Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Rumo a "Social Machines" sobre Dados Criminais

Eduardo Ferreira dos Santos

Dissertação apresentada como requisito parcial para conclusão do
Mestrado Profissional em Computação Aplicada

Orientadora
Prof.ª Dr.ª Fernanda Lima

Brasília
2015

Ficha catalográfica elaborada automaticamente,
com os dados fornecidos pelo(a) autor(a)

SS237r

Santos, Eduardo Ferreira dos
Rumo a "Social Machines" sobre Dados Criminais / Eduardo Ferreira dos Santos; orientador Fernanda Lima. -- Brasília, 2015.
113 p.

Dissertação (Mestrado - Mestrado Profissional em Computação Aplicada) -- Universidade de Brasília, 2015.

1. Social Machines. 2. Web Semântica. 3. Redes Sociais. 4. PNL. 5. Crowdsourcing. I. Lima, Fernanda , orient. II. Título.

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Rumo a "Social Machines" sobre Dados Criminais

Eduardo Ferreira dos Santos

Dissertação apresentada como requisito parcial para conclusão do
Mestrado Profissional em Computação Aplicada

Prof.ª Dr.ª Fernanda Lima (Orientador)
Departamento de Ciência da Computação/UnB

Prof. Dr. Rodrigo Bonifácio de Almeida
Departamento de Ciência da Computação/UnB

Prof. Dr. Claudio Gottschalg Duque
Faculdade de Ciência da Informação/UnB

Prof. Dr. Marcelo Ladeira
Coordenador do Programa de Pós-graduação em Computação Aplicada

Brasília, 30 de junho de 2015

Dedicatória

Dedico esse trabalho à vida. Durante anos refleti sobre a vida e seus objetivos, sobre o esforço e as conquistas, sobre a gratidão e a realização. Contudo, ao chegar no final de mais uma longa jornada, percebo finalmente a importância de percorrer o caminho. A vida não é sobre os objetivos alcançados, e sim sobre as pessoas que nos tornamos no processo.

Dedico ainda àqueles que estão no começo de suas jornadas: tenham certeza que, para quem acredita em si mesmo, o único caminho possível é o sucesso, e a ferramenta para atingi-lo é o trabalho.

Agradecimentos

Agradeço primeiramente a meu pai e minha mãe, por me proporcionarem educação, saúde e, acima de tudo, pela vida. Agradeço ao amigo Luis Felipe Costa, por ter me incentivado a fazer a inscrição no processo seletivo para o Mestrado, sem o qual não teria chegado até aqui. Agradeço também à minha orientadora e inspiração, professora Fernanda Lima, por ter acredito e me incentivado quando nem mesmo eu era capaz de fazê-lo. Mais do que o título de Mestre, agradeço pela lição de vida.

Finalmente, agradeço do fundo da alma à minha esposa e companheira Daniela Abreu. Mesmo quando o caminho parecia conduzir a um beco sem saída, com seu amor segurou em minha mão e me mostrou a direção correta.

Resumo

Social machine é uma abordagem relativamente nova para tratar problemas relevantes à sociedade, integrando em um software elementos computacionais e sociais. Pode ser considerada uma extensão da Web Semântica, criando o processo por meio do qual as pessoas executam as tarefas criativas e as máquinas realizam a administração dos dados. Essa Dissertação de Mestrado apresenta uma proposta de aplicação do conceito ao tema violência e criminalidade, assunto bastante relevante nos países da América Latina e Caribe – LAC. Trata-se de uma extensão do conceito de *Social Machines* que aplica duas estratégias publicadas recentemente para obter informações semânticas de dados oriundos de redes sociais, além de publicar o resultado como um serviço de Dados Abertos. O processo de desenvolvimento foi documentado para fornecer um procedimento sistemático, e uma aplicação exemplo foi construída para identificar eventos relacionados a violência e criminalidade. O procedimento proposto foi validado e testado em modelos formais recentemente desenvolvidos para o tema. Os resultados da extração de dados são também comparados aos dados oficiais, de forma a identificar similaridades.

Palavras-chave: Social Machines, Web Semântica, Redes Sociais, SRL, LDA, PNL, Crowdsourcing

Abstract

Social machine is a rather new approach to deal with relevant problems in society, blending computational and social elements into software. It can be an extension of the Semantic Web, creating processes in which people do the creative work and the machine does the data administration. This professional masters dissertation presents a proposal to apply this approach in violence and criminality domain, a relevant matter to Latin America and Caribbean – LAC – countries. It extends Social Machines by applying two recently published strategies to obtain semantics over social networks data and publishing it as a Linked Open Data service. The development procedure was documented to provide a systematic procedure and an example application is presented to identify violence and criminality events. The resulting procedure validation was be done by testing against recently developed formal models in the research area. Criminal activity data extraction results were also compared to official data, in order to identify similarities.

Keywords: Social Machines, Semantic Web, Social Networks, SRL, LDA, NLP, Crowdsourcing

Sumário

Lista de Figuras

Lista de Tabelas

Exemplos

Glossário

Crowdsourcing Reunir um conjunto de usuários em prol da construção de um artefato duradouro para o benefício de toda a comunidade.

Computação Humana Intervenção humana em sistemas computacionais para executar tarefas computacionais que não poderiam ser feitas por máquinas.

Computação Social Incorporação da computação humana como um componente algorítmico convencional em *social machines*.

Social Machines Entidade computacional que incorpora processos sociais e computacionais.

Capítulo 1

Introdução

Dados recentes mostram que a violência é o segundo maior problema para 18% da população do Brasil, logo após a saúde, que lidera para 45% da população [47]. A sensação de insegurança também é parte da vida dos cidadãos no Brasil e em outros países da América Latina e Caribe (LAC). É a região com a maior taxa de assassinatos no mundo. O estudo das Nações Unidas – UN – sobre cidades na LAC explica o problema através da desigualdade social [55][p.XII]:

> As cidades da América Latina e Caribe são duais, divididas, segregadas, e esse aspecto se expressa socialmente e espacialmente. (...) Ainda que o percentual da população vivendo abaixo da linha de pobreza tenha sido reduzido nas últimas décadas, o número total de pessoas aumentou para 111 milhões.

O mesmo estudo cita que os países da LAC "obtiveram avanços consideráveis na luta contra a pobreza nos últimos 10 anos", supondo que a condição de vida geral melhorou na região. Todavia, o número total de pessoas vivendo abaixo da linha de pobreza ainda é muito grande, representando uma de cada quatro pessoas nas zonas urbanas [55][p.39]. A população vive numa atmosfera de tensão social, que acaba por levar à violência.

Ainda que seja um problema importante para a população, somente nos últimos dez anos o governo brasileiro criou um sistema unificado para centralizar as atividades criminais. Antes de 2003, com a criação do Sistema Único de Segurança Pública – SUSP – "o gerenciamento das ações de policiamento e segurança se caracterizaram pela ausência de cooperação entre as organizações" [28]. As estatísticas dependiam da entrega manual dos dados pelas delegacias de polícia ao Governo Federal. Mesmo em países mais desenvolvidos como o Reino Unido, os dados criminais representam um problema real: "do ponto em que um crime realmente aconteceu até o fim do processo de registro de ocorrência, existem áreas onde os dados podem ser perdidos" [30].

O conceito de *social machines*, tecnologias da Web para tratar problemas reais da sociedade [61], pode ser um caminho para minimizar a ausência de dados. "Dados não

estruturados, partes não confiáveis e problemáticas, protocolos não escaláveis, todos representam características da Internet que têm evoluído nos últimos 40 anos" [50]. É necessário pensar na Web como uma plataforma de serviços conectados, e as redes sociais representam um nó importante no grafo dessa rede.

Em [16] os autores afirmam que "softwares sociais com base na Web (chamados coletivamente de 'Web 2.0' que consiste de blogs, redes sociais, compartilhamento de vídeo, etc) podem ser vistos como versões iniciais de Social Machines". Quando um incidente violento ocorre é provável encontrar reação nas redes sociais. Se for possível conceber uma tecnologia para a Web capaz de identificar e registrar o que os cidadãos estão falando sobre violência, inserindo a interação como um elemento computacional, tal tecnologia estaria realizando um trabalho de computação social [61].

O ato de produzir conteúdo nas redes com objetivo social pode ser considerado um tipo de *crowdsourcing*. A definição de *Crowdsourcing Systems* sugere arregimentar "uma multidão de usuários que colabore explicitamente para construir colaborativamente um artefato duradouro que será benéfico para toda a comunidade" [25]. Colaboração está no núcleo da atividade de redes sociais, e o "Twitter provou ser uma fonte de dados efetiva para identificar tópicos importantes e a reação do público" [4]. Um sistema com dados oriundos do Twitter, utilizando temas relacionados à violência e criminalidade, atende os modelos de computação social e *crowdsourcing*, formando a base para uma *social machine*.

Contudo, o Twitter é uma fonte de dados que deve ser utilizada com cuidado, devido à dificuldade de tratamento das informações. Experimentos recentes mostram que a aplicação do Processamento de Linguagem Natural – NLP – nos dados extraídos, principalmente a análise semântica e modelagem de tópicos (*topic modeling*), pode trazer melhores resultados do que uma simples busca por palavras nos *tweets* [72]. A ideia é aumentar a eficiência da busca colocando as palavras utilizadas no contexto da linguagem, permitindo a identificação de eventos e criação de um modelo de tópicos por categorias. Assim, ao identificar um dado, é possível calcular a probabilidade de o *tweet* extraído estar relacionado a um conjunto pré-definido de tópicos.

Alguns autores abordam ainda a criação de modelos preditivos com base nos dados oriundos de redes sociais, mas apontam dois importantes problemas: impossibilidade de determinar uma amostra representativa e influência deliberada nos resultados [30]. Algumas possíveis soluções para a questão envolvem a redução do escopo da análise, seja escolhendo contas específicas que tratam do tema a ser estudado [72], seja utilizando como entrada fontes previamente treinadas, com eficácia conhecida, no tema a ser estudado [19]. O modelo de *social machines* permite a utilização da computação social como uma alternativa: se for possível construir uma ferramenta que permita às pessoas classificar a acurácia dos dados fornecidos pelas redes sociais, os problemas podem ser mitigados.

A questão é abordada por [14], com foco na importância de fornecer melhores interfaces para que os cientistas sociais trabalhem nos dados. Apresentar informações estruturadas a partir de fontes legíveis apenas por humanos é o foco da técnica *Semantic Uplift*, definida como "o processo de converter dados não-RDF (...) em representação do conhecimento baseada em RDF". O objetivo é integrar os incidentes de violência política em recursos da Web Semântica, proporcionando a integração de outras fontes para adição de metadados e permitindo a construção dos modelos de análise pelo próprio cientista social. A ferramenta proposta pelo trabalho permite que o usuário escolha os termos a serem levados em consideração na classificação da informação, atualizando uma ontologia gerada automaticamente para fornecer diferentes visões sobre as mesmas fontes de dados.

A integração entre dados de diferentes fontes é possível principalmente pela iniciativa de Dados Abertos, ou "dados que podem ser livremente utilizados, reutilizados e redistribuídos por qualquer um" [24]. Ao disponibilizar informações em formato padronizado, legível tanto por homens quanto por máquinas, as aplicações para a Web Semântica permitem o cruzamento de informações de diferentes domínios. Utilizar modelos de licenciamento que privilegiem o compartilhamento irrestrito permite, por exemplo, a construção de aplicações verticais que trabalhem no mesmo domínio, com a mesma fonte de dados e forneçam visões diferentes. As *social machines* representam uma visão que incorpora e expande o espectro computacional das aplicações da Web Semântica, colocando na fila de processamento tarefas executadas tanto por homens quanto por máquinas. Descrever, implementar e avaliar o problema está entre os objetivos da pesquisa a serem descritos nas próximas seções.

1.1 Pergunta de pesquisa e validação

A teoria e prática de *social machines* está sendo abordada por diferentes autores. Na pesquisa teórica, há estudos propondo uma definição formal [52], construindo um mapeamento sistemático [16], propondo relações com outras áreas da computação [61], entre outros. No campo da pesquisa prática, há estudos desenvolvendo aplicações de exemplo [26] e discutindo possibilidades de implementação [30], entre outros.

No campo prático, uma pergunta interessante foi trazida recentemente por [50], ao propor uma álgebra descritiva para *social machines*: é possível desenvolver uma aplicação utilizando o modelo formal? Quais são os aspectos relativos à implementação, considerando benefícios e desafios? As perguntas ressaltam a importância de testar os modelos em diferentes cenários, tanto no que tange às restrições tecnológicas quanto no que tange às características de implementação. Responder a tais perguntas traz a primeira hipótese da pesquisa:

H_1. É possível definir um modelo arquitetural para o desenvolvimento de *social machines*.

A hipótese trabalha em conjunto com a proposta de utilização de análise semântica de dados oriundos de redes sociais como uma ferramenta de computação social [72]. Ao propor um modelo estruturado para a Ciência da Informação, o autor constrói um diagrama temporal que organiza a história da aquisição de conhecimento (*knowledge acquisition*) durante a evolução da humanidade. O tempo atual se encaixa na área definida como *Symbiosis Science* [36]. No contexto da aquisição de conhecimento, *Social machines* podem ser definidas ainda como "sistemas de aquisição de conhecimento em escala e máquinas que possuem contexto social".

Trabalhar com computação social requer um subconjunto de técnicas para lidar com as características que não são facilmente identificadas por computadores, tais como confiança, confiabilidade e comunicação em linguagem natural [41]:

> Trazer humanidade ao *loop* de informações requer estruturas de dados e técnicas computacionais que permitam o tratamento de expectativas sociais e regras legais como objetos de primeira classe na arquitetura da Web.

Ao pensar em redes sociais como a primeira geração de *social machines* que "incorporam componentes algorítmicos convencionais" [61] é possível conceber um novo sistema de regras. Tal sistema deve ser capaz de utilizar dados oriundos de redes sociais como entrada para reforçar a computação social.

Considerando que violência é um problema relevante na LAC e a importância do *crowdsourcing* na definição de *social machines*, a escolha de um problema social relevante como domínio da pesquisa pode facilitar a obtenção dos resultados. Assim é possível construir a segunda hipótese:

H_2. O modelo arquitetural, se existir, pode ser aplicado à implementação de uma *social machine* para identificar atividade criminal.

Como violência e criminalidade são assuntos importantes na LAC, se for possível desenvolver uma tecnologia para identificar atividades criminais, é possível também imaginar que uma demanda social está expressa. Para construir ferramentas de computação social deve ser utilizado um modelo de desenvolvimento simplificado [61]:

> As abordagens atuais de computação social, tais como *crowdsourcing* (juntar pessoas para resolver um problema), co-criação (fornecer um 'espaço social' para gerar conteúdo ou aumentar a produtividade) e compartilhamento em redes sociais (compartilhar informações através de conexões sociais), envolvem comparativamente softwares não muito sofisticados com envolvimento social intenso.

A complexidade da computação social é também abordada por [16], supondo que maior engajamento social pode surgir sem a necessidade de um suporte de software especial. Tratar problemas relevantes pode ser uma forma de aumentar o engajamento dos usuários, o aspecto mais importante de *crowdsourcing* [25]. É o caso em que parte do processamento é feito por humanos, reduzindo a quantidade de tarefas computacionais executadas pelos computadores. Um exemplo é a utilização de *hashtags* no Twitter para agrupar informações relacionadas [3]. A adição de um caracter especial antes das sentenças (#) não surgiu através dos programadores do Twitter: foi uma ideia dos usuários. A implementação específica que permitia o agrupamento em *hashtags* foi uma resposta para melhorar a experiência dos usuários e aumentar a colaboração. Em última análise, trata-se de um componente algorítmico realizado naturalmente pelas pessoas, reforçando o caráter de computação social e ajudando na identificação de demandas sociais.

