazem uso das redes sociais como fonte de obtenção de dados, em que o Twitter assume um papel de destaque, dada a sua característica de atualização em tempo real.

É nesse contexto que se encaixa este trabalho, no qual apresentamos um *framework* de ferramentas que utilizam técnicas da área de Extração do Conhecimento para o problema de Reconhecimento de Entidades Nomeadas (*Named Entity Recognition - NER*) no Twitter, aplicado ao contexto de mobilidade urbana como sendo parte das Cidades Inteligentes. Mais especificamente, o foco está na identificação de eventos de trânsito em centros urbanos brasileiros a partir da análise de mensagens no Twitter. A precisão e eficácia dos métodos utilizados foram avaliadas para determinar o grau de utilidade do *framework* proposto. Para o desenvolvimento deste *framework*, foi necessária a adaptação de algoritmos de extração, tratamento e identificação para a língua portuguesa. Muitos destes foram preparados para utilização na língua inglesa, que apesar de ter características em comum com o português, também possui muitas diferenças que impactam nos passos mencionados. Além disso, a linguagem coloquial que predomina no Twitter também impõe outros desafios que eventuais modelos criados para o português não resolvem.

As contribuições deste trabalho se situam no fato de ter sido adaptado um modelo de aprendizagem de máquina para a língua portuguesa, através da adaptação da ferramenta StanfordNER, e sua consequente aplicação na extração de localidade e eventos do Twitter, além da criação de uma ferramenta para visualização em mapa de tais eventos extraídos, a partir das respectivas localidades. Ademais, destaca-se também o ganho de acurácia no modelo obtido neste trabalho em relação ao trabalho de [Anantharam et al. 2015].

2. Fundamentação teórica

2.1. Cidades Inteligentes

Atualmente, 54% das pessoas no mundo moram em cidades, sendo que, no Brasil são mais de 85% e em países de primeiro mundo o percentual está entre 75% até aproximadamente 90% [CIA 2017]. Este grande número tem se tornado um desafio para governos e aos próprios cidadãos, que estão sempre em busca de uma qualidade de vida adequada. No entanto, como obtê-la em grandes cidades onde problemas surgem com o acúmulo de pessoas? Poluição, engarrafamentos constantes, acidentes, crimes e outros problemas só aumentam com o crescimento populacional das cidades. Hoje, com o advento da tecnologia e a era da informação, todos têm se voltado para a tecnologia para buscar

soluções para problemas dos mais variados tipos e as cidades são grandes protagonistas com o conceito de cidades inteligentes.

Cidade inteligente é um conceito muito abrangente que envolve vários aspectos de uma cidade, desde o transporte e trânsito, à saúde. Desta forma, existem diferentes definições para cidades inteligentes que focam em diferentes aspectos. Sustentabilidade e eficiência são duas características muito valorizadas entre as definições de cidades inteligentes. Em [P Hancke and Silva 2012], cidades inteligentes foram definidas como cidades que utilizam dispositivos inteligentes para monitorar e controlar a infraestrutura e serviços, garantindo eficiência e sustentabilidade. Esta definição é mais voltada ao monitoramento e controle.

Em [P Hancke and Silva 2012] também foram descritos exemplos em que sensores poderiam ser utilizados em várias áreas de cidades inteligentes, como em *Smart Transportation* onde sensores são usados para determinar a intensidade e velocidade do trânsito contribuindo para o planejamento de novas vias e aperfeiçoamento de semáforos, que também podem ser inteligentes. Em *Smart Eletricity and Water distribution* (distribuição inteligente de eletricidade e água) é possível utilizar sensores para identificar vazamentos de água ou cortes na eletricidade. Também é possível melhorar o consumo desses recursos utilizando sensores para coletar dados, utilizando medidores inteligentes.

Em [Neirotti et al. 2014], cidades inteligentes também foram divididas em diferentes domínios que englobam diferentes aspectos de uma cidade. Os autores agruparam estes domínios em *hard domains* e *soft domains*, no que concerne ao grau de utilização da tecnologia em cada domínio. *Hard domains* como transportes, energia e segurança pública utilizam a tecnologia por meio de sensores que monitoram e controlam estes domínios, similar ao que foi afirmado em [P Hancke and Silva 2012]. Em *soft domains* como educação, inclusão social e administração pública, a tecnologia tem uma influência mais limitada possuindo um papel mais auxiliador.

Em [Zanella et al. 2012] os autores falaram sobre a indefinição do significado do conceito de cidades inteligentes, mas também o definem de uma forma mais voltada à parte econômica e governamental ao fazer alusão à sustentabilidade e à eficiência, com o objetivo de reduzir custos para a administração e aumentando a qualidade. Esse trabalho também falou sobre *Internet of Things* (IoT), assunto também abordado em [P Hancke and Silva 2012] e [Neirotti et al. 2014], dando vários exemplos de como IoT pode auxiliar no desenvolvimento de soluções para diferentes problemas, entre eles: gerenciamento de lixo, congestionamentos, consumo de energia, monitoramento de poluição sonora, etc. Neste trabalho, em particular, os autores definiram IoT como um novo paradigma que colocará dispositivos do dia-a-dia na internet através de componentes de rede e protocolos próprios. Muitos destes problemas fazem parte tanto das áreas apresentadas por [P Hancke and Silva 2012], quanto dos domínios apresentados por [Neirotti et al. 2014]. Assim, embora haja certa diferença entre as definições, é possível perceber que há um acordo entre pesquisadores que cidades inteligentes e IoT têm papel importante em áreas como trânsito, segurança, consumo de energia e de água, etc.