O conjunto de hipóteses apresentado leva à seguinte pergunta de pesquisa:

Q_1. Existe um modelo arquitetural para a implementação de *social machines* utilizando dados de redes sociais? Se for possível definir um, pode ser utilizado na construção de um protótipo para identificar atividade criminal?

O objetivo geral dessa pesquisa é investigar uma proposta de modelo arquitetural para uma *social machine* no domínio de violência e criminalidade que utilize dados oriundos de redes sociais. A implementação de um protótipo funcional será utilizada para validar o modelo, além de fornecer suporte à construção de diferentes visões sobre o mesmo dado. A Tabela 1.1 apresenta a estratégia referente à pergunta de pesquisa, resultados e validação, segundo as diretrizes de pesquisa em Engenharia de Software. Do lado esquerdo da tabela estão os critérios de classificação, enquanto do lado direito estão os valores nos quais a proposta se encaixa.

Critério de Classificação	**Valor**
Tipo de pergunta de pesquisa	Estudo de viabilidade ou exploratório
Tipo de resultado de pesquisa	Solução específica, protótipo, pergunta ou avaliação
Tipo de validação	Exemplo

Tabela 1.1: Classificação da pesquisa segundo as diretrizes da Engenharia de Software [62]

Ainda que se trate de um trabalho prático, as estratégias de validação envolvem aspectos teóricos e práticos. A hipótese H_1 demanda a investigação de modelos formais para *social machines* e como eles se encaixam na implementação proposta. É possível encaixar a arquitetura construída nos modelos existentes? Como eles são diferentes? A hipótese H_2

afirma ainda que é possível identificar atividade criminal utilizando uma *social machine*. O objetivo não é construir uma ferramenta oficial para registro de ocorrências criminais, mas os eventos extraídos podem ser comparados aos dados oficiais publicados pelo SUSP para identificar similaridades. Sabendo que não é possível comparar os números absolutos, como é a acumulação de atividades criminais nas principais cidades e estados?

Também é necessário realizar uma comparação com estudos similares no que tange a identificação de atividades criminais utilizando dados de redes sociais. Como a análise semântica, implementada como uma *social machine*, pode ser aplicada aos modelos existentes? Utilizar um conjunto maior de informações apresenta resultados similares?

A Tabela 1.2 apresenta as hipóteses (H) e estratégias de validação (V).

Hipótese	**Validação**
H_1	V_1. Descrição da arquitetura a partir dos modelos formais para *social machines*
H_2	V_2. Comparação com outros modelos de identificação de atividade criminal e com os relatórios do SUSP para as principais cidades e estados

Tabela 1.2: Estratégia de validação das hipóteses

1.2 Objetivos e resultados esperados

Os objetivos da proposta podem ser descritos da seguinte forma:

Objetivo geral Investigar uma proposta de modelo arquitetural para uma *social machine* no domínio de violência e criminalidade que utilize dados oriundos de redes sociais.

Objetivos específicos Identificação de aspectos da implementação:

- G_1 Construir um protótipo funcional de *social machine* para identificar atividade criminal em dados oriundos de redes sociais, desenvolvendo um banco de dados como uma aplicação de dados abertos;
- G_2 Validar a aplicação da análise semântica em dados de redes sociais e a utilização de *crowdsourcing* para a identificação de atividades criminais.

Considerando os objetivos de pesquisa, um resumo dos resultados esperados está apresentado na Tabela 1.3. O resultado mais importante dessa pesquisa é definir um modelo arquitetural para utilizar dados de redes sociais na construção de *social machines*. Tal

Objetivo	**Resultado esperado**
Objetivo principal	Modelo arquitetural de *social machine* no domínio de violência e criminalidade.
G_1	R_1. Protótipo funcional de *social machine* implementada em um banco de dados de informações criminais desenvolvido como uma aplicação de dados abertos.
G_2	R_2. *Crowdsourcing system* construído a partir de modelos de identificação de atividades criminais.

Tabela 1.3: Resultados esperados

abordagem deve ser capaz de responder demandas sociais, trabalhando na identificação de eventos sociais.

Como um subproduto do processo de desenvolvimento, o objetivo G_1 demanda a validação das teorias e modelos formais de *social machines* aplicados à identificação de atividades criminais. Com o objetivo de "tratar cada aspecto de uma SM, como descrito formalmente, foi criada como uma linguagem de alto nível para descrição da arquitetura" [52]. Tal linguagem deve ser capaz de descrever as relações envolvidas no processo de desenvolvimento considerando a necessidade de composição. Contudo, como se trata de uma nova abordagem, os autores enfatizam a necessidade de mais casos de estudo para validação [26]:

> Como desdobramentos futuros temos a intenção de fornecer mais casos de estudo práticos de ambas as representações visuais e textuais da SMADL [1], utilizando-a para especificar sistemas da Web, tais como *crodsourced platforms*, combinando as API's populares já existentes, como facebook e Twitter, para adquirir e processar informação, criando assim sistemas sociais práticos.

Os modelos propostos por [52] e [26], junto com as teorias de [16] e [61] são as principais referências para a definição formal. O objetivo é testar o modelo proposto na construção de um banco de dados de informações criminais.

Ainda que não seja possível, dentro do escopo do Mestrado, produzir uma integração completa dentro do que se espera para a Web Semântica, o banco de dados produzidos deverá levar em conta temas como acesso, disponibilidade e uso dos dados produzidos na pesquisa, aproximando-se ao máximo da definição de dados abertos [24].

O último passo é construir um *crowdsourcing system* a partir dos modelos de identificação de atividades criminais, como apresentado no objetivo G_2. Um sistema desenvolvido para trabalhar com foco nas necessidades dos cidadãos deve levar em consideração os aspectos relativos à entrada de dados por parte dos usuários [25]:

[1] Acrônimo para *Social Machines Architecture Description Language*

> Eles normalmente misturam entradas do usuário, tais como classificação de filmes, utilizando fórmulas automáticas.

Assim, o modelo construído deve levar em consideração a opinião dos usuários, obtida através do processamento dos dados enviados para as redes sociais. No que diz respeito à identificação de atividade criminal, um comparativo entre os resultados apresentados em [72] and [71] com os dados obtidos na pesquisa deve ser realizado.

Os próximos capítulos estão descritos da seguinte forma: o capítulo 2 apresenta o histórico das aplicações para a Web, até os conceitos e definições de *social machines*. O capítulo 3 apresenta as estratégias de obtenção de informações de redes sociais, além de comentar sobre a análise de qualidade e dados criminais em geral. O capítulo 4 descreve a proposta de modelo arquitetural para desenvolvimento de *social machines* da pesquisa, comentando sobre a solução teórica, proposta de implementação e definição da arquitetura do sistema. Já o capítulo 5 apresenta os resultados e comenta sobre a validação, enquanto o capítulo 6 discorre sobre a experiência de desenvolvimento e o atendimento das estratégias de validação. O capítulo 7 traz as considerações finais, contribuições, restrições e apresenta possibilidades de trabalhos futuros.

Capítulo 2

Social Machines

O termo *Social Machines* trata de uma abordagem relativamente nova para descrever e desenvolver tecnologias computacionais com motivação social. O capítulo apresenta os conceitos e definições de *social machines*, iniciando com a Web Semântica para criar redes de colaboração e considerando problemas que são relevantes para a sociedade. Essa visão sobre as tecnologias Web introduz a interação humana como um processo computacional, tal qual descrito nos trabalhos relacionados apresentados no final do capítulo.

2.1 Web Semântica

Conhecimento pode ser definido como "o acúmulo de técnicas para resolver problemas específicos de cada geração", e tais técnicas podem ser definidas como tecnologias [36]. Como a aquisição, disseminação e utilização do conhecimento representam uma preocupação importante desde o começo da humanidade, novos sistemas de captura e armazenamento se tornaram cruciais. Na biblioteca de Alexandria, fundada por Ptolomeu 2400 anos atrás e considerada o primeiro repositório de conhecimento humano, as técnicas de catalogação e indexação utilizadas eram similares às utilizadas hoje [36]. Acessar o conhecimento armazenado se revelou um fator crítico para seu sucesso.

Desde que o fogo consumiu a biblioteca de Alexandria, o maior arquivo que fomos capazes de construir é a *World Wide Web*. Ao explicar a motivação por trás da criação do *Hiper Text Transmission Protocol* [6] – HTTP – os autores falam sobre a necessidade de um protocolo universal de visualização de informações:

> Existe um potencial muito grande na integração de diferentes sistemas de forma que os usuários sigam *links* apontando de um pedaço de informação para o outro. Formar uma rede de nós de informação ao invés de uma árvore hierárquica ou lista ordenada é o conceito básico por trás do *HyperText*.

Na época da idealização do protocolo HTTP era possível pensar que um documento fosse capaz de apontar para todos os outros possíveis documentos que tratassem do mesmo assunto. Contudo, o serviço *The Size of World Wide Web* [1] mostra que existiam pelo menos 4,65 bilhões de páginas no dia 12 de junho de 2015. Com toda essa quantidade de dados armazenados, o desafio de encontrar informações relevantes se revelou crítico. O problema foi parcialmente resolvido pelo Google e seu algoritmo de *PageRank* [15], mas "a maior parte do conteúdo da Web hoje é construído para ser lido por humanos, não para a manipulação conceitual de outros programas de computador" [8].

Na era da *Sysmbiosis Science* [36], começando com o fenômeno da Web 2.0 [56], novos sistemas de aquisição do conhecimento são necessários. Hendler e Berners-Lee defendem que "da mesma forma que comunidades humanas se conectam na sociedade, elas devem estar conectadas na Web" [41].

A base para a conexão entre as tecnologias sociais e computacionais está na visão inicial da Web Semântica [8]. De acordo com o W3C, Web Semântica é a Web de Dados, e o "o objetivo final da Web de dados é permitir que computadores realizem trabalhos mais úteis e desenvolver sistemas que auxiliem interações confiáveis na rede" [70]. A proposta é estender a Web para fornecer um significado bem definido para a informação, "permitindo que computadores e humanos trabalhem em cooperação". Com a semântica adicionada aos documentos da Web, é possível criar uma rede global de documentos sem se preocupar em como eles serão armazenados e distribuídos. O desafio de criar conexões associadas a tipo entre redes diferentes é definido como *Linked Data* [10].

2.2 Linked Data

Considerando a Web Semântica como a Web de Dados, "é importante que a enorme quantidade de dados disponíveis na Web esteja disponível num formato padrão, acessível e gerenciado por ferramentas da Web Semântica". As diretrizes de desenvolvimento para implementação de *Linked Data* estão definidas em quatro regras [5]:

1. Utilize URI's como nome para as coisas;
2. Utilize URI's HTTP de forma que as pessoas possam buscar pelos nomes;
3. Quando alguém procurar uma URI forneça informação útil que obedeça os padrões (RDF, SPARQL);
4. Inclua *links* para outras URI's, de forma que as pessoas possam descobrir mais coisas.

O *Resource Description Framework* [46] – RDF – é o formato utilizado para trocar documentos na Web Semântica. O novo formato é construído para fornecer a estrutura

[1] Disponível no endereço `http://www.worldwidewebsize.com/`

necessária para integração em larga escala na Web. Os conjuntos de dados que compartilham contexto comum são organizados em ontologias, definindo "um vocabulário comum para pesquisadores que precisam compartilhar informação em um domínio" [53]. À medida que as máquinas processam documentos no formato da Web Semântica, elas são capazes de adicionar automaticamente informações sobre o contexto disponíveis na Web de Dados.

O processo de conexão automática é feito através de URI's, ou *Uniform Resource Identifiers*. A teoria afirma que é possível identificar unicamente um documento da Web, onde todo o conteúdo relativo ao domínio de conhecimento estaria disponível. Também é utilizado para armazenar a referência original aos documentos, de forma que outras aplicações sejam capazes de saber de onde eles vieram originalmente. Ainda que a definição de URI esteja presente desde a concepção da Web, a visão contemporânea [20] utiliza *schemes* como *namespaces*: `http:` é um *URI scheme* representando o *namespace* `http`. Na Web Semântica, as ontologias são agrupadas em *namespaces*.

Com tais recursos é possível desenvolver aplicações que se comunicam umas com as outras. Tais aplicações trabalham no espaço de *Linked Data*, idealizado com a Web Semântica. "O formato dos dados, ontologias e softwares de processamento devem operar como uma grande aplicação na *World Wide Web*, analisando todos os dados puros armazenados em bancos de dados *online*, assim como todas as informações referentes a textos, imagens, vídeos e comunicações contidos na Web" [34]. Nos cenários de aplicação para a Web de Dados, "existem navegadores de *Linked Data* genéricos que permitem aos usuários começar navegando em uma fonte de dados e seguir os *links* para navegar em fontes de dados relacionadas" [10].

O crescimento das aplicações de *linked open data* foi representado pelo Projeto *Linked Open Data* – LOD – *Cloud* [22]. A nuvem é uma abstração do projeto *Linked Open Data*, cujos objetivos são definidos como [10]:

> O objetivo original e em desenvolvimento do projeto é iniciar a Web de Dados identificando conjuntos de dados existentes que estejam disponíveis com licenças abertas, convertendo-os para RDF de acordo com os princípios de *Linked Data* e publicando-os na Web.

A nuvem LOD representa as aplicações construídas com dados abertos, que se conectam uns com os outros de alguma maneira, seja por termos comuns na taxonomia ou por reuso dos dados. O projeto *Linked Open Data* rastreia as aplicações e mapeia os relacionamentos entre elas.

2.2.1 Dados abertos

No momento é importante dar ênfase ao termo **aberto**. Ainda que as aplicações sejam capazes de comunicar-se umas com as outras, não significa que tal comunicação seja feita livremente. Ao desenvolver utilizando a API do Google Maps [2] é possível acessar diferentes tipos de *Linked Data*, mas não significa que eles sejam abertos. A página de licenciamento da API [3] apresenta várias restrições para a utilização dos dados, tais como o número máximo de requisições e a necessidade de renderizar as informações obtidas exclusivamente no Google Maps. Não é possível utilizar a API para encontrar informações geográficas e renderizá-la em mapas privativos.

Considerando os problemas emergentes da utilização, armazenamento e visualização dos dados nas aplicações de *Linked Data*, a *Open Knowledge Foundation* [4] – OKFN – organizou um Livro de Dados Abertos [24] onde encontramos a definição: "dados que podem ser livremente utilizados, reutilizados e redistribuídos por qualquer um – sujeitos somente, no máximo, aos requisitos de atribuição da autoria e compartilhamento com a mesma licença (*sharealike*)". Para definir as fronteiras dos conceitos eles também organizaram a *Full Open Definition*. Para ser considerada aberta, a fonte de *Linked Data* tem que seguir diversos princípios, tais como:

Disponibilidade e Acesso Os dados devem estar disponíveis por inteiro e por não mais do que um custo de reprodução razoável, preferencialmente baixando pela Internet. Os dados também devem estar disponíveis num formato conveniente e modificável.

Reutilização e Redistribuição Os dados devem ser fornecidos sob termos que permitam o reuso e redistribuição sem a necessidade de se misturar com outros conjuntos de dados.

Participação Universal Todos devem ser capazes de utilizar, reutilizar e redistribuir – não deve haver discriminação sobre campos de atuação, pessoas ou grupos. Por exemplo, restrições 'não-comerciais' que impeçam uso 'comercial', ou restrições de uso para certos propósitos (somente em educação, por exemplo) não são permitidas.

Aplicações desenvolvidas para a Web Semântica com *Linked Data*, utilizando a *Full Open Definition* levam à implementação de *Linked Open Data*.

[2] A API do Google Maps é um serviço fornecido pelo Google onde é possível desenvolver aplicações para a Web com dados geográficos para serem renderizados no Google Maps. Mais informações podem ser encontradas no endereço da API: `https://developers.google.com/maps/`

[3] `https://developers.google.com/maps/licensing`

[4] *Open Knowledge Foundation* – OFKN – é uma organização sem fins lucrativos com base no Reino Unido cuja missão é promover os Dados Abertos no mundo. Maiores informações podem ser encontradas no endereço da organização: `http://okfn.org/about/`

2.2.2 Linked Open Data

Aplicações para *Linked Open Data* podem ser definidas como aquelas que obedecem às diretrizes apresentadas na *Full Open Definition* [24]. Um dos maiores desafios em tornar os dados disponíveis no formato aberto está no fato de que "muitos dos dados estão sendo gerados por camadas construídas sobre bancos de dados relacionais ou API's, e ainda precisam ser extraídos antes de agrupados ou analisados" [10]. Conhecendo o problema, e para incentivar o desenvolvimento de *Linked Open Data*, o modelo de cinco estrelas foi desenvolvido [5]:

1 estrela Disponível na web (em qualquer formato) **mas com uma licença livre para ser Dados Abertos**;

2 estrelas Disponível em dados estruturados passíveis de processamento por máquina (Excel ao invés de uma imagem de tabela, por exemplo);

3 estrelas Assim como o modelo de 2 estrelas, mais formatos não proprietários (CSV ao invés de Excel);

4 estrelas Todos os acima, adicionando a utilização de padrões abertos do W3C (RDF ou SPARQL) para identificar as coisas, de forma que as pessoas possam apontar para seus dados;

5 estrelas Todos os acima, adicionando: *links* dos seus dados ao de outras pessoas para fornecer contexto.

O modelo de cinco estrelas representa uma diretriz para iniciar a disponibilização como *Linked Open Data*. O trabalho foi revisado no Livro de Dados Abertos [40] descrevendo as receitas para publicação de *linked data*. As receitas podem ser agrupadas em três áreas [60]:

Conectando Utilize *links* RDF para permitir a conexão com outros conjuntos de dados;

Utilização de Vocabulários Prefira a utilização de vocabulários populares. Se for necessário fornecer seus próprios vocabulários, criando o que pode ser chamado de vocabulários proprietários, eles devem referenciar outros vocabulários conhecidos;

Provisão de Metadados Tente fornecer dados os mais auto-descritíveis possível publicando metadados. A licença deve ser um dos metadados fornecidos.

A análise detalhada da última versão da nuvem LOD [60] apresenta um crescimento de 271% no total de aplicações *Linked Open Data* entre 2011 e 2014, totalizando 188 milhões de triplas analisadas e 1014 conjuntos de dados. O fragmento de dados apresentado na Tabela 2.1 mostra um aumento expressivo nos conjuntos de dados de Governo. Em 2011 Brasil, Indonésia, México, Noruega, Filipinas, África do Sul, Reino Unido e Estados

Categoria	Datasets 2014	Percentual	Datasets 2011	Crescimento
Mídia	22	2%	25	-4%
Governo	183	18%	49	306%
(...)				
Total	1014	-	294	271 %

Tabela 2.1: Número de *datasets* por categoria [60] (fragmento)

Unidos assinaram uma Declaração de Governo Aberto, criando a Parceria para Governo Aberto [54].

A Declaração para Governo Aberto reforça a adoção de *Linked Open Data*, assim como os países signatários estão comprometidos a:

- Aumentar a disponibilidade de informações sobre atividades governamentais;
- Fomentar a participação do cidadão;
- Implementar os maiores padrões de integridade profissional nas administrações;
- Aumentar o acesso a novas tecnologias para abertura e auditoria.

O cenário supõe que governos e sociedade estão avançando na adoção de padrões e tecnologias de *Linked Open Data*. Contudo, "ainda houve muito pouco avanço no entendimento dessa nova capacidade: como elas realmente permitem a conexão da Web com as pessoas que irão utilizá-las?" [8] Conectar as pessoas na Web de dados é a lacuna a ser preenchida pelas pesquisas mais atuais.

2.3 Da Web Semântica às Social Machines

Analisando a enorme quantidade de informação disponível na Web, é necessário rever a maneira pela qual as pessoas interagem para criar aplicações numa camada superior de abstração do que apenas tabelas em um *website*. O novo modelo deve se comportar como um "grafo global de pessoas e ideias interconectadas" [41], além de permitir a conexão entre processos sociais e computacionais.

O cenário descrito cria um problema com "a segmentação dos dados e os problemas relativos à comunicação entre os sistemas" [50]. Ainda que o movimento de Dados Abertos tenha se iniciado com a motivação de disponibilizar informações, "nenhum dos recursos é minimamente bom sem esquemas de contexto que permitam a interpretação sobre seu significado" [61]. Algum tipo de mapeamento entre dados processados por computador e as demandas sociais precisa acontecer, sugerindo a necessidade de criação de um novo tipo de computação com contexto social. O termo **Social Machines** é introduzido para

descrever "um crescimento evolucionário dos motores sociais" [41] como o próximo passo das aplicações da Web Semântica.

2.3.1 Definições de Social Machines

As primeiras menções ao termo social machines surgiram em meados dos anos 2000 [7], mas um marco importante foi a introdução de redes sociais e *linked data* [41]. Um modelo conceitual mais expressivo está sendo construído [26], junto com um matemático [50] e outro computacional [17]. Os esforços realizados até o momento têm o objetivo de responder a seguinte sentença [36]: seriam tais tecnologias capazes de ajudar o cidadão comum?

Em junho de 2013, quando a Copa das Confederações da FIFA começou, o Brasil presenciou um levante popular que explodiu em uma onda de protestos que começou na região Sul e se espalhou para todos os cantos do país. O maior fator de impacto é que nenhuma das autoridades e instituições do país foi capaz de prever o que estava por vir, já que "as mídias sociais aumentam a taxa de transmissão de um protesto através de sociedades suscetíveis" [49]. Mackenzie fala de uma rede sem liderança e sem hierarquia baseada em auto-organização. De fato, ainda que os protestos sugiram uma falta de organização política, não significa que eles não tenham consciência política. Um estudo analisando as localizações geográficas dos participantes dos protestos no Twitter [3] defende, entre outras hipóteses, que os usuários focaram sua participação *online* nas regiões mais ricas. Os dados apontam alguma consciência sobre a importância de mirar seus protestos nas estruturas políticas de poder.

O exemplo sugere que algum tipo de intervenção humana para atender necessidades sociais deve ser adicionado à troca de dados em redes sociais, ainda mais se forem considerados *mashups* com *linked data*. É possível definir o núcleo das *social machines* em várias estágios [61], situando a definição como um sistema de aquisição de conhecimento (*knowledge acquisition system*):

Crowdsourcing Usuários são capazes de definir a relevância das informações por si mesmos, além de compartilhar seus *rankings* pessoais com os outros;

Computação social O colaboração é introduzida de volta na aplicação de alguma forma e auxilia outros usuários.

Crowdsourcing é normalmente o primeiro passo da computação social. Para construir *social machines* de acordo com a definição citada anteriormente, é possível desenvolver aplicações capazes de introduzir informações relevantes como um serviço para auxiliar outros usuários a tratar suas demandas sociais. Outros autores contribuíram para a noção

de *social machines* realizando um estudo de mapeamento sistemático [16]. As definições apresentadas são organizadas em visões e introduzem um aspecto importante: softwares como entidades sociáveis. Com o objetivo de definir um modelo algébrico em três visões, os seguintes conceitos foram propostos:

Pessoas Definidas como unidades computacionais, a visão de pessoas envolve tarefas e comportamentos possíveis somente a seres humanos, tais como *crowdsourcing*;

Software social Tecnologias para a Web que envolvem algum tipo de troca de informação entre usuários, tais como redes sociais da Web 2.0 e plataformas de API com Twitter e facebook;

Associação de software Softwares e serviços para a Web que contêm uma ou mais camadas de integração com outros softwares sociais, tais como eles mesmos, outras aplicações de *linked data* ou redes sociais com outros *webservices.*

O modelo de três visões permite pensar em *social machines* como um "modelo abstrato para descrever sistemas de informação para a Web que podem ser uma forma prática de lidar com a complexidade da emergente Web programável" [26]. Elas também representam as bases para o desenho e implementação de *social machines*.

2.3.2 Implementação de Social Machines

A equação 2.1 representa uma definição formal que pode ser parte das diretrizes de desenvolvimento [50]:

$$SM = < Rel, WI, Req, Resp, S, Const, I, P, O > \tag{2.1}$$

A equação representa um modelo mental para a Web como plataforma, descrevendo as relações entre os serviços conectados. A descrição é apresentada pelos seguintes elementos:

SM *Social Machine*

P Internal *Processing Unit* para SM;

WI *Wrapper Interface* que espera por *Req* e envia *Res* para outras *social machines*;

Req *Requests* enviados para a SM;

Res *Responses* enviadas pela SM;

I *Inputs* recebidas pelas unidade de processamento P;

O *Outputs* produzidas pela unidade de processamento P;

S *States* permitidos para a SM;

Rel *Relationships* permanentes ou intermitentes com outras *social machines*;

Const *Constraints* para os *Rel* estabelecidos.

Uma tentativa de validar a definição é apresentada pela aplicação *WhatHere* [26], que funciona como uma composição de *social machines*. O trabalho representa uma das bases para o *Social Machines Architectural Style* [17] – SoMar – que define "uma série de restrições e diretrizes de desenvolvimento que orientam o desenvolvimento de software com o objetivo de satisfazer diferentes atributos de qualidade". O estilo é baseado nos mesmos princípios discutidos nas seções anteriores:

Relacionamentos Uma *Social Machine* deve possuir diferentes tipos de relacionamento, considerando as interações em diferentes níveis com outras *social machines* e tecnologias Web;

Neutralidade A *Social Machine* deve fornecer serviços capazes de abstrair detalhes específicos referentes à implementação;

Transparência A conexão entre processos humanos e computacionais deve ser o mais transparente possível.

O modelo é descrito incrementalmente em quatro diretrizes [17]:

1. Definir blocos de desenvolvimento;
2. Especificar serviços;
3. Desenhar integrações
4. Desenhar modelos de interação.

Enquanto *Social Machines* representam uma nova abordagem para o desenvolvimento de software, ainda há uma lacuna na validação do modelo de desenvolvimento em outros domínios. Contudo, fornece ferramentas interessantes para começar a construção de um novo sistema que trabalhe com software social.

2.4 Trabalhos relacionados

Ainda que haja uma discussão em curso sobre *social machines*, "não há estudo de mapeamento sistemático caracterizando a área de Social Machines como um todo" [16]. Tal mapeamento, realizado pelos autores, tenta preencher a lacuna e definir "um novo paradigma para o desenvolvimento de software". Uma definição formal do mesmo grupo está descrita na seção 2.3.1. Como prova de conceito da álgebra proposta, o grupo apresenta a aplicação exemplo *WhatHere* [26], construída como uma composição de *social machines*. A aplicação deve funcionar como uma porta de entrada para as informações extraídas de redes sociais populares, "para ajudar as pessoas que não estejam familiarizadas com uma cidade ou área geográfica (...) a reunir informações sobre locais próximos".

Para o exemplo as redes sociais foram tratadas como *social machines*, extraindo as seguintes informações:

- Visualização do mapa do Google Maps;
- Foursquare e/ou Google Places para informações sobre os locais;
- Wikipedia e/ou Google Search para informações textuais;
- Flickr para fotos;
- Twitter para comentários em tempo real.

A aplicação proposta preenche os requisitos, uma vez que turismo é um negócio relevante para a economia brasileira. Como exemplo de validação da álgebra, os autores criaram duas classes de *social machines* – Internas e Externas – e chegaram ao modelo simplificado da equação 2.2:

$$SM =< Rel, Req, Resp, S, Const > \tag{2.2}$$

Outra aplicação de *social machine* no domínio de criminalidade analisa a aplicação de *Linked Data* disponível no endereço `www.police.uk` e apresenta os conceitos da aplicação *Risk Alert* [30]. Enquanto o primeiro representa um serviço fornecido pelo governo do Reino Unido aos cidadãos, o segundo propõe a utilização de *crowdsourcing* para identificar atividades criminais. Tratam-se de aplicações complementares que agem na falta de informações criminais por parte do governo, mas que apresentam uma série de restrições sociais. Os autores concluem que "as consequências indesejadas de implementar uma máquina que tenta 'combater o crime' podem ser inaceitavelmente altas, a menos que as ramificações das 'restrições sociais' da utilização de tecnologia para solução de problemas sejam consideradas mais a fundo".

A discussão sobre as restrições sociais deixa muitas questões em aberto: como pode "ser utilizada tal máquina, por exemplo, na Europa e na Ásia?". Quais são as restrições legais? Como garantir anonimidade? Como podem ser tratados fenômenos da Internet como *trolls*, relatos falsos e vigilantismo? Em uma frase: "é possível que ela seja capaz de resolver mais problemas do que criou?" Essas interessantes perguntas reforçam a necessidade de redução do escopo da pesquisa. A maior parte delas permanece sem resposta e não pode ser ignorada.

Um modelo baseado mais na Web Semântica e *Linked Data* é apresentado através da definição de um vocabulário de violência política [14]. O exemplo apresenta um arquitetura de software onde é possível aumentar o vocabulário adicionando novos termos à medida que aparecem. Contudo, é necessário passar por um processo de coleta de dados primeiro, sugerindo a necessidade de trabalhos futuros para estender a funcionalidade da ferramenta de extração de dados.

Mais exemplos podem ser encontrados se as aplicações Web 2.0 forem consideradas como a primeira geração de *social machines*. Wikipedia, Galaxy Zoo e até mesmo Twitter e Facebook estão no mesmo domínio [61], já que utilizam algum tipo de computação humana. Todavia, a tecnologia desejada deve ser mais próxima dos cidadãos, além de permitir o desenvolvimento de uma arquitetura que seja reutilizável para diferentes domínios.

Capítulo 3

Estratégia para obtenção de informações em dados de redes sociais

A comunicação em redes sociais acontece através da troca de mensagens em linguagem natural. Assim, o processamento e análise do conteúdo trocado pelos usuários está no cerne do trabalho de análise de dados. Este capítulo apresenta técnicas de Processamento de Linguagem Natural – NLP (*Natural Language Processing*) – para tratamento de dados oriundos de redes sociais organizadas em etapas, que serão cobertas das seções 3.3 a 3.4. O processo envolve a etapa de extração da informação, processamento de linguagem natural para tratamento dos dados encontrados e análise da qualidade dos dados. A seção 3.5 encerra o capítulo tratando de obtenção de dados criminais e seus desafios, citando exemplos de Brasil e Estados Unidos.

3.1 Extração de informações de redes sociais

A extração de informação em dados de redes sociais depende da disponibilidade de uma camada de busca que forneça uma visão do que está sendo enviado pelos usuários. No caso do twitter estão disponíveis duas formas de consulta aos dados gerais:

- API de busca (*Search API*);
- API do fluxo de atividades público (*Public Stream API*).

A busca tem como objetivo fornecer uma coleção de documentos relativa a um conjunto conhecido de parâmetros. Ao se enviar uma consulta por termo, espera-se que os *tweets* [1]

[1]Tweet é o nome dado às publicações realizados pelos usuários no Twitter. Trata-se de um texto de no máximo 140 caracteres, com links, fotos e vídeos agregados.

retornados como resultados estejam relacionados. Os resultados são organizados em ordem de relevância, o que pode ser reforçado por parâmetros específicos da API [67]. Importante ressaltar que os resultados não possuem foco na completude, e sim num conjunto específico para atender os critérios de busca enviados.