2.2. *Named Entity Recognition*

O reconhecimento de entidades nomeadas é um problema da área de Extração do Conhecimento que tem como objetivo reconhecer entidades que possuem nome próprio em

textos escritos em linguagem natural e indicar estas entidades, geralmente em diferentes categorias, por meio de etiquetas. Algumas categorias correspondem a: Organização, Localidade e Pessoa.

Por exemplo, em uma frase como "O Rio de Janeiro continua lindo" a frase seria etiquetada da seguinte forma: "O \O Rio \B-LOCATION de \I-LOCATION Janeiro \I-LOCATION continua \O lindo \O".

A notação utilizada no exemplo acima chama-se BIO e é muito adotada no campo de linguística computacional [Anantharam et al. 2015]. Os trabalhos de [Puiu et al. 2016] e [FarajiDavar et al. 2017] também adotaram essa notação na abordagem ao problema. Os termos são etiquetados respeitando as seguintes regras:

- B-tipo de entidade, por exemplo B-LOCATION, quando esta é a primeira palavra de um nome de uma entidade;
- I-tipo de entidade, por exemplo I-LOCATION, para todas as palavras após a primeira que pertencem ao nome de uma entidade; e
- "O", que significa *Other* (outro), é atribuída a toda palavra que não faz parte do nome de uma entidade.

Com esta notação, é possível delinear a relação entre as palavras mais facilmente, além de ser mais fácil de conseguir extrair o nome completo de uma entidade já que o início e o fim de um nome estão claramente definidos.

2.3. *Conditional Random Fields*

Conditional Random Fields (CRF) é um modelo estatístico geralmente representado por um grafo não-direcionado que mostra a relação entre entidades. No contexto deste trabalho ele representa a relação entre palavras (chamadas de tokens) e as etiquetas que são atribuídas aos tokens.

Esse modelo trabalha com valores que representam a taxa de ocorrência de determinadas sequências no corpo de texto. As arestas no grafo não só representam relações entre *tokens* e *tags*, e entre *tags* e outras *tags*, mas também representam funções que possuem o valor da probabilidade dessa relação existir. Por exemplo, utilizando a frase de exemplo anterior e extraindo somente "Rio de Janeiro" para simplificar o grafo, podemos representar em CRF esta relação conforme expresso na Figura 1.

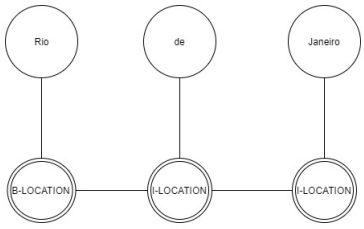


Figura 1. Exemplo de grafo com *tokens* e *tags*.

Esses fatores podem representar a probabilidade de tais relações aparecerem em um *corpus*: qual é a probabilidade de Rio ser etiquetada como B-LOCATION? Qual é a probabilidade de B-LOCATION ser seguida de I-LOCATION? É fácil entender que esta relação é forte quanto se trata da palavra Rio de Janeiro. Ainda assim, é possível deixar

o CRF ainda mais complexo e inteligente adicionando mais fatores. Como, por exemplo, levando em consideração a *tag* atribuída aos *tokens* adjacentes ou levar em consideração o *token* anterior para atribuir uma *tag* a uma nova palavra.

Os resultados alcançados pelos estudos que utilizaram CRFs são bastantes animadores. [Anantharam et al. 2015] utilizou como validação dos eventos extraídos do Twitter uma base de dados de eventos de tráfego do governo da cidade de São Francisco, na Califórnia, onde os experimentos foram realizados: no repositório 511.org. Do total de eventos detectados (1.042), 40% deles (454) coexistiram em tempo e localidade com eventos reportados no 511.org. Com isso, os autores concluiram que ainda há muitos eventos a descobrir, mas o 511.org não oferece a granularidade necessária para extrair todo o potencial da ferramenta. O etiquetamento em si obteve uma precisão média de 71.5% para localizações e 59% para eventos e *Recall* de 76% e 32%, respectivamente.

[Puiu et al. 2016] também utilizou CRF associada a outra técnica de aprendizagem de máquina chamada *Convolutional Neural Network* (CNN) com o objetivo de obter resultados melhores. Esse trabalho foi posteriormente utilizado em [FarajiDavar et al. 2017] na criação de um *framework* para cidades inteligentes.

3. O *Framework*

Este *framework* tem como principal objetivo tornar mais fácil o processo de extração de conhecimento do Twitter. Neste trabalho, mais especificamente, o objetivo é extrair conhecimento sobre eventos de trânsito em tempo real, sendo possível receber tweets através da API do Twitter, tratá-los para retirar possíveis ruídos que possam atrapalhar o etiquetamento posterior, armazená-los e entregar à ferramenta de etiquetamento onde é feito o reconhecimento de palavras que indica a localização e o evento associado, de modo a indicá-los em um mapa. A localização de todos os eventos é indicada somente pelo que está escrito no tweet, as coordenadas atribuídas a um tweet pelo Twitter, quando disponíveis, não são utilizadas em nenhum momento para localizar o evento.