Já na API do fluxo de atividades público [66] o objetivo é oferecer uma "fotografia" do que acontece em tempo real no Twitter. Os desenvolvedores tentam disponibilizar sempre um volume aproximado de 1% do total de mensagens enviadas pelos usuários como resultados. Além do foco em um conjunto reduzido de informações que represente o total, o fluxo de atividades público também possui uma grande diferença arquitetural: enquanto a busca trata do envio de requisições e obtenção de resultados, o fluxo de atividades necessita de uma conexão permanente ao *endpoint* do Twitter que envia os dados à medida que vão sendo introduzidos na rede social pelos usuários.

Ambas as API's trabalham com o conceito de **atualização de status** (*status updates*), ou simplesmente *tweets*. Conhecer os dados e seus formatos é importante, mas como ressaltado na documentação dos objetos do Twitter [69], "a adição de novos campos ou alteração dos parâmetros de ordenação deve ser tolerada". Os *tweets* possuem um conjunto de metadados associados, que podem mudar dependendo das buscas realizadas e conjunto de dados retornados. Assim, qualquer sistema que utilize dados oriundos do Twitter deve possuir flexibilidade para tratar a adição e tratamento de novos metadados.

Em ambas as API's o resultado retornado é uma coleção de atualizações de status, disponíveis no formato JSON. Como o volume de informações inseridos pode ser demasiadamente grande, elas também possuem limites relativos à quantidade de requisições que podem ser enviadas e, no caso da busca, o total de *tweets* que podem ser retornados na coleção de resultados. Na busca o limite varia entre 180-450 requisições por janela de 15 minutos, dependendo do tipo de autenticação utilizada. Já no fluxo de atividades público o limite está restrito a dois critérios:

1. Filtros da busca;

2. Total de atividades no momento.

Embora os desenvolvedores procurem fornecer sempre um valor próximo de 1% do total de *tweets* no número de resultados, podem ocorrer oscilações dependendo dos filtros utilizados. A quantidade de informações recolhidas pode chegar a alguns milhares de *tweets* por hora, então é papel do desenvolvedor escolher o conjunto de regras que mais se adapte às suas necessidades.

Uma última observação importante diz respeito à disponibilidade das informações. Ainda que os dados estejam disponíveis através de múltiplas interfaces de visualização

e até exportação, os termos de uso do Twitter restringem sua livre utilização [68]. As restrições são relativas ao uso, disponibilização, visualização e análise das informações.

3.2 Processamento de Linguagem Natural

Processamento de Linguagem Natural – NLP [2] – pode ser definido como "qualquer tipo de manipulação por computador da linguagem natural" [9]. Por linguagem natural entendemos a comunicação realizada entre seres humanos, que não pode ser facilmente definida por regras explícitas como as linguagens de programação ou notações matemáticas. A comunicação em linguagem natural evolui com o tempo e possui características distintas para cada população, tornando ainda mais difícil encontrar um conjunto universal de regras e significados.

Em qualquer linguagem de programação o texto escrito em linguagem natural nada é mais do que um tipo de dado, que a princípio não tem significado. É possível pensar ainda num texto como um conjunto de palavras, e nesse ponto é possível aproximar as regras e definições formais da comunicação em linguagem natural. Sabemos das regras da língua portuguesa que, para a comunicação acontecer com sucesso, é necessário utilizar um conjunto de palavras que seja válido na linguagem: a isso chamamos de **vocabulário**. Em uma definição mais completa, podemos dizer que o vocabulário é o conjunto de palavras válidas em uma determinada linguagem ou contexto da linguagem.

Para construção do vocabulário utilizamos uma técnica chamada de *tokenização* [3], demonstrada no exemplo 3.1. Para os exemplos utilizamos um *framework* da linguagem Python chamado NLTK (*Natural Language Toolkit*), um conjunto bastante abrangente de ferramentas para NLP [9].

Exemplo 3.1: Exemplo básico de tokenização

```
Python 2.7.8 (default, Sep  9 2014, 22:08:43)
[GCC 4.9.1] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> from nltk import word_tokenize
>>> text = u"O rato roeu a roupa do rei de Roma"
>>> sent = word_tokenize(text)
>>> sent[:10]
[u'O', u'rato', u'roeu', u'a', u'roupa', u'do', u'rei', u'de', u'Roma']
```

No exemplo 3.1 o texto "O rato roeu a roupa do rei de Roma" é convertido em uma lista de palavras, ou *tokens*. Uma primeira regra de processamento para validação do vocabulário poderia consistir no seguinte algoritmo:

1. Leia sequencialmente cada palavra da lista de palavras;

[2] Acrônimo do nome em inglês: *Natural Language Processing*
[3] Do original em inglês *tokenization*

2. Se a palavra é válida na língua portuguesa, armazene no vocabulário;

3. Caso não seja válida, descarte.

Agora vamos supor a adição de um novo texto em nossa análise, como demonstrado no exemplo 3.2. Adicionamos o texto `Frangos fritos finos` e aplicamos as regras de tokenização. Se construirmos uma lista composta dos textos `text` e `text2` estamos criando um primeiro conjunto de **documentos**. Trata-se de uma coleção dos textos que serão utilizados na análise.

Exemplo 3.2: Exemplo básico de criação de corpus

```
>>> text2 = u"Frangos fritos finos."
>>> sent2 = word_tokenize(text2)
>>> sent2[:10]
[u'Frangos', u'fritos', u'finos', u'.']
>>> documents = [text, text2]
>>> documents
[u'O rato roeu a roupa do rei de Roma', u'Frangos fritos finos.']
>>> corpus = [sent, sent2]
>>> corpus
[[u'O', u'rato', u'roeu', u'a', u'roupa', u'do', u'rei', u'de', u'Roma'], [u'Frangos', u'
    fritos', u'finos', u'.']]
```

Um outro conceito importante apresentado no exemplo 3.2 é a construção de **corpus**. Se o vocabulário é o conjunto de palavras válidas em um processo de comunicação por meio da linguagem natural, o corpus é o conjunto de palavras que vai constituir o objeto da análise. Se os textos `text` e `text2` constituem a coleção de documentos, o corpus é constituído pelos conjuntos de tokens `sent` e `sent2`.

O procedimento de análise e as técnicas utilizadas em cada etapa dependem do tipo de informação que se deseja extrair do texto. Voltando ao exemplo do ponto final(.), que foi removido do corpus, vamos supor o caso em que é necessário identificar o fim de uma sentença. Nesse caso, a presença do ponto é bastante importante, da mesma forma que utilização de vírgula (,) para conectar duas sentenças e outras ferramentas da língua portuguesa que possuem a mesma função. Pode ser necessário entender o **contexto** de cada palavra do vocabulário, onde será necessário entrar no domínio da **semântica**.

3.2.1 Semantic Role Labeling

A técnica conhecida como *Semantic Role Labeling* – SRL – é uma tarefa do processamento semântico da linguagem onde "para cada predicado em uma sentença, o objetivo é identificar todos os constituintes que possuem papel semântico, além de determinar cada um dos seus papéis (agente, paciente, instrumento, etc) e seus complementos (advérbios de tempo, lugar, etc)" [38]. Os argumentos "têm a capacidade de identificar quem fez o quê com quem, quando, para quê e como" [35].

À medida que o trabalho de identificação dos verbos em uma linguagem e seus constituintes semânticos evolui, uma notação que unifique os resultados se torna importante; daí surgiu o Projeto Propbank [42]. Seu objetivo é construir uma notação única para que os autores possam publicar os resultados do mapeamento dos verbos com seus possíveis constituintes semânticos. O exemplo 3.1 demonstra a aplicação de SRL em uma sentença genérica da língua portuguesa no formato proposto pelo Propbank.

$$[_{Arg_0} \text{ O rato }] \text{ roeu } [_{Arg_1} \text{ a roupa do rei de roma }] \tag{3.1}$$

Vemos que o verbo `roeu` recebe dois argumentos: `O rato` e `a roupa do rei de roma`.

A página do Projeto [4] aponta uma quantidade significativa de verbos já anotados na sintaxe proposta para a língua inglesa, mas somente nos últimos cinco anos se iniciou um trabalho de anotação dos verbos para a língua portuguesa; trata-se da base Propbank.br [27].

A utilização de anotação semântica em dados oriundos de redes sociais foi explorada em [72]. O trabalho descreve a utilização de SRL para processamento de *tweets*, tentando encontrar "eventos mencionados nos *tweets*, as entidades envolvidas e o papel das entidades com respeito aos eventos". O objetivo é reduzir o escopo da análise ao criar corpus com dados oriundos do Twitter. Os autores defendem a hipótese de que, ao buscar eventos identificados através de SRL relacionados com acidentes no trânsito, poderia haver uma correlação com eventos futuros no mesmo tema. O conjunto da análise trabalhou somente com o caso em que ocorre atropelamento ou colisão seguido de fuga, ou seja, o motorista envolvido no acidente foge do local da batida. A ideia é que uma busca orientada aos eventos teria resultados melhores na identificação de atividades criminais do que uma simples busca por tokens.

A identificação de eventos é apresentada pelo seguinte exemplo:

ALERTA DE TRÂNSITO: Rua 20 fechada por causa de carro quebrado

Após a aplicação do SRL obtemos a seguinte sentença:

$$[_{e_1}\text{ALERTA}][_{e_1:aviso}\text{DE TRÂNSITO}] :$$
$$[_{e_2:entidade}\text{Rua 20}][_{e_2}\text{fechada}][_{e_2:causa}\text{por causa de carro quebrado}]$$

Os eventos e_1 e e_2 foram identificados, significando respectivamente `ALERTA` e `fechada`. É possível encontrar ainda uma entidade (`Rua 20`), que pode ser utilizada em análises posteriores para identificação de outros eventos relacionados, por exemplo.

[4] Disponível no endereço `https://verbs.colorado.edu/propbank/propbank-status-en.html`

A aplicação da técnica de SRL em língua portuguesa utilizando a base Propbank.br é possível através do módulo nlpnet [35]. Desenvolvido na linguagem Python e integrado ao *framework* NLTK, o módulo permite a identificação dos constituintes semânticos de sentenças em português, utilizando a notação proposta pelo Propbank. Para exemplificar o funcionamento do módulo vamos aplicar SRL através do nlpnet no tweet da figura 3.1.

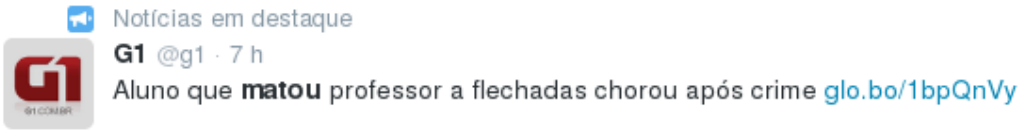


Figura 3.1: Tweet extraído em 21/04/2015

Exemplo 3.3: SRL aplicado ao *tweet* importado

```
>>> import nlpnet
>>> tagger = nlpnet.SRLTagger()
>>> exemplo = "Aluno que matou professor a flechadas chorou óaps crime http://glo.bo/1
    bpQnVy"
>>> srl_tag = tagger.tag(exemplo)
>>> pprint(srl_tag[0].arg_structures)
[(u'matou',
  {u'A0': [u'que'],
   u'A1': [u'professor'],
   u'A2': [u'a', u'flechadas'],
   u'V': [u'matou']}),
 (u'chorou',
  {u'A0': [u'a', u'flechadas'],
   u'AM-TMP': [u'ap\xf3s', u'crime', u'http'],
   u'V': [u'chorou']})]
```

A aplicação do nlpnet pode ser vista no exemplo 3.3. Como o *tweet* continha dois verbos (`matou` e `chorou`), a análise semântica foi feita com base nos elementos relacionados a eles. Se considerarmos somente os eventos, temos os seguintes dados:

- Verbo `matou`

  A0 que

  A1 professor

  A2 a flechadas

- Verbo `chorou`:

  A0 a flechadas

  AM-TMP após o crime

Percebemos que a análise semântica nos fornece informações sobre o tempo (`após o crime`) e consegue identificar um dos envolvidos(`o professor`). Os autores reportam uma eficácia decrescente na identificação dos eventos, que varia entre 74,17% para o primeiro (A0), decaindo para 35,29% para o terceiro (A2). A partir do terceiro a eficácia cai para zero e não é mais possível identificar os eventos.

3.3 Recuperação de Informação

A área de Recuperação de Informação – IR [5] – trabalha com "representação, armazenamento, organização e acesso à informação" e teve originalmente, como seu principal objetivo, "indexação e busca em coleções de documentos" [2]. Pode-se dizer que a palavra relevância está no coração da disciplina, pois mais importante do que encontrar uma coleção abrangente de documentos é conseguir interpretar os resultados para informar ao usuário o quão próximo estão de suas reais necessidades.

À medida que a quantidade de informações disponíveis na Internet aumenta, torna-se mais difícil encontrar o que se está procurando, reforçando a necessidade por ferramentas computacionais que nos permitam organizar, pesquisar e entender os dados armazenados. Não há dúvidas que ferramentas como o Google e seu algoritmo de PageRank [15] facilitam bastante o trabalho, à medida que fornecem um conjunto de documentos relacionados às buscas. Ao selecionar os resultados é possível então navegar pelos links encontrados até encontrar a informação desejada.

Contudo, é possível imaginar um cenário diferente: "se ao invés de encontrar os documentos através de uma busca por palavra-chave (*keyword*) fosse possível encontrar o tema em que estamos interessados, para só então encontrar os documentos relacionados ao tema?" [11] A analogia pode ser reduzida a um escopo menor, onde já conhecendo a organização de determinado site ou portal, seja possível navegar pelos temas de que tratam seus conteúdos, sem a necessidade de uma classificação prévia por parte de seres humanos. Os pesquisadores da área de *Machine Learning* e IR desenvolveram uma série de algoritmos na área de modelagem probabilística de tópicos (*probabilistic topic modeling* em inglês) que têm como objetivo processar bases muito grande de documentos para encontrar informações temáticas.

Se tal organização fosse possível, as técnicas descritas na seção 2.2 para Dados Conectados (*Linked Data*) poderiam ser levadas a outro nível: conectar conteúdos com temas semelhantes em diferentes portais, mais uma vez sem a necessidade de intervenção humana. A adição da análise de tópicos à classificação em taxonomias representa o elemento capaz de conectar ambas as tecnologias.

[5] Do acrônimo em inglês *Information Retrieval*

3.3.1 Latent Dirichlet Allocation

Dentro da modelagem de tópicos, *Latent Dirichlet Allocation* – LDA – é um dos modelos mais utilizados. Para entender seu funcionamento é preciso antes delimitar o conceito de **tópico**: considerando um vocabulário fixo de palavras, um tópico é uma "distribuição sobre um vocabulário fixo". É esperado que um tópico sobre futebol apresente uma probabilidade bastante alta de palavras relacionadas ao futebol. Imaginando a coleção de documentos que forma um corpus já definida na seção 3.2, é possível apresentar uma definição de LDA utilizando os conceitos de documentos e tópicos [12]:

> LDA é um modelo construtor probabilístico para um corpus. A ideia básica é de que documentos são representados como uma distribuição aleatória nos tópicos latentes, onde cada tópico é caracterizado por uma distribuição sobre o total de palavras.