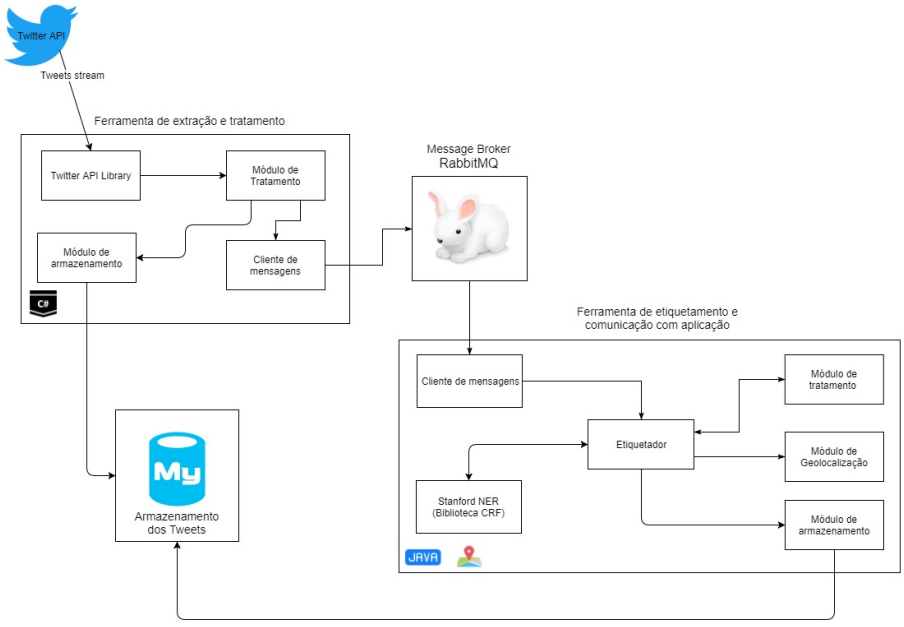


Figura 2. Visão Geral do *Framework*.

O *framework*, como pode ser visto na Figura 2, é composto de quatro componentes principais: (1) Ferramenta de extração e tratamento; (2) Módulo de armazenamento; (3) Serviço de *message broker* RabbitMQ; e (4) Ferramenta de etiquetamento e comunicação com a aplicação.

As setas no diagrama indicam o fluxo de dados dentro do *framework*, ou seja, elas indicam basicamente o caminho que cada tweet percorre até estar pronto para ser utilizado por uma aplicação cliente. Os Algoritmos 1 e 2 descrevem com mais detalhes o funcionamento em alto nível do *framework*, desde o recebimento do tweet, até o armazenamento final no banco de dados. As duas aplicações possuem modos manuais e automáticos, assim, os algoritmos definem o modo automático, onde é possível a utilização do *framework* em tempo real.

Algorithm 1 Ferramenta de Extração e Tratamento

```
1: Aplicação de Extração é iniciada
2: while tempo limite OU número de tweets não for atingido do
3:     Aplicação de Extração recebe TWEET
4:     TWEET é armazenado em Lista TWEETS
5:     TWEET é enviado para o Módulo de Tratamento
6:     TWEET é enviado para o Cliente de Mensagem
7:     if TWEETS atingir tamanho estipulado then
8:         Criar JSON com TWEETS para Módulo de Armazenamento
9:         TWEETS são enviados para o Cliente de Mensagem
10:        Módulo de armazenamento recebe TWEETS
11:        TWEETS são inseridos no banco de dados
12: Finalizar Aplicação de Extração
```

O Algoritmo 1 apresenta o funcionamento da Ferramenta de Extração e Tratamento em alto nível. Na linha 1, a aplicação é iniciada. Na linha 2 um loop é definido sendo a condição de parada um tempo limite ou número de tweets estipulado pelo usuário sendo um parâmetro configurável da aplicação. Dentro do loop a aplicação espera por tweets serem enviados pela API do Twitter. Ao receber um tweet na linha 3, o tweet é armazenado em uma lista de tweets na linha 4. O tweet passa por um processo de tratamento na linha 5 e, após isto, é enviado ao Cliente de Mensagens na linha 6, sendo enviados a Ferramenta de Etiquetamento. Na linha 7 a aplicação testa se a lista de tweets possui um tamanho estipulado pelo usuário, este tamanho também é um parâmetro configurável pelo usuário. Caso este limite tenha sido atingido, a lista é convertida em uma mensagem JSON na linha 8 e enviada para o Cliente de Mensagens na linha 9. Desta vez o Cliente de Mensagens irá enviar os tweets para outra fila destinada ao Módulo de Armazenamento. Este módulo recebe os tweets na linha 10 e este insere os mesmos no banco de dados na linha 11. A linha 12 finaliza a aplicação após o loop.

Algorithm 2 Etiquetamento e Comunicação

```
1: Aplicação de Etiquetamento é iniciada
2: Aguarda receber TWEET
3: while tempo limite não for atingido do
4:     if TWEET recebido then
5:         TWEET é etiquetado pelo Etiquetador
6:     if TWEET ETIQUETADO possuir entidades de localização E evento then
7:         Entidades de localização são extraídas
8:         Consulta é enviada à API de geolocalização
9:         Armazenar no banco de dados
10: Finalizar Aplicação de Etiquetamento
```

O Algoritmo 2 descreve o funcionamento em alto nível da Ferramenta de Etiquetamento e Comunicação. Na linha 1, a aplicação é iniciada. A ferramenta é baseada no conceito de consumidor, assim, na linha 2 a aplicação aguarda o recebimento de tweets iniciando um loop na linha 3 até o tempo limite ser atingido, assim como na Ferramenta de Extração e Tratamento. Caso um tweet seja recebido na linha 4, o mesmo é etiquetado pelo módulo Etiquetador na linha 5. Na linha 6, caso o tweet já etiquetado possua ambas entidades de localização e evento, na linha 7 as entidades de localização são extraídas do tweet. Na linha 8, uma consulta à API de geolocalização é enviada com o objetivo de obter as coordenadas da localização extraída. Ao receber o resultado da consulta na linha 9, o tweet é atualizado no banco de dados com as coordenadas encontradas. Ao final do *loop* a aplicação é finalizada na linha 10.

4. Componentes

Nesta seção, são detalhados os componentes do *framework*: Ferramenta de Extração e Tratamento, o módulo de armazenamento, o RabbitMQ Message Broker Server e a Ferramenta de Etiquetamento e Comunicação.

4.1. Ferramenta de Extração e Tratamento

O primeiro componente do *framework* é a ferramenta de extração, tratamento e armazenamento. Esta é uma aplicação simples que tem como função receber os tweets do Twitter utilizando a API disponibilizada pela própria rede social. Após isto, os tweets são tratados de modo a retirar *hyperlinks* e *emoticons* do texto. Por fim, os tweets são armazenados pela aplicação em um banco de dados. Tal ferramenta foi construída utilizando a linguagem C# .NET Core 2. A escolha pela linguagem C# se deveu à afinidade dos autores com a linguagem e também pela característica do *framework* .NET Core 2.0, um *framework* multiplataforma que possibilita a utilização de uma aplicação em diferentes sistemas operacionais.

A API do Twitter é responsável por prover acesso a outras aplicações a recursos do Twitter como os tweets, postagem de tweets, administração de contas, entre outros recursos. Neste trabalho foram utilizados os recursos de pesquisa e *streaming* de tweets. A pesquisa de tweets é realizada utilizando REST API através de uma linguagem de consulta básica, podendo ser configurados vários parâmetros para refinar a busca. Por exemplo, a consulta a seguir procura por tweets com o texto "@Twitter" na língua inglesa: https://api.twitter.com/1.1/search/tweets.json?q=@Twitter&lang=eu

O recurso de busca foi utilizado neste trabalho em situações em que havia a necessidade de se obter tweets de perfis específicos, especialmente de perfis oficiais da cidade do Rio de Janeiro, que compartilham informações sobre eventos de trânsito, entre outros eventos e alertas. Os tweets destes perfis oficiais foram utilizados para treinamento e testes. Porém, como um dos objetivos do *framework* proposto não é só extrair eventos de trânsito de perfis oficiais, mas também de perfis de cidadãos comuns que comentam fatos vividos por eles, o recurso mais utilizado da API Twitter foi a Streaming API [Twitter 2019], responsável por enviar à aplicação requisitante tweets em tempo real que respeitam os filtros configurados pela aplicação.

O módulo de armazenamento do *framework* utiliza como sistema de gerenciamento de banco de dados (SGBD) o MySQL da Oracle, que é compatível com Windows,

Linux e outros sistemas operacionais baseados em UNIX, o que vai ao encontro da proposta de prover um *framework* compatível com o maior número de plataformas possível. Outros SGBDs podem ser utilizados com ligeiras alterações na aplicação de extração, de modo a ser possível a conexão.

Para realizar a comunicação entre as duas aplicações que compõem o *framework* primeiramente se pensou em utilizar o banco de dados como intermediário, assim a aplicação de extração dos tweets gravaria as informações no banco e a ferramenta de classificação leria estas informações e posteriormente atualizaria os dados com o tweet etiquetado. Porém, dado o custo de comunicação entre aplicação e banco de dados discutida anteriormente, foi necessário buscar uma solução mais eficiente, que auxiliasse o *framework* a manipular os dados em tempo real, objetivo do projeto. Para satisfazer esses requisitos, foi implementada uma arquitetura de *Publish/Subscribe* utilizando um *message broker*.

Publish/subscribe é um paradigma de interação distribuída que permite o desenvolvimento de sistemas fracamente acoplados e altamente escaláveis [Eugster et al. 2003]. RabbitMQ, uma das implementações *open-source* mais populares deste paradigma, permite que aplicações se comuniquem de maneira assíncrona através da publicação e consumo de mensagens em um *message broker* [Dobbelaere and Esmaili 2017]. A decisão de utilizar um *message broker* é baseada na forma como o CityPulse Framework [Puiu et al. 2016] realiza a comunicação entre seus diferentes módulos: o RabbitMQ, também foi utilizado em [Bajaj et al. 2016] com o mesmo objetivo. Outra vantagem de utilizarmos o RabbitMQ, é o fato de ser uma solução de *message broker* multiplataforma. Além disso, a ferramenta pode administrar filas de trabalho onde dois agentes consomem a mesma fila e dividem o trabalho. Também é possível disparar a mesma mensagem para várias aplicações, criar roteamentos, dentre outras funcionalidades. Neste trabalho, porém, a ênfase está na utilização do envio de mensagem simples.