O modelo matemático parte do princípio de que os tópicos a serem extraídos são conhecidos antes da análise. Uma interessante analogia é apresentada por [11]: se ao realizar a leitura de um jornal o nome das seções é conhecido, ao realizar a leitura aleatória das matérias contidas no jornal seria possível encontrar a probabilidade de a matéria fazer parte de cada seção do jornal. Se o modelo estiver bem ajustado, os documentos vão se encaixar naturalmente nos tópicos corretos.

O exemplo 3.4, extraído do tutorial [73], apresenta a criação de um corpus no modelo LDA utilizando a ferramenta Gensim [74], desenvolvida em Python para auxiliar no processamento de corpus extremamente grandes. No caso o conjunto de documentos é construído e tratado para remover do corpus final as palavras que não farão parte da análise.

Exemplo 3.4: Criando coleção de documentos [73]

```
>>> from gensim import corpora, models, similarities
>>>
>>> documents = ["Human machine interface for lab abc computer applications",
>>>              "A survey of user opinion of computer system response time",
>>>              "The EPS user interface management system",
>>>              "System and human system engineering testing of EPS",
>>>              "Relation of user perceived response time to error measurement",
>>>              "The generation of random binary unordered trees",
>>>              "The intersection graph of paths in trees",
>>>              "Graph minors IV Widths of trees and well quasi ordering",
>>>              "Graph minors A survey"]
>>>
>>> # remove common words and tokenize
>>> stoplist = set('for a of the and to in'.split())
>>> texts = [[word for word in document.lower().split() if word not in stoplist]
>>>           for document in documents]
>>>
>>> # remove words that appear only once
>>> from collections import defaultdict
```

```
>>> frequency = defaultdict(int)
>>> for text in texts:
...     for token in text:
...         frequency[token] += 1
...
>>> texts = [[token for token in text if frequency[token] > 1]
...           for text in texts]
...
>>> from pprint import pprint   # pretty-printer
>>> pprint(texts)
[['human', 'interface', 'computer'],
 ['survey', 'user', 'computer', 'system', 'response', 'time'],
 ['eps', 'user', 'interface', 'system'],
 ['system', 'human', 'system', 'eps'],
 ['user', 'response', 'time'],
 ['trees'],
 ['graph', 'trees'],
 ['graph', 'minors', 'trees'],
 ['graph', 'minors', 'survey']]
```

O exemplo 3.5 apresenta a aplicação do LDA ao corpus anterior. O resultado é uma lista de tópicos, onde em cada token identificado no tópico está uma probabilidade associada. A tabela 3.1 apresenta os tokens dos tópicos organizados por ordem de probabilidade, onde é possível observar a formação de dois grupos: um contendo as palavras `system`, `user` e `response` como os mais prováveis, e outro contendo as palavras `trees`, `graph` e `minor`. Analisar o significado de cada tópico, assim como definir o número que deve ser utilizado na geração do modelo, é o trabalho fundamental do pesquisador.

Exemplo 3.5: Criando modelo LDA [73]

```
>>> dictionary = corpora.Dictionary(texts)
>>> bow_corpus = [dictionary.doc2bow(text) for text in texts]
>>> from gensim.models import ldamodel
>>> model = ldamodel.LdaModel(bow_corpus, id2word=dictionary, num_topics=2)
>>> topics_lista = model.show_topics(num_topics=2, formatted=False)
>>> pprint(topics_lista)
[[(0.14403321551367465, u'system'),
  (0.11884705753525804, u'user'),
  (0.091546822221701629, u'response'),
  (0.091209377369637412, u'time'),
  (0.086561605619184148, u'computer'),
  (0.085153871837270745, u'survey'),
  (0.078560530602166334, u'human'),
  (0.072264984999513879, u'interface'),
  (0.071671453965504867, u'eps'),
  (0.058851704776940322, u'graph')],
 [(0.15897834761693569, u'trees'),
  (0.1484426874182575, u'graph'),
  (0.1030192455897791, u'minors'),
  (0.092925362591344721, u'system'),
  (0.075141469371409655, u'eps'),
  (0.074361297738823731, u'interface'),
  (0.069581314335760619, u'user'),
  (0.066086067280080971, u'human'),
```

```
25    (0.057419396961525447, u'survey'),
      (0.055568989990446904, u'computer')]]
```

Tópico #1	Tópico #2
system	trees
user	graph
response	minors
time	system
computer	eps
survey	interface
human	user
interface	human
eps	survey
graph	computer

Tabela 3.1: Modelo LDA produzido para dois tópicos

O trabalho de Wang [72] introduz a ideia de utilizar LDA em dados de redes sociais para encontrar a probabilidade de um evento relatado pelo usuário estar relacionado a alguma atividade criminal. O primeiro passo da técnica é organizar os eventos encontrados nos *tweets* em uma coleção de documentos associados ao dia d. Para encontrar os eventos, todos os *tweets* passam antes por um processo de SRL, que também tenta eliminar todas as palavras que não possuem relevância para os tópicos, como conectivos e *stopwords*. Assim, uma coleção de documentos doc_d será formada pelo conjunto de eventos e_i associados ao dia d, formando a equação 3.2 que representa todos os eventos identificados no dia d.

$$doc_d = \{e_1, e_2, ..., e_{n_d}\} \tag{3.2}$$

A próxima etapa é definir o conjunto de tópicos $\{t_1, t_2, ..., t_k\}$ que se deseja investigar, onde k é o número total de tópicos. Os documentos extraídos na equação 3.2 passarão pelo processo da LDA para calcular a probabilidade de estarem relacionados aos tópicos $\{t_1, t_2, ..., t_k\}$. O objetivo é tentar encontrar a probabilidade do tópico t estar presente no dia d, para a qual obtemos o modelo da equação 3.3. Assim, $T_{d,i}$ é a probabilidade do documento d estar relacionado ao tópico i.

$$P(T_d) = \{T_{d,1}, T_{d,2}, ..., T_{d,k}\} \tag{3.3}$$

3.3.2 Semantic Uplift

A importância da Web Semântica e dos conceitos de Linked Data já foram introduzidos nas seções 2.1 e 2.2. Contudo, qualquer sistema que deseje entrar no cenário da Web

de dados apresentado na LOD Cloud deve primeiro passar pelo trabalho de disponibilização das informações. Várias técnicas foram desenvolvidas desde então para facilitar a conversão dos dados no formato da Web Semântica, mas via de regra é necessário o auxílio de um programador ou administrador de dados para consolidar a conversão. No processo de comunicação entre o especialista da informação e o desenvolvedor responsável pela implementação, algum ruído pode eventualmente acabar acontecendo. Aproximar o especialista da informação do processo de disponibilização dos dados pode ser uma forma de aumentar a precisão.

A técnica de *Semantic Uplift* pode ser definida como o "processo de converter dados não-RDF (...) numa representação do conhecimento baseada em RDF" [14]. Trata-se de um trabalho específico de conversão de um conjunto de dados não estruturados relacionados à violência política no formato RDF para a Web Semântica. Os dados são utilizados como entrada para um software chamado DaCura apresentado pelo autor, que deve ajudar usuários sem conhecimento prévio de Web Semântica a criar visualizações sobre seus próprios dados.

Ainda que a ferramenta seja somente um ponto de partida e um vocabulário não seja suficiente para criar uma ontologia reutilizável, mais importante do que definir uma ontologia formal é encontrar utilidade para os dados [41]:

> Mais importante do que focar nos desafios relativos à criação de ontologias grandes e expressivas por especialistas em conhecimento, os grandes mecanismos sociais que buscamos necessitam da descoberta de uma maneira de minimizar ao máximo a tarefa de tornar conhecimento humano desorganizado num espaço de informação compartilhado que seja útil para todos.

A utilização da técnica de *Semantic Uplift* é uma tentativa de aplicar a simplicidade e aproximar o dado do cientista social, aquele que efetivamente vai analisar as informações obtidas.

3.4 Qualidade da Análise

Ao final de qualquer análise envolvendo dados estatísticos é necessário quantificar a qualidade dos resultado obtidos. Perguntas do tipo "os nossos resultados são melhores do que os anteriores?" [59] devem ser respondidas de maneira clara e objetiva. Uma revisão sistemática da área de medidas de performance para tarefas de classificação [64] reuniu vinte e quatro tipos de medidas, que dependem dos tipos de dados que estão sendo classificados.

Para a aplicação de SRL ao texto existem duas possibilidades de validação: ou a estrutura semântica foi identificada da forma correta ou não. A descrição dos resultados

obtidos pelo módulo nlpnet [35] apresenta a precisão em termos do método *F1 Score* para cada um dos classificadores do modelo Propbank.

A tabela 3.2 ajuda a entender a construção dos métodos de avaliação, apresentando uma matriz de classificação para resultados de análise binária. No caso duas classes de dados são apresentadas: *pos*, para o caso em que o resultado foi classificado como positivo, e *neg*, para o caso em que o resultado foi classificado como negativo. Para ambas as classes existem duas possibilidades: ou o dado foi encaixado na classe correta (verdadeiro) ou foi encaixado na classe errada (falso). É esperado que um bom algoritmo de classificação seja capaz de classificar a maior parte dos dados na classe correta.

Classe de dados	**Classificado como** *pos*	**Classificado como** *neg*
pos	positivo verdadeiro (tp)	falso negativo (fn)
neg	falso positivo (fp)	negativo verdadeiro (tn)

Tabela 3.2: Matriz de classificação binária [64]

No caso da SRL um resultado é classificado como *pos* se o argumento foi classificado na estrutura, enquanto o *neg* representa a ausência de compatibilidade com a estrutura semântica. Para todos os resultados identificados, verificamos quais estão corretamente identificados na estrutura para formar o conjunto tp; os que estão erroneamente classificados formarão o conjunto fp. Da mesma forma, os que não foram classificados são analisados, e caso tenham sido erroneamente descartados formarão o conjunto fn; caso contrário completarão o conjunto tn.

Os classificadores são assim definidos em termos de *Precision* e *Recall* [64]:

Precision Medida do total de dados classificados corretamente relativa ao total de dados classificados como positivos (exemplo 3.4).

$$\frac{tp}{tp + fp} \tag{3.4}$$

Recall Medida do total de dados classificados corretamente relativa ao total de dados efetivamente positivos (exemplo 3.5).

$$\frac{tp}{tp + fn} \tag{3.5}$$

O exemplo 3.6 apresenta a definição *F1 Score* como a média harmônica dos termos de *Precision* e *Recall* [9][p.267].

$$F = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{3.6}$$

Pela equação é possível observar que quando os termos são perfeitos o resultado tende a 1.0, da mesma forma que tende a 0 quando os dados são muito ruins. Um bom trabalho de análise deve envolver a identificação dos valores ideais, já que "normalmente há um *trade-off* se o desejo é aumentar *precision* ou *recall*, pois é muito difícil obter ambos". O *F1 Score* é uma medida consagrada na análise dos resultados obtidos em técnicas de NLP.

3.5 Dados criminais

Ao propor um *framework* para a mineração de dados criminais, [19] apresenta a técnica de extração de entidades:

> A extração de entidades identifica padrões particulares em dados como texto, imagens ou arquivos de áudio. A técnica tem sido utilizada para identificar automaticamente pessoas, endereços, veículos e características específicas dos relatórios policiais.

O autor apresenta uma análise comparativa entre as técnicas de mineração de dados criminais e a capacidade de identificação para classificação do crime. De acordo com o artigo, a extração de entidades é a única capaz de identificar todos os tipos de crime com acurácia aceitável.

Definir atividade criminal é o primeiro passo para entender e analisar dados relativos à violência. "Um crime pode englobar uma ampla gama de atividades, desde infrações civis como estacionamento em local proibido, até assassinatos em massa de organização internacional, como os ataques de 11/09." [19] As ocorrências policiais no Brasil trabalham com o conceito de **incidente**, representando qualquer tipo de atividade criminal [48][p.148].

Nos EUA o FBI mantém o *National Incident-Based Reporting System* [31] – NIBRS – que tem como objetivo organizar os dados criminais baseados em incidentes em duas categorias: Grupo A, para o qual uma quantidade significativa de dados é coletada, e Grupo B, para o qual somente os dados relativos a prisões são reportados. O sistema é parte do projeto *Uniform Crime Record* – UCR [6] – responsável por gerar estatísticas uniformes para o país. Os dados do NIBRS vêm de mais de 5.000 agências representando 20% da população e 16% de todas as estatísticas criminais no país.

No Brasil, construir uma base uniforme de dados criminais ainda é um grande desafio. O desenho de um Sistema Unificado de Segurança Pública [48][p.141] – SUSP – supõe a utilização de um sistema de coleta de dados em três níveis. O fluxo de dados no sistema está representado na Figura 3.2. Os autores defendem que a informação deve ser coleta o mais próximo possível da fonte, começando nas prefeituras.

[6] Mais informações podem ser obtidas na página do projeto: `http://www2.fbi.gov/ucr/ucr.htm`

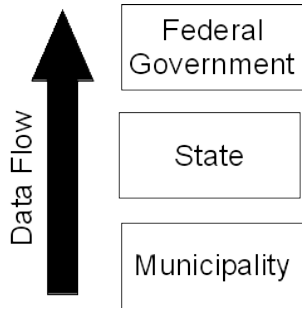


Figura 3.2: Fluxo de dados criminais

A lei brasileira também define diferentes tipos de polícia para diferentes atividades: a polícia militar é responsável pela prevenção, enquanto a polícia civil trabalha na investigação. Ambos utilizam diferentes sistemas para registro de ocorrências criminais, que mantêm diferentes bancos de dados. O processo de registro de incidentes está apresentado na Figura 3.3. Do momento em que o incidente acontece até o registro da ocorrência, o primeiro passo é comparecer à polícia civil ou militar. Ainda que o incidente esteja registrado na polícia militar, é necessário passar a informação à polícia civil, onde pode seguir três caminhos diferentes:

1. "Encerrado no balcão" ou mediação de conflitos civis. O policial tenta resolver o problema entre as partes envolvidas sem iniciar um registro de ocorrência;

2. Encaminhamento para outros órgãos, no caso em que o incidente não é classificado como crime e não inicia um inquérito policial;

3. Preenchimento do boletim de ocorrência. No caso o delegado responsável pode submeter o incidente à uma rotina administrativa ou iniciar um inquérito policial.

A complexidade do cenário supõe que é difícil registrar um boletim de ocorrência. De fato, refletindo sobre um sistema de segurança unificado [13], os autores enfatizam o problema:

> Grande parte dos eventos, acidentes, incidentes, desordens, incivilidades, conflitos e violências a que está submetida a população tem como resposta soluções civis não policiais. Este fenômeno é designado comumente pelo termo sub-registro e resulta da decisão da população de não registrar nos órgãos de segurança pública os eventos a que tenham sido vítimas.

Encontrar uma maneira de identificar as soluções civis não-policiais pode ajudar governos e população a ter um melhor entendimento sobre violência e criminalidade.