4.2. Ferramenta de Etiquetamento e Comunicação

A ferramenta de etiquetamento e comunicação é responsável por receber os tweets extraídos pela ferramenta de extração e etiquetá-los utilizando *Conditional Random Fields* como técnica de etiquetamento de entidades nomeadas.

4.2.1. Etiquetamento

A classe responsável pelo etiquetamento dos tweets utiliza a biblioteca do *Stanford Natural Language Processing Group* para realizar NER utilizando CRF. A biblioteca Stanford NER possui várias *features* (características) mais utilizadas para a aplicação de CRF a NER já implementadas, em que é necessária apenas uma questão de configuração. Outra opção para esse objetivo seria a biblioteca LingPipe, que foi utilizada por [Anantharam et al. 2015], mas que neste trabalho foi preterida pela Stanford NER em virtude de facilidade de implementação.

4.2.2. Geolocalização

Com os dados de localização e tipo de eventos acessíveis, essa localização em texto retirada de um *tweet* pode ser transformada em coordenada e marcada em um mapa utilizando uma biblioteca de geolocalização. Neste trabalho foram utilizadas duas bibliotecas para este fim: Google Places API Web Service e Nominatim do Open Street Maps API. Ambas foram escolhidas por serem as ferramentas mais conhecidas em relação à geolocalização, a Open Street Maps API em particular foi utilizada em vários artigos relacionados como base de dados de localizações, como por exemplo na construção de um dicionário para reconhecimento de entidades como feito por [Anantharam et al. 2015] e [FarajiDavar et al. 2017]. Já a Google Places API Web Service é uma das APIs relacionadas ao Google Maps que são oferecidas comercialmente a empresas e desenvolvedores sendo altamente renomada no mercado.

4.2.3. Treinamento

Durante o período de experimentos foram extraídos 4.499.313 tweets, utilizando a API do Twitter, no período de 24 de Agosto de 2017 até 4 de Outubro de 2017. Destes tweets, 11.080 foram utilizados durante o processo de treinamento. Devido ao grande número de tweets e ao fato de que a base de dados de tweet mencionado foi acumulada durante quase dois meses, o processo de treinamento também foi realizado gradualmente, porém sem prejuízo sobre os resultados. A Figura 3 mostra como o treinamento foi realizado, dando origem a modelos intermediários e, por fim, a um modelo final.

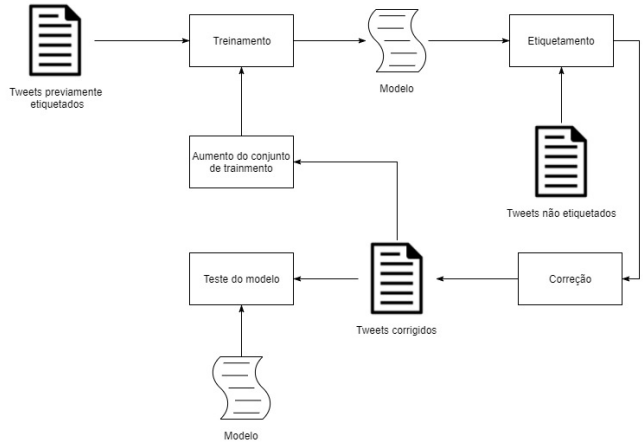


Figura 3. Processo de treinamento.

Em um primeiro momento, para obtermos um conjunto de tweets já etiquetados, usamos a técnica utilizada por [Anantharam et al. 2015] em que é usado um dicionário de localizações e eventos para etiquetar as palavras dos tweets extraídos. O dicionário de localizações foi obtido da base do Open Street Maps 10, enquanto o dicionário de eventos foi retirado manualmente de tweets do perfil do Centro de Operações do Rio de Janeiro. Esta etapa foi destinada a obter tweets já etiquetados, poupando o trabalho de etiquetar todos os tweets do início. Com o corpo etiquetado pela técnica de etiquetamento por dicionário, bastou corrigir manualmente as falhas encontradas, transformando essa amostra

em um gabarito que pôde ser usado para treinar o modelo CRF. Como o intuito não era avaliar a precisão do método baseado em dicionário, não houve registros de Precisão, Recall e F-measure nesta etapa. Esta primeira amostragem deu origem ao primeiro modelo CRF treinado.

Uma outra amostragem de tweets foi retirada da base e etiquetada utilizando o modelo CRF treinado anteriormente. A partir desta etapa, somente o modelo CRF foi utilizado, o dicionário foi utilizado apenas como ponto de partida, como mencionado anteriormente. O corpo etiquetado pelo modelo passou por um processo de correção manual onde as palavras erroneamente etiquetadas foram corrigidas. Após isto, para se obter um resultado preliminar da precisão, recall e F-measure do modelo, a amostra corrigida foi utilizada para testar o modelo. O StanfordNER [Finkel et al. 2005] proporciona este mecanismo de teste, no qual uma amostra corrigida (gabarito) é dada como entrada e o modelo é utilizado para etiquetar aquela amostra e comparar com o gabarito, retornando as métricas de Precisão, Recall e F-measure separadas por entidade, neste caso localização e evento. Este é um método válido de testes, já que o modelo não tem conhecimento daquela amostra. Os testes foram feitos para obter dados do avanço gradual da qualidade do etiquetamento, enriquecendo os resultados.

Após a correção manual e a avaliação preliminar, os tweets corrigidos foram somados aos tweets utilizados para treinar o modelo anterior aumentando assim o conjunto de treinamento. Com este novo conjunto de treinamento um novo modelo foi treinado, em teoria com mais conhecimento que o anterior.

Este processo foi repetido no total de 9 vezes, gerando 10 modelos que foram gradualmente sendo enriquecidos. Em um dado momento do processo de treinamento, após o treinamento do modelo 6, foi percebido que o modelo não estava conseguindo identificar muitos eventos nas amostras utilizadas. Acredita-se que isto se deva à natureza esparsa dos tweets em si, poucos usuários relatam eventos de interesse comparado ao grande número de tweets onde somente comentam eventos pessoais, entre outros assuntos. Assim, foram coletados tweets de perfis do Twitter conhecidos por relatar eventos de trânsito, o perfil do Centro de Operações Integradas do Rio de Janeiro e da Linha Amarela, ambos administrados por entidades oficiais. Foram utilizadas amostras compostas exclusivamente de tweets destes perfis para preparar o modelo CRF para também identificar eventos e localizações em tweets com a estrutura utilizada por entidades oficiais, utilizando uma linguagem mais formal. Também, vários termos utilizados por estes perfis podem ser utilizados por pessoas comuns, sendo assim era esperado que a utilização destes tweets fosse aumentar a qualidade do etiquetamento.

A Tabela 1 resume a quantidade de tweets utilizadas em cada amostra e o total de tweets utilizado no processo de treinamento.

Amostra 1 (Dicionário)	4200
Amostra 2 (Geral)	5510
Amostra 3 (Oficial)	1370
Total	11080

Tabela 1. Números das amostras de treinamento.

Embora toda a base de tweets possuisse mais de quatro milhões de tweets, apenas uma pequena parcela destes possuia ou alguma localização, ou algum evento, ou ambos.

Para efeito de experimento a base de tweets como um todo foi etiquetada utilizando CRF e os tweets com alguma entidade foram contados totalizando apenas 173.012 tweets. Ainda assim, o total de tweets de treinamento pode ser considerado pequeno, porém, observando os bons resultados conquistados durante a análise dos resultados, foi possível concluir que este número de tweets no conjunto de treinamento foi suficiente para conseguir excelentes resultados. Estes resultados serão apresentados na seção a seguir.

5. Resultados

Os testes finais foram realizados utilizando três amostras diferentes para avaliar vários aspectos do etiquetamento. Foi percebido durante testes preliminares realizados que a identificação de eventos em tweets do público geral não é uma tarefa fácil devido ao conjunto de dados esparso, ou seja, há poucos tweets onde algum evento é mencionado em comparação ao grande número de tweets onde nenhum evento é mencionado. A identificação de localidades conseguiu resultados notáveis já que localidades são mencionadas mais frequentemente e houve oportunidade do modelo aprender a identificá-las com eficácia. Com o objetivo de obter resultados relevantes na identificação de eventos, foram extraídos tweets de perfis próprios para notificação de eventos de trânsito.

Os testes finais foram realizados com o objetivo de responder a seguinte pergunta. O treinamento com tweets de perfis oficiais melhora a performance de identificação de eventos em tweets do público geral?

Responder esta pergunta é importante para decidir o melhor caminho em casos de uso: se somente tweets de perfis oficiais serão usados, por exemplo, bem como concluir sobre a eficácia do etiquetamento em ambos os métodos: utilizando perfis públicos e oficiais. Assim, foram construídos três novos conjuntos de tweets para o treinamento final como dispostos abaixo: (1) Conjunto com tweets do público geral: 1993 tweets. (2) Conjunto com tweets de perfis oficiais: 1000 tweets. (3) Conjunto com tweets misto: 1018 tweets oficiais e 936 de perfis públicos, 1954 no total.

Amostra 1						
Entidade	**Precisão**	**Recall**	**F-Measure**	**TP**	**FP**	**FN**
EVENT	0	0	0	0	0	4
LOCATION	0.9505	0.9275	0.9389	192	10	15
Total	0.9505	0.91	0.9298	192	10	19

Amostra 2						
Entidade	**Precisão**	**Recall**	**F-Measure**	**TP**	**FP**	**FN**
EVENT	1	0.9645	0.9819	624	0	23
LOCATION	0.9797	0.9445	0.9618	1499	31	88
Total	0.9856	0.9503	0.9676	2123	31	111

Amostra 3						
Entidade	**Precisão**	**Recall**	**F-Measure**	**TP**	**FP**	**FN**
EVENT	0.9829	0.9198	0.9503	516	9	45
LOCATION	0.9735	0.9674	0.9704	1395	38	47
Total	0.976	0.9541	0.9649	1911	47	92

Tabela 2. Resultados dos testes finais.

Os três conjuntos foram confeccionados com o objetivo de responder a pergunta acima. Os tweets advindos de perfis públicos foram retirados da base de tweets anteriormente mencionada, já os tweets de perfis oficiais foram extraídos separadamente, entre o período de 10 de outubro de 2017 a 17 de outubro de 2017 para o segundo conjunto

e entre 2 de novembro de 2017 e 10 de novembro de 2017 para o terceiro conjunto. Os resultados dos testes podem ser observados na Tabela 2.