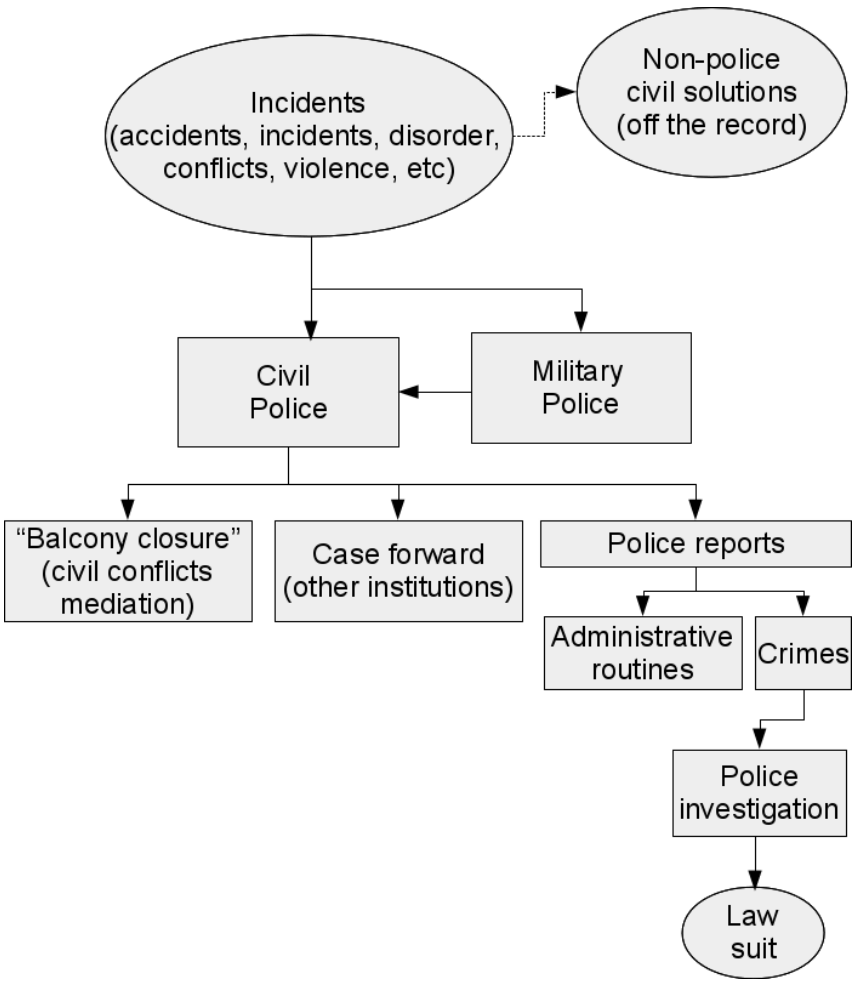


Figura 3.3: Registro de ocorrências policiais no Brasil [13]

Capítulo 4

Modelo arquitetural

O capítulo apresenta uma proposta para responder à pergunta de pesquisa mencionada na seção 1.1: se for possível definir um modelo arquitetural para *social machines*, ele pode ser utilizado na construção de um protótipo para identificar atividade criminal? É importante ressaltar que o trabalho é parte do processo de desenvolvimento do arcabouço prático e teórico para a definição de *Social Machines*. A ideia apresentada nas próximas seções se baseia na utilização de dados oriundos de redes sociais como uma resposta não-policial à criminalidade. O vocabulário de informações criminais pode ser organizado em termos de significado e interpretado dentro de fronteiras sociais, ajudando o observador a identificar atividade criminal através das redes.

4.1 Solução teórica

Trabalhos recentes [30] abordam os problemas relativos à implementação de uma Social Machine com foco em violência e criminalidade:

> É possível definir um sistema capaz de criar conhecimento sobre crime suficiente para substituir os dados criminais abertos ou os dados da pesquisa de vitimização?

Realizada tanto no Reino Unido [30] quanto no Brasil [33] e nos Estados Unidos [31], a Pesquisa sobre Vitimização (*Crime Victims Survey*) tem como objetivo retratar a sensação de segurança utilizando a metodologia de pesquisa populacional para descobrir quantos cidadãos já foram vítimas de assalto. É uma tentativa de coletar dados estatísticos que não estão diretamente relacionados ao registro formal de ocorrências.

No escopo de uma dissertação de Mestrado é difícil construir uma resposta embasada que permita a substituição dos dados coletados em pesquisas populacionais. Assim, a solução teórica possui dois lados: uma proposta de implementação com foco mais no cidadão do que nos governos e a documentação da arquitetura utilizada no processo de

desenvolvimento. A primeira trata da entrega de uma solução para a população, enquanto a segunda situa o processo de desenvolvimento no arcabouço teórico existente.

Sem um suporte total de governos ao redor do mundo, uma aplicação do tipo não é capaz de substituir as ferramentas oficiais. Ao invés de definir mecanismos de auto-regulação para garantir que a informação seja confiável, está a cargo dos usuários realizarem *crowdsourcing* dos dados como lhes for mais conveniente. Ao invés de criar um ambiente formal para lidar com todas as possíveis restrições sociais, a proposta é construir um modelo baseado em pares (*peer to peer*) para auxiliar os usuários a fornecer respostas não-policiais [13].

A construção da solução teórica obedece os princípios da *Evidence Based Software Engineering* [44] – EBSE – citados em [45] e revistos em [43]. As diretrizes selecionadas estão apresentadas na lista abaixo:

D1 Identificar a população de onde os dados individuais serão coletados;

D2 Definir o processo de seleção dos dados;

D3 Definir o processo de tratamento dos dados;
(...)

D5 Definir a unidade experimental;

D6 Para experimentos formais, realizar um pré-experimento ou pré-cálculo para identificar ou estimar o tamanho mínimo da amostra.

A importância da seleção da população descrita nas diretivas *D1* e *D2* também foi citada por [72], ao considerar que os melhores resultados seriam obtidos utilizando os *tweets* enviados por uma pequena agência de notícias de uma cidade nos Estados Unidos. Seu trabalho recebia como entrada o conjunto de todos os *tweets* enviados pela agência, e como o corpus que eles possuíam tinha como origem dados reais relacionados a acidentes de trânsito, era necessário que os dados processados estivessem no mesmo contexto para obter maior acurácia. O processo completo de classificação envolvia uma combinação de SRL + LDA , descrita na seção 3.3.1.

Sabendo da importância da definição da população, uma extração preliminar de dados foi realizada no dia 09 de Março de 2014 para tentar verificar a viabilidade de utilização de dados obtidos através da API do fluxo de atividades público do Twitter. A análise seguiu um procedimento simples: execução de busca através da API utilizando palavras intuitivamente relacionadas a atividades criminais. A busca foi realizada coletando todos os *tweets* que continham as palavras `arma`, `roubo` e `ladrão` por um intervalo de 100 segundos. Os resultados obtidos estão apresentados na Tabela 4.1. Vale observar que, por tratar-se de uma análise manual dos resultados obtidos, não é possível obter informações sobre os resultados negativos. Os valores de *precision* e *recall* dos resultados trazidos pelo Twitter não são informados. A classificação foi realizada utilizando o seguinte critério:

caso o incidente identificado através do *tweet* estivesse realmente relacionado a violência, considera-se um positivo verdadeiro. Caso contrário, trata-se de um falso positivo.

Termo de busca	**Número de tweets**	**Falsos positivos**	**Precision**
Weapon	325	64	0,84
Thief, Steal	80	53	0,60
Total de tweets obtidos: **425**			

Tabela 4.1: Dados extraídos do Twitter em 09 de Março de 2014

A análise preliminar aponta a importância de selecionar termos apropriados para a busca. A utilização de termos relacionados à atividade criminal por parte dos usuários acontece em contextos diversos, como demonstrado no seguinte exemplo:

> Perdemos por causa do juiz ladrão

Trata-se de um falso positivo, onde os torcedores demonstram a paixão por seus times atacando o árbitro de futebol. Não é possível afirmar que houve um crime de fato pela simples análise da sentença. É possível concluir então que realizar uma busca pela palavra `roubo` durante um jogo de futebol pode trazer como resultado uma série de falsos positivos.

É preciso então validar a hipótese de que a busca localizada em contas específicas pode trazer melhores resultados. No Brasil, existe uma tradição jornalística onde agências de notícias locais possuem um foco maior em notícias relacionadas a violência, enquanto as grandes redes promovem uma agenda nacional. O exemplo da Tabela 4.2 foi extraído da conta de Twitter do programa DFAlerta [1], um programa da TV Brasília, emissora local de Brasília. O programa é pautado pelo acompanhamento das notícias sobre criminalidade e violência na região, utilizando uma carga pesada de humor em contato direto com as ocorrências policiais. A análise dos dados da tabela sugere que a abordagem de utilização de contas específicas pode trazer melhores resultados.

Termo de busca	**Número de tweets**	**Falsos positivos**	**Precision**
Violência	43	1	0,98
Total de tweets analisados: **43**			

Tabela 4.2: Dados extraídos da conta do usuário DFAlerta no Twitter

Contudo, o objetivo principal de uma *social machine* é atender as demandas sociais, e não somente apresentar uma seleção da pauta publicada por agências de notícias. Ela deve fornecer mais capacidade de análise à população, e não depender dos interesses de terceiros na seleção do conjunto de dados. Assim, é necessário criar uma arquitetura

[1] Disponível em `http://twitter.com/DFAlerta`

híbrida, deixando claro aos usuários onde os dados podem ser confiáveis, mas deixando a cargo de quem vai utilizar a informação a análise da qualidade da fonte. A confiabilidade de uma informação oriunda de redes sociais está diretamente relacionada à identificação de quem foi o responsável pelo conteúdo, assim a identificação da fonte original permite uma avaliação mais fácil em termos de qualidade. A lógica é de que fontes mais confiáveis possuem mais qualidade.

A diretriz *D3* fala sobre a necessidade de definir um processo de tratamento dos dados que seja capaz de apresentar bons resultados. A técnica de SRL descrita na seção 3.2.1 identifica eventos dos *tweets*, e a utilização do módulo nlpnet permite a aplicação da técnica em Língua Portuguesa. Após a identificação, os eventos são organizados em tópicos através de LDA como descrito na seção 3.3.1. A solução proposta é uma adaptação do trabalho de [71] que utiliza SRL + LDA para classificar os *tweets* de acordo com a categoria a que estão relacionados. Categorizar, identificar o contexto e permitir a avaliação da informação por parte do usuário devem ajudar a aumentar a eficácia da técnica.

A última observação sobre as diretrizes vem da definição da unidade experimental (*D5*). Os dados oficiais sobre violência e criminalidade, como apresentado na seção 3.5, trabalham com o conceito de **incidente**. Cada *tweet* identificado corretamente na estrutura de tópicos representa uma unidade na classificação e identificação de atividade criminal. A Tabela 4.3 apresenta um resumo da aplicação das diretrizes à pesquisa.

Diretriz	**Teoria**	**Implementação**
D1	População	*Tweets* extraídos através da API
D2	Processo de seleção	Seleção de termos relacionados à violência e criminalidade
D3	Estratégia de tratamento	SRL + LDA
D5	Unidade experimental	Incidentes relacionados à criminalidade e violência
D6	Pré-cálculo	Dados das Tabelas 4.1 e 4.2

Tabela 4.3: Diretrizes adotadas na pesquisa [45]

4.2 Procedimento de implementação

A seção 2.4 apresenta um conjunto de trabalhos relacionados que foram utilizados como fundamento para construção da proposta de pesquisa. Todavia, apesar de apresentar definições importantes e apontar caminhos, os autores não apresentam detalhes específicos referentes à implementação ou como os conceitos poderiam ser estendidos a outras apli-

cações. Esta seção apresenta uma proposta de implementação embasada na introdução de padrões de projeto para preencher esta lacuna.

Um sistema que supõe a coleta de tendências sociais precisa ter a capacidade de recolher informações sobre determinada sociedade. Ceilândia Muita Treta [2] é uma página do facebook com foco em uma cidade satélite [3] de Brasília, mantida por um grupo de moradores da região de Ceilândia. Trata-se de uma das maiores e mais violentas cidades nos arredores da capital, e os mantenedores da página a utilizam como ferramenta para falar de seu orgulho da cidade. Ela é conhecida pelo humor e por ser uma boa fonte de informações para a imprensa: como os donos da página são jovens com bastante tempo livre, normalmente a informação sobre o que acontece na região chega mais rápido à página do facebook.

No dia 6 de Junho de 2014 eles postaram a mensagem da Figura 4.1, referente a três homicídios ocorridos em Ceilândia. Ainda que eles tenham, em sua visão, relatado um crime ocorrido na região, uma série de detalhes não foram fornecidos, como o nome da vítima, causa da morte, etc. Com as informações publicadas na página não é possível dizer, com certeza, se as mortes foram registradas oficialmente seguindo procedimento descrito na seção 3.5.

Figura 4.1: Post no facebook da página *Ceilândia Muita Treta*. Endereço original: `https://www.facebook.com/ceilandiamuitatretta/posts/753732047981961`

Para descrever a proposta de implementação do protótipo de *social machine* desenvolvido nesta pesquisa, é possível supor o caso em que uma busca no facebook utilizando o termo `homicídio` traga a mensagem da página como um resultado. Três informações importantes estão presentes de forma implícita no texto:

Evento Homicídio

Tempo Hoje

[2] Endereço da página: `http://www.facebook.com/ceilandiamuitatretta`
[3] Cidade satélite é o nome dado às cidades do Distrito Federal

Local na Ceilândia

O processamento utilizando SRL pode trazer como resultado a identificação dos papéis semânticos do texto, fornecendo as informações de evento, tempo e local descritas acima. A identificação possibilita diferentes facetas da análise, como agrupar eventos no mesmo tema (homicídio) e que ocorram na mesma faixa de tempo (hoje). Tratam-se de novos metadados, que apesar de não serem diretamente fornecidos no momento da busca, são encontrados após o processamento e permitem análises posteriores.

Uma observação mais aprofundada pode ser feita através do papel semântico identificado como local. Considerando a disponibilidade de metadados fornecidos pelo facebook, é possível supor o caso onde a informação do local é fornecida de maneira precisa como um metadado. Contudo, ainda que não sejam fornecidas as coordenadas geográficas relativas ao evento, algumas bases de dados públicas permitem a associação entre termos de busca e suas localidades geográficas. É o caso da *API de Geocoding do Google Maps*: fornecendo algum termo de busca como parâmetro, o serviço fornece as coordenadas geográficas das possíveis localidades, em ordem de probabilidade. O exemplo da Figura 4.2 trata da utilização da API do Google buscando a cidade de `Ceilândia` identificada como localidade através da aplicação de SRL ao texto extraído do facebook. As coordenadas geográficas podem então ser armazenadas como um novo metadado, juntamente com o resultado do processamento da SRL.


```
  formatted_address: "Ceilândia, Brasília - DF, Brasil",
▾ geometry: {
  ▾ location: {
        lat: -15.813415,
        lng: -48.1044183
    },
    location_type: "APPROXIMATE",
  ▾ viewport: {
    ▾ northeast: {
        lat: -15.802679,
        lng: -48.0884109
      },
    ▾ southwest: {
        lat: -15.8241504,
        lng: -48.1204257
      }
    }
}.
```


Figura 4.2: Exemplo de resposta da *API de Geocoding do Google Maps* [39] buscando pelo termo `Ceilândia`

Considerando que a busca foi realizada com um tema específico (homicídio), é possível imaginar que o resultado esteja relacionado ao tópico encontrado. Mas será que todas as buscas pela palavra homicídio vão trazer resultados que tratam realmente de homicídio? Na seção 3.3.1 foi introduzida a técnica LDA para modelagem de tópicos, permitindo

descobrir o tema do documento ou conjunto de documentos, desde que se saiba o número de tópicos a ser encontrado. Para realizar a descoberta automática é preciso ter então uma base de treinamento, com acurácia medida, que seja capaz de encontrar, em um conjunto desconhecido de documentos, a probabilidade para cada um deles estar relacionado ao tópico. A probabilidade também constitui um novo metadado para o documento, juntamente com o nome do tópico mais provável.

Como o processamento dos módulos é feito em diferentes etapas, o modelo de armazenamento a ser utilizado deve ser capaz de trabalhar com dados cuja estrutura seja mutável. No caso de dados oriundos de redes sociais, o Twitter sugere a utilização de formatos JSON nativos no motor de armazenamento, de forma a garantir que a provável evolução do conjunto de metadados não afete a integridade de sistemas que trabalhem com seus dados. O processamento e armazenamento representam por si só uma importante etapa do desenvolvimento.

A última questão em aberto na proposta de arquitetura diz respeito às diferentes formas de consumir os dados, uma vez processados e armazenados. Quais são as possíveis interfaces? Qual o modelo de disponibilização? As perguntas apresentadas são exemplos da etapa de planejamento do acesso aos dados, que não só influencia o processo de desenvolvimento, como também pode nortear os princípios arquiteturais. Um sistema que tem como objetivo fornecer uma interface pública para consumir os dados deve ser desenvolvido de maneira diferente de um que apenas processa e apresenta os resultados como saída para a visualização.

A *social machine* proposta no trabalho possui um papel importante neste ponto: se os usuários possuem telefones celulares com o GPS disponível, ou se forneceram informações sobre seus locais de interesse, as notícias podem chegar a eles rapidamente. Algum tipo de sistema de notificação para celulares utilizando *push* pode notificá-los sobre os homicídios na região, assim como o Google Maps é capaz de marcar os locais onde acontecerem incidentes identificados. Se estão caminhando próximos à cena do crime e sabem sobre o que foi enviado no facebook, eles podem fornecer algum tipo de *feedback*, informando que os homicídios aconteceram de verdade, por exemplo. Trata-se do poder do *crowdsourcing* para fornecer informações mais confiáveis sobre os crimes, representando os alicerces para um banco de dados construídos pelos usuários (*crowdsourced database*).

A aplicação das diretrizes da EBSE descritas na seção 4.1 aos modelos de processamento, armazenamento e visualização descritos ao longo da seção permitem desenhar o seguinte processo de desenvolvimento incremental baseado em etapas:

1. Determinar a população ou fonte de dados;

2. Escolher o objeto da pesquisa, ou seja, qual o domínio da informação a ser tratada;

3. Selecionar as técnicas de processamento e armazenamento das informações para o domínio;

4. Planejar a interface de visualização/apresentação das informações coletadas.

4.3 Arquitetura do sistema

Existe um longo caminho entre a identificação de uma atividade nas redes sociais, sua identificação como relacionada a um crime, até um banco de dados construído pelos usuários. Assim, a arquitetura de implementação será dividida em camadas, como ilustrado na Figura 4.3. Os procedimentos de integração e comunicação entre as camadas serão discutidos nas próximas seções.

4.3.1 L1: Camada de extração

Como mencionado ao longo do capítulo 3, a atividade das redes sociais é baseada em linguagem natural, normalmente no idioma nativo dos usuários. Para analisar o conteúdo, é preciso primeiro extrair e processar os dados. A escolha da fonte deve levar em conta temas como acesso, disponibilidade e restrição ao uso dos dados. Apesar de se tratar de uma fonte popular do ponto de vista dos usuários, o facebook, a partir da versão 2.0 de sua API, fechou as buscas públicas [4], significando que não é mais possível extrair dados para tratamento. O fato, que ocorreu durante a pesquisa, praticamente inviabilizou a utilização dos dados oriundos do facebook.

Assim, a camada L1 descreve o procedimento de extração de dados utilizando a API do Twitter, bastante popular em pesquisas acadêmicas: [3], [72] e [37] são bons exemplos. A API fornece duas diferentes formas de buscar status, conforme descrito na seção 3.3:

- API de busca;
- API do fluxo de atividades público.

A implementação da busca utiliza o módulo Python Twitter [65], que implementa as principais chamadas da API pública do Twitter na linguagem Python. O objetivo é executar chamadas ao procedimento de busca fornecido pelo módulo para identificar, numa sentença escrita em Português, o elemento que esteja assumindo o papel de *evento*. A identificação é feita através da técnica de SRL descrita na seção 3.2.1, "uma tarefa de processamento semântico superficial na qual, para cada predicado em uma sentença, o objetivo é identificar todos os constituintes e seus papéis semânticos" [57].

[4] A alteração pode ser vista no *changelog* da versão 2.0 disponível no seguinte endereço: `https://developers.facebook.com/docs/apps/changelog`

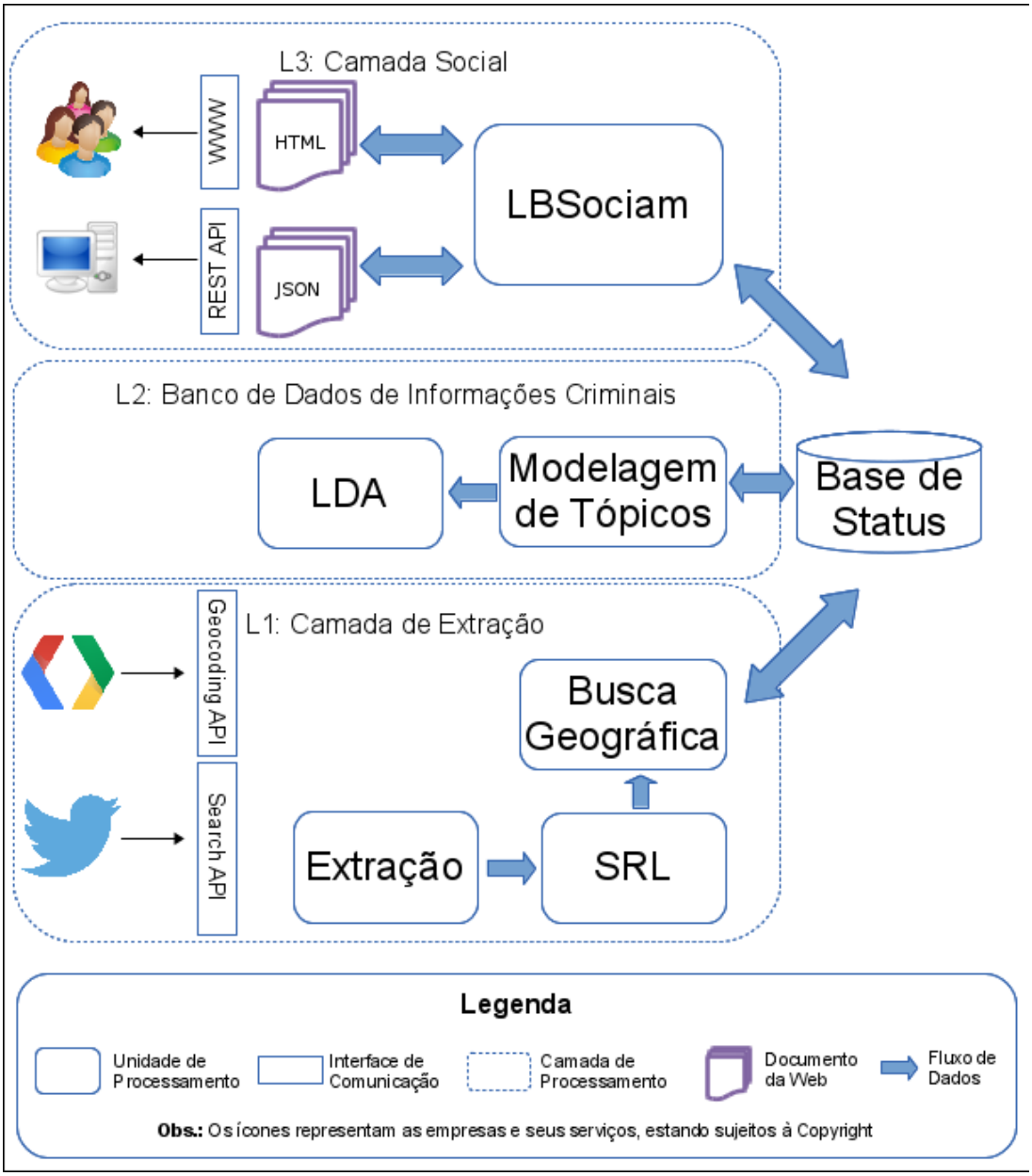


Figura 4.3: Definição da Social Machine. Adaptada da definição de arquitetura do Projeto Euclid [29]

Um exemplo de utilização pode ser visto na aplicação da técnica ao status da Figura 4.4, extraído do Twitter no dia 07 de fevereiro de 2015. O texto original trata de uma apreensão de drogas realizada pela polícia, relatada na rede social e extraída através da API. A aplicação de SRL ao status utilizando o módulo nlpnet citado na seção 3.2.1 está apresentada no exemplo 4.1.

O módulo foi capaz de encontrar dois predicados: `apreende` e `Tira`. Concentrando a análise no predicado `apreende`, é possível observar a ocorrência do evento $A2$, `drogas`,

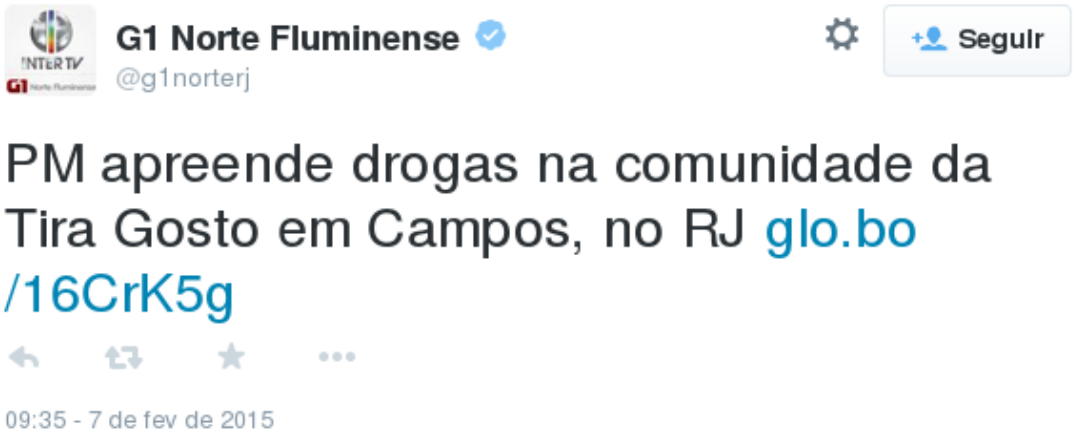


Figura 4.4: Status extraído do twitter no dia 07/02/2015. Link: `https://twitter.com/NOTMCCALL/status/563928175278030848`

indicando tratar-se assim de um evento de apreensão de drogas. Também foi identificado o token $AM-LOC$ que, na sintaxe do Propbank [42] significa lugar. Os tokens `em campos no RJ` indicam o local de incidência do predicado `apreende`. A informação completa diz respeito a uma apreensão de drogas na comunidade de Tira Gosto no Rio de Janeiro, o que já parecia óbvio ao ler a frase. Contudo, ao identificar os papéis semânticos dos constituintes da sentença, a mesma informação está disponível para o processamento computacional, sem a intervenção humana para realizar a classificação.

Exemplo 4.1: Exemplo de aplicação de SRL a um status do Twitter

```
>>> import nlpnet
>>> tagger = nlpnet.SRLTagger()
>>> exemplo = u"PM apreende drogas na comunidade da Tira Gosto em Campos, no RJ"
>>> srl_tag = tagger.tag(exemplo)
>>> pprint(srl_tag[0].arg_structures)
[(u'apreende',
  {u'A0': [u'PM'],
   u'A1': [u'drogas'],
   u'A2': [u'na', u'comunidade', u'da', u'Tira', u'Gosto'],
   u'AM-LOC': [u'em', u'Campos', u',', u'no', u'RJ'],
   u'V': [u'apreende']}),
 (u'Tira',
  {u'A0': [u'PM'],
   u'A1': [u'Gosto'],
   u'AM-LOC': [u'no', u'RJ'],
   u'V': [u'Tira']})]
```

É importante notar ainda que o texto foi extraído utilizando o token *drogas*. De acordo com os argumentos apresentados na seção 3.3, a seleção dos termos utilizados como filtros na busca afeta diretamente a qualidade dos resultados. Para melhorar a qualidade da extração, a técnica de [72] sugere a utilização de SRL para identificação dos eventos. Ao

encontrar o token `drogas` como um evento após o processamento, a hipótese defendida pelos autores diz que as chances do *tweet* estar relacionado com o termo de busca aumenta consideravelmente.

O próximo passo para o processamento dos dados é a aplicação da técnica LDA. Sua importância pode ser melhor explicada levando em consideração a organização em temas [11]:

> Mais do que apenas encontrar documentos utilizando somente a busca por palavras-chave (*keywords*) é necessário encontrar primeiro o tema que estamos interessados e, a partir daí, encontrar os documentos relacionados a ele.

A técnica LDA apresenta os termos extraídos das redes sociais como um conjunto de palavras (*bag of words*). O objetivo é identificar as relações entre os eventos extraídos da SRL e a saída do primeiro processamento da LDA. "As palavras que não estão relacionadas a eventos são filtradas e o algoritmo de estimativas padrão é aplicado ao conjunto de documentos para estimar os parâmetros da LDA" [71]. Os eventos válidos são organizados em tópicos, e os dados que não fizerem parte dos tópicos identificados são descartados.

O banco de dados construído como resultado terá todas as informações originais do status extraídas da rede social, adicionado dos parâmetros obtidos através de SRL e LDA. Para facilitar a visualização e agrupamento das informações alguns metadados são armazenados em destaque, gerando como resultado o modelo do exemplo 4.2 apresentado no formato JSON. O campo `events_tokens` contém o conjunto de eventos extraído após a aplicação de SRL, e o conjunto de `events_tokens` de todos os documentos constitui o corpus para cálculo do modelo LDA.

Exemplo 4.2: Conjunto de metadados extraídos das redes sociais

```
{
    "tokens": [
        "Text"
    ],
    "hashtags": [
        "Text"
    ],
    "inclusion_datetime": "DateTime",
    "search_term": "Text",
    "text": "Text",
    "events_tokens": [
        "Text"
    ],
    "inclusion_date": "Date",
    "positives": "Integer",
    "arg_structures": [
        {
            "predicate": "Text",
            "argument": [
```

```
                {
                    "argument_name": "Text",
                    "argument_value": [
                        "Text"
                    ]
                }
            ]
        }
    ],
    "source": "Json",
    "origin": "Text",
    "selected_category": "Text",
    "negatives": "Integer",
    "location": {
        "latitude": "Decimal",
        "city": "Text",
        "id_location": "Integer",
        "loc_origin": "Text",
        "longitude": "Decimal"
    }

}
```

4.3.2 L2: Banco de dados de informações criminais

A camada L2 demonstra como os dados obtidos na camada de extração (L1) serão processados para adição de informações semânticas. O procedimento é uma extensão da técnica *Semantic Uplift* descrita seção 3.3.2, e as categorias identificadas serão utilizadas como entrada para o número de tópicos no modelo LDA. Serão considerados somente os eventos relacionados a violência e criminalidade.

Para que os resultados da pesquisa sejam verificados em diferentes contextos, como defendido na seção 1.2, é necessário encontrar uma forma de situar as atividades de redes sociais no contexto de violência e criminalidade. No Brasil a utilização de dados estatísticos para análise criminal é feita através do Anuário Brasileiro de Segurança Pública, apresentado na seção 3.5. Na edição 2014 do Anuário [33] os crimes são classificados em categorias, e as principais são apresentadas em tabelas. No que tange à classificação dos dados em categorias, também é importante considerar o dicionário de dados do Sistema Nacional de Estatísticas em Segurança Pública e Justiça Criminal [51]. Fazendo um cruzamento dos dados e diminuindo o nível de detalhamento em alguns casos, é possível obter a taxonomia para dados relativos à violência e criminalidade [5] apresentada na Figura 4.5.