Os resultados obtidos nos três testes foram em sua maioria excelentes, exceto o resultado da identificação de eventos na Amostra 1, composta por tweets de perfis do público geral. Este resultado responde a pergunta no qual os testes foram baseados. De fato, em artigos onde foram utilizadas técnicas similares, resultados semelhantes foram obtidos. Em [Anantharam et al. 2015] foi obtido 68% e 50% de precisão para as etiquetas B-EVENT e I-EVENT e *recall* de 57% e 7%, respectivamente. É importante destacar que ao contrário dos testes realizados em [Anantharam et al. 2015], no presente trabalho os testes foram realizados levando em consideração as entidades como um todo, não separando por etiquetas. Assim, para comparar os resultados, foram geradas as matrizes de confusão apresentadas nas Tabelas 3, 4, 5 com o resultado de cada etiqueta separadamente, como em [Anantharam et al. 2015].

	B-LOCATION	**I-LOCATION**	**B-EVENT**	**I-EVENT**	**O**	**Total**	**Precision**
B-LOCATION	194	0	0	0	13	207	93.72%
I-LOCATION	0	303	0	0	20	323	93.81%
B-EVENT	0	0	0	0	4	4	0.00%
I-EVENT	0	0	0	0	0	0	N/A
O	8	15	0		18613	18636	99.88% 0
Total	202	318	0	0	18650	19170	
Recall	96.04%	95.28%	N/A	N/A	99.80%		

Tabela 3. Matriz de confusão referente à Amostra Final 1.

	B-LOCATION	**I-LOCATION**	**B-EVENT**	**I-EVENT**	**O**	**Total**	**Precision**
B-LOCATION	1503	5	0	0	79	1587	94.71%
I-LOCATION	7	1144	0	0	53	1204	95.02%
B-EVENT	1	0	622	0	22	645	96.43%
I-EVENT	0	0	0	637	28	665	95.79%
O	65	73	1	2	11317	11458	98.77%
Total	1576	1222	623	639	11499	15559	
Recall	95.37%	93.62%	99.84%	99.69%	98.42%		

Tabela 4. Matriz de confusão referente à Amostra Final 2.

	B-LOCATION	**I-LOCATION**	**B-EVENT**	**I-EVENT**	**O**	**Total**	**Precision**
B-LOCATION	*1398*	3	0	0	41	1442	96.95%
I-LOCATION	4	*1084*	0	0	35	1123	96.53%
B-EVENT	1	0	*513*	0	44	558	91.94%
I-EVENT	0	0	0	*530*	17	547	96.89%
O	73	79	28	9	*19607*	19796	99.05%
Total	1476	1166	541	539	19744	*23466*	
Recall	94.72%	92.97%	94.82%	98.33%	99.31%		

Tabela 5. Matriz de confusão referente à Amostra Final 3.

6. Caso de Uso

Uma aplicação web foi construída para demonstrar as capacidades do *framework*. Esta aplicação tem como funcionalidade mostrar os eventos e sua localidade em um mapa de fácil entendimento e interação para o usuário, utilizando informações contidas no tweet. A Figura 4 mostra a aplicação em funcionamento e exibe um mapa com os pontos de eventos espalhados por todo o território da cidade do Rio de Janeiro e região metropolitana. Para esta demonstração foram usados parte dos tweets utilizados no teste do modelo CRF

extraídos entre 2 de novembro a 10 de novembro de 2017, não sendo uma demonstração em tempo real. É possível ver que eventos ocorreram em toda a cidade neste período de tempo, com uma concentração deles, como era de se esperar, na região do Centro e Zona Sul, visto que o trânsito é mais denso nesta região, principalmente em dias úteis.

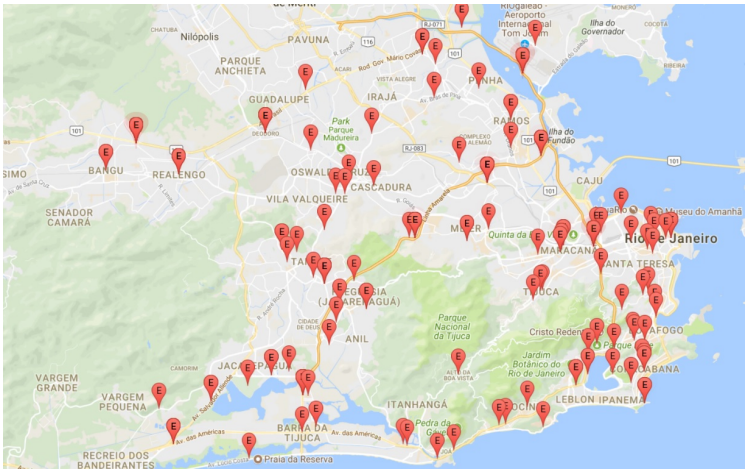
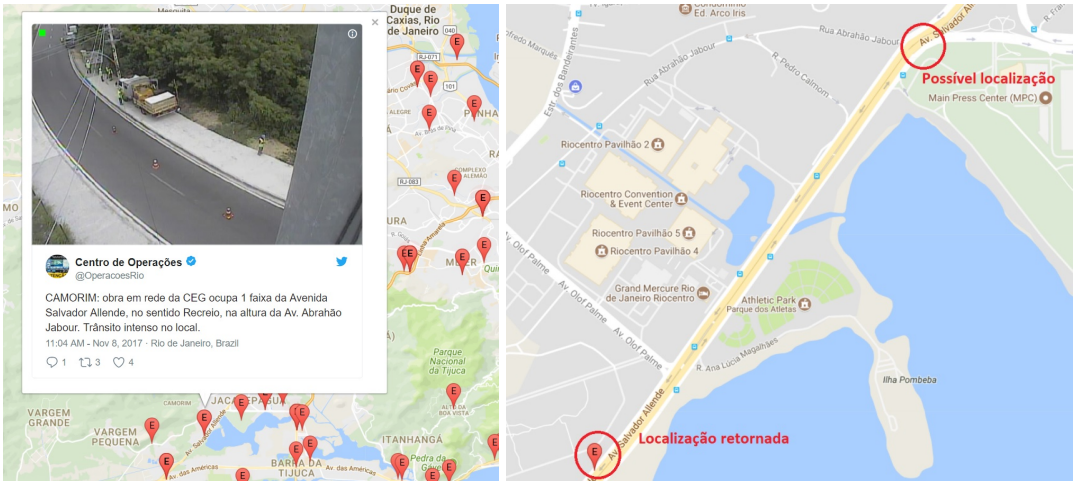


Figura 4. Visão geral da aplicação de mapa.



(a) (b)

Figura 5. Avaliação de precisão da API em (a) e em (b) a relação de um Tweet associado a um evento.

Na Figura 5 (a) é mostrado um exemplo de tweet que aparece ao clicar no marcador correspondente. Como dito anteriormente, as coordenadas dos marcadores são resultado de uma busca utilizando a Google Places API Web Service com base na informação retirada do tweet. Nesse exemplo, a API retornou a localização correta com sucesso com uma imprecisão tolerável. A Av. Abrahão Jabour (nomeada na Figura 5 (a) como Rua Abrahão Jabour) está alguns quilômetros afastada da localização retornada pela API como é possível ver na Figura 5 (b). Como eventos de trânsito costumam impactar o trânsito de uma região, esta imprecisão pode ser considerada aceitável. Porém, é possível obter uma precisão maior utilizando todas as informações que o tweet proporciona. No exemplo, a Av. Abrahão Jabour é mencionada no tweet. Por questão de simplicidade, o *framework* utilizou a primeira localização encontrada para obter as coordenadas. Mas, isto pode ser

melhorado utilizando as diferentes localizações presentes no tweet se houver mais de uma localização mencionada e, dessa forma, aumentar a precisão.

7. Conclusão

Este trabalho teve como objetivo construir um *framework* que proporcionasse a base para aplicações informativas sobre eventos ocorridos em cidades, neste caso o Rio de Janeiro. O *framework* proposto possui funcionalidades que permitem configurar e extrair tweets utilizando a API do Twitter, assim como classificar tais tweets utilizando CRF. Sendo assim possível realizar a classificação utilizando dados de qualquer cidade, desde que um modelo apropriado seja treinado para obter resultados mais precisos. Por fim, no que se refere à utilização de CRF para classificar e extrair as informações, os testes realizados comprovaram que o modelo treinado funciona muito bem na identificação de localizações em ambos tweets do público geral e em tweets oficiais, com o devido treinamento.

Referências

Anantharam, P., Barnaghi, P., Thirunarayan, K., and Sheth, A. (2015). Extracting city traffic events from social streams. *ACM Trans. Intell. Syst. Technol.*, 6(4):43:1–43:27.

Bajaj, G., Agarwal, R., Bouloukakis, G., Singh, P., Georgantas, N., and Issarny, V. (2016). Towards building real-time, convenient route recommendation system for public transit. In *2016 IEEE International Smart Cities Conference (ISC2)*, pages 1–5.

Dobbelaere, P. and Esmaili, K. S. (2017). Kafka versus rabbitmq: A comparative study of two industry reference publish/subscribe implementations: Industry paper. In *11th ACM DEBS*, pages 227–238.

Eugster, P. T., Felber, P. A., Guerraoui, R., and Kermarrec, A.-M. (2003). The many faces of publish/subscribe. *ACM Computing Surveys (CSUR)*, 35(2):114–131.

FarajiDavar, N., Kolozali, S., and Barnaghi, P. M. (2017). A deep multi-view learning framework for city event extraction from twitter data streams. *CoRR*, abs/1705.09975.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *43rd annual meeting on association for computational linguistics*, pages 363–370.

Neirotti, P., De Marco, A., Cagliano, A. C., Mangano, G., and Scorrano, F. (2014). Current trends in smart city initiatives: Some stylised facts. *Cities*, 38:25–36.

P Hancke, G. and Silva, B. (2012). The role of advanced sensing in smart cities. *Sensors (Basel, Switzerland)*, 13:393–425.

Puiu, D., Barnaghi, P., Tönjes, R., Kumper, D., Ali, M. I., Mileo, A., Parreira, J., Fischer, M., Kolozali, S., Farajidavar, N., Gao, F., ggena, T., Pham, T.-L., Nechifor, C.-S., Puschmann, D., and Fernandes, J. (2016). Citypulse: Large scale data analytics framework for smart cities. *IEEE ACCESS*, 4:1086–1108.

Twitter (2019). Twitter streaming api connection guidelines. `https://developer.twitter.com/en/docs/tutorials/consuming-streaming-data.html`. Accessed: 2019-03-24.

Zanella, A., Bui, N., Castellani, A., Vangelista, L., and Zorzi, M. (2012). Internet of things for smart cities. *Internet of Things Journal, IEEE*, 1.