[5] A taxonomia foi gerada com a ferramenta *bubbl.us*, disponível no endereço `http://bubbl.us/`

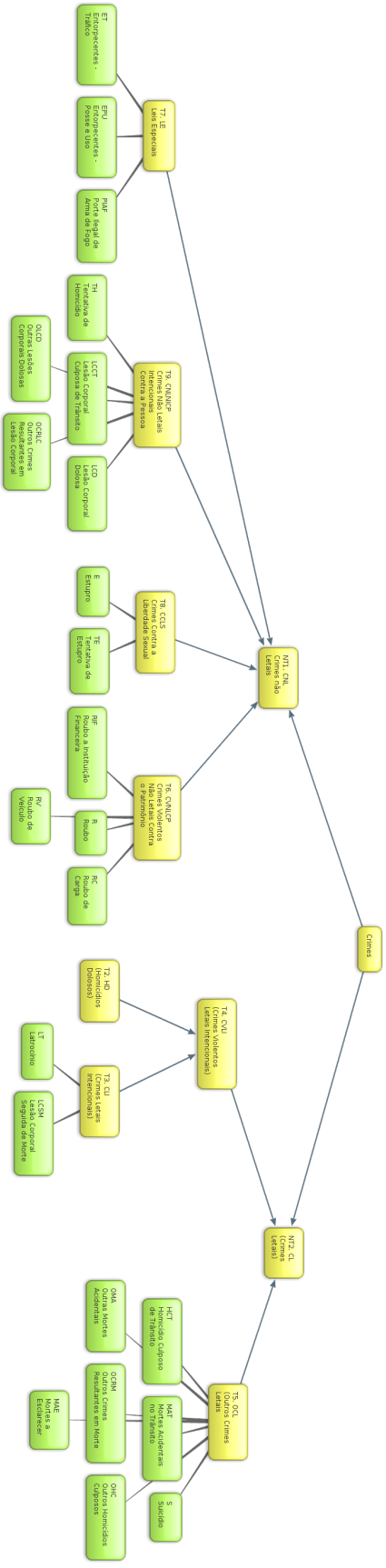


Figura 4.5: Taxonomia para eventos relacionados a violência e criminalidade

Os termos foram agrupados em siglas resumidas, e todos os que possuem o prefixo **T** possuem dados tabulados disponíveis no Anuário de Segurança Pública [33]. Os números são uma referência à identificação da tabela no anuário, assim T1 representa a Tabela 1. Por se tratar de um conjunto muito abrangente de informações, é difícil conseguir identificar detalhadamente cada um dos termos da taxonomia considerando como fonte de entrada dados oriundos de redes sociais. Foi necessário então construir uma versão simplificada que obedecesse a dois requisitos básicos:

1. Permitisse a identificação única, ou seja, fosse possível encontrar um conjunto reduzido de palavras relacionadas ao termo que poderiam alimentar a API de busca do Twitter;

2. Possuísse dados tabulados oficiais para efeito de comparação posterior.

Levando em conta os critérios definidos foi possível chegar à taxonomia reduzida da Figura 4.6.

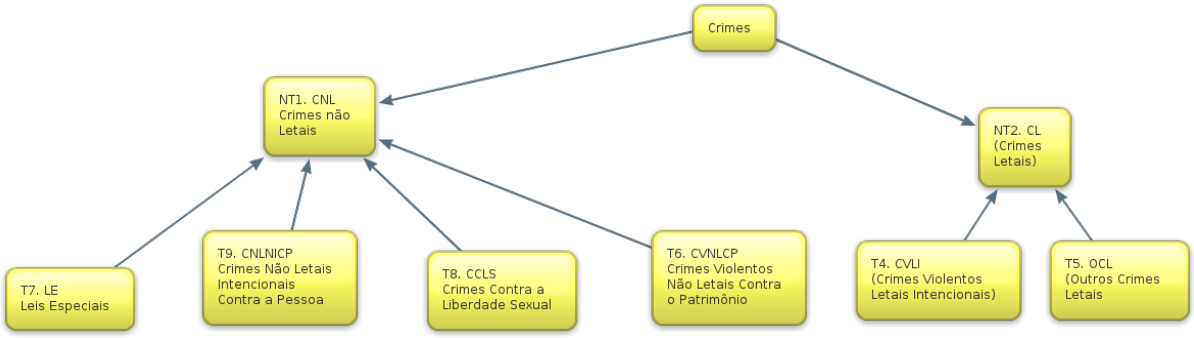


Figura 4.6: Taxonomia reduzida para eventos relacionados relacionados a violência e criminalidade

Os EUA também mantém uma base que tenta unificar os dados de segurança pública para todo o país [31], alimentada por mais de 3000 instituições. A base possui uma lista de 22 atividades criminais que são coletadas nas instituições participantes do sistema. Analisando os termos do Grupo A e comparando com a taxonomia reduzida do Brasil, é possível encontrar semelhanças na base de dados americana:

Homicide Offenses Intentional and violent death incidents, ou crimes violentos letais e intencionais (CVLI)

Larceny/Theft Offenses Pocket-Picking, Purse-Snatching, Shoplifting, Theft from Building, Theft from Coin-Operated Machine or Device, Theft from Motor Vehicle, Theft of Motor Vehicle Parts or Accessories, All Other Larceny

Robbery All violent theft offenses, such as kidnapping and robbering, armored car robbery, gunfire robbery, etc.

Drugs – traffic [6].

Gunfire possession Unauthorized gunfire possession

Sex offenses Forcible - Forcible Rape, Forcible Sodomy, Sexual Assault With An Object, Forcible Fondling

Assault Offenses Aggravated Assault, Simple Assault, Intimidation

Others Any other crime not in above categories

O banco de dados de informações semânticas construído no trabalho deve ser simples o suficiente para ser compreendido por usuários leigos e expressivo o suficiente para permitir a integração na Web de Dados. A Figura 4.7 mostra a representação do domínio, uma versão reduzida do modelo proposto por [14]. O objetivo é construir um serviço baseado em dados abertos para facilitar a integração e tornar disponíveis as informações.

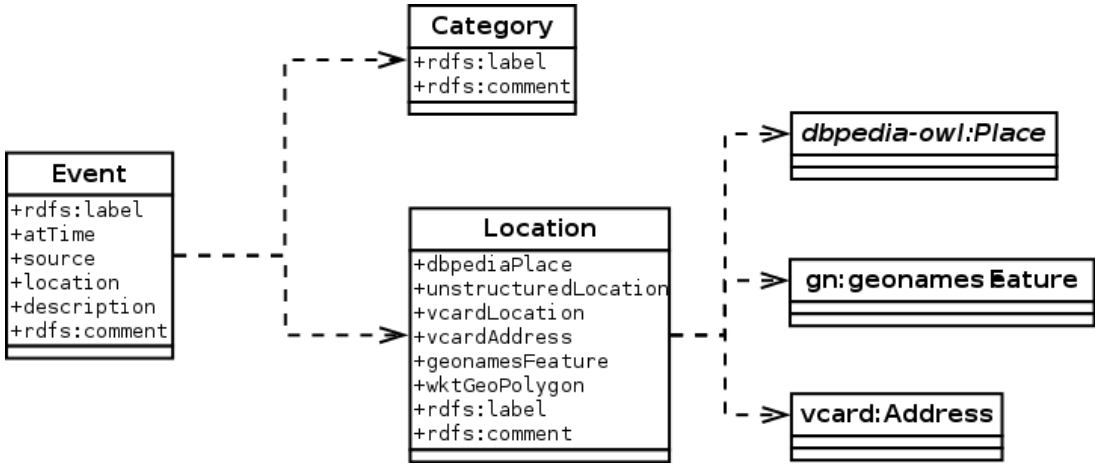


Figura 4.7: UML para o conjunto de dados relacionados à violência e criminalidade

Uma integração possível está representada na Figura 4.7 através da definição de *Place* da DBPedia [23]. Recolhendo dados geográficos através da API é possível utilizar a estratégia de integração para permitir uma visão global dos dados importados. Outros tipos de *mashups* que consideram outras aplicações de Dados Abertos também podem consumir a mesma base de dados utilizam documentos JSON [21]. O conjunto de informações coletadas na análise da taxonomia também constituem o número de tópicos a serem considerados na construção do modelo LDA.

O procedimento pode ser resumido nos seguintes itens:

1. Coletar os dados do Twitter utilizando a API de busca;

[6]Na lei brasileira somente o tráfico é considerado crime

2. Aplicar SRL aos *tweets coletados*;

3. Extrair os eventos da SRL;

4. Construir um corpus contendo somente os eventos coletados na SRL;

5. Construir o modelo LDA com base no corpus de eventos.

4.3.3 L3: Camada Social

Esta seção descreve a camada de aplicação da Social Machine construída como uma composição de dados oriundos de redes sociais, recebendo *semantic uplift* e computação humana (*human computation*). O conceito de *crowdsourcing* representa o núcleo do trabalho de computação humana: realizar inferências subjetivas sobre os dados obtidos. Trata-se do tipo de tarefa que não pode ser executada diretamente por computadores com a tecnologia atual. Assim, avaliação dos dados, interface para os usuários e acesso através de API no formato de dados abertos são as saídas esperadas do subsistema representado na camada L3.

A camada de Social Machine – SOCIAM [7] – deve fornecer um canal de comunicação em duas vias com os usuários: fornecer informações no domínio relativo a violência e criminalidade e colher *feedback* sobre a qualidade dos dados. A interação acontece através das seguintes interfaces:

API REST A API REST oferece uma interface de comunicação onde os usuários podem utilizar a base de dados construída coletivamente (*crowdsourced database*) para construir aplicações que utilizam a mesma fonte de dados, dentro dos princípios de dados abertos descritos na seção 2.2.1. Devem também permitir a coleta de *feedback* dos usuários. A troca será feita através de arquivos no formato JSON [21];

WWW Interface HTML para visualização dos dados. Fornece um exemplo de aplicação para *designers* de interface.

API REST (Dados Abetos)

A API REST pode ser definida em termos do modelo reduzido do exemplo 2.2, discutido na seção 2.3.1. Como a Social Machine não fornece informações sobre o estado, a variável S será desconsiderada. O quadro da Tabela 4.4 apresenta o modelo de interação simplificado resumido em operações numeradas. A variável $\{id\_doc\}$ representa o identificador

[7] O termo SOCIAM, além de representar um acrônimo para Social Machines, também é utilizada para representar um grupo de trabalho e um Workshop realizado na conferência WWW. Mais informações podem ser obtidas no endereço do Projeto: `http://sociam.org/`

único do status importado, enquanto a variável $\{select\}$ apresenta uma sintaxe especial para busca na estrutura do documento JSON. Um mecanismo de autenticação básico é fornecido através da variável $\{api\_key\}$, que fornece uma sessão a ser utilizada ao realizar as alterações via PUT.

Acrônimo	**Atributo**	Valor
Rel	Relationships	Twitter
Req	Requests	1. GET /status/doc 2. GET /status/doc/{id_doc} 3. GET /status/doc?$$={select} 4. POST /login?api_key={api_key} 5. PUT /status/doc/{id_doc}/positives 1 6. PUT /status/doc/{id_doc}/positives 1
Resp	Responses	1. JSON com a lista dos status; 2. JSON com todos os metadados relativos ao status; 3. JSON com resultado do processamento do SELECT; 4. HTTP Status 200 se autenticado com sucesso; 5. HTTP Status 200 se atualizado com sucesso; 6. HTTP Status 200 se atualizado com sucesso.
Const	Constraints	1. Não se aplica; 2. Não se aplica; 3. Não se aplica; 4. Conexão realizada necessariamente utilizando SSL; 5. Usuário deve possuir permissão para atualização; 6. Usuário deve possuir permissão para atualização.

Tabela 4.4: Definição da Social Machine no modelo do exemplo 2.2

Todas as operações são realizadas levando em consideração o conjunto de metadados do modelo do banco de dados semântico descrito no exemplo 4.2. Os status são armazenados em uma base que permite o formato JSON, identificando os dados da fonte original e seus metadados. A consulta por metadados é realizada através da utilização da instrução {*select*}, apresentada no exemplo 4.3. Seu objetivo é permitir a busca e apresentação de valores navegando na estrutura do documento.

Exemplo 4.3: Sintaxe do comando SELECT

```
{
    "select":[  ver<1>  ] <ou> "select": “”*,

    "filters":
        [
        {
        "field":"  ver<2>  ",
            "term":"  ver<3>  ",
            "operation":"  ver<4>  "
        },

        {
        "field":"  ver<2>  ",
            "term":"  ver<3>  ",
            "operation":"  ver<4>  "
        }
    ],

    "literal": "  ver<5>  ",

    "limit" : "   ver<6>   ",

    "offset" : "  ver<6>  ",

    "order_by":{"asc":[  ver<1>   ], "desc":[  ver<1>   ]}
}
```

A legenda dos comandos disponíveis na estrutura do JSON enviado como {*select*} é apresentada como se segue:

1. Apelido dos campos separados por vírgula. É possível também passar "*"definindo que se quer todos os campos;
2. Nome de um campo;
3. Termo a ser buscado;
4. Operadores (=, >=,<=, >, <, contains, like);
5. Condição de busca igual a um WHERE, não sendo possível usar palavras reservadas como (select, update, insert);

6. Valor numérico que limita a busca.

Para evitar a alteração do modelo da base de treinamento, as interações realizadas através da API REST têm como foco a apresentação e busca pelos resultados. A ideia é permitir ao desenvolvedor realizar buscas com critérios específicos e obter o conjunto de status que sirva à sua análise, assim como dados sobre a fonte original (Twitter). A disponibilização da interface caracteriza o sistema como uma aplicação de dados abertos, ainda que não possua todas as características necessárias para cumprir o modelo de *Linked Open Data*. Uma aplicação exemplo utilizando os dados abertos para construção da interface será apresentada na seção a seguir.

Interface WWW

Para construção da interface foram utilizados como exemplo duas aplicações relacionadas à criminalidade. Uma é o mapa de crimes da cidade de Belfast na Irlanda apresentado na Figura 4.8, baseado nos dados abertos fornecidos pelo município. Os círculos pretos apresentam a quantidade de crimes que foram identificados em determinada região, de forma que os cidadãos possam visualizar a taxa de criminalidade na região de maneira direta. A outra está representada na Figuras 4.9 e 4.10, extraídas do serviço Onde fui Roubado [8]. A aplicação disponibiliza uma interface para dispositivos móveis onde é possível ao cidadão reportar incidentes criminais em tempo real. O exemplo trata de uma garota que foi assaltada no caminho de volta para casa, e como o relato veio de um telefone celular, a informação de posição obtida pelo GPS do telefone permite a visualização precisa do local da ocorrência.

Apesar de se tratarem de bons exemplos, ambas as aplicações apresentam desafios que motivam o execução da pesquisa. O mapa de crimes da cidade de Belfast traz informações precisas, categorizadas e validadas por um oficial de polícia, pois é alimentada pelo sistema que realiza o registro de ocorrências policiais. Contudo, como foi apresentado na seção 3.5, já é um desafio por si só realizar o registro de ocorrência, principalmente no Brasil. Assim, os dados oficiais estão sempre defasados em relação à sensação dos cidadãos.

Já o serviço Onde Fui Roubado depende de informação fornecida pelos usuários, que devem baixar o aplicativo para seus telefones e escrever relatos precisos sobre as ocorrências. Apesar de se tratar de um serviço interessante e com interface simplificada, não é tão fácil para os usuários quanto simplesmente postar um relato no Twitter ou no facebook.

Considerando a arquitetura proposta e os exemplos apresentados, a implementação da interface que tenta obter o melhor de ambas as aplicações está apresentada na Figura 4.11. No exemplo os status são coletados através da API de busca do Twitter, processados

[8] `http://www.ondefuiroubado.com.br/`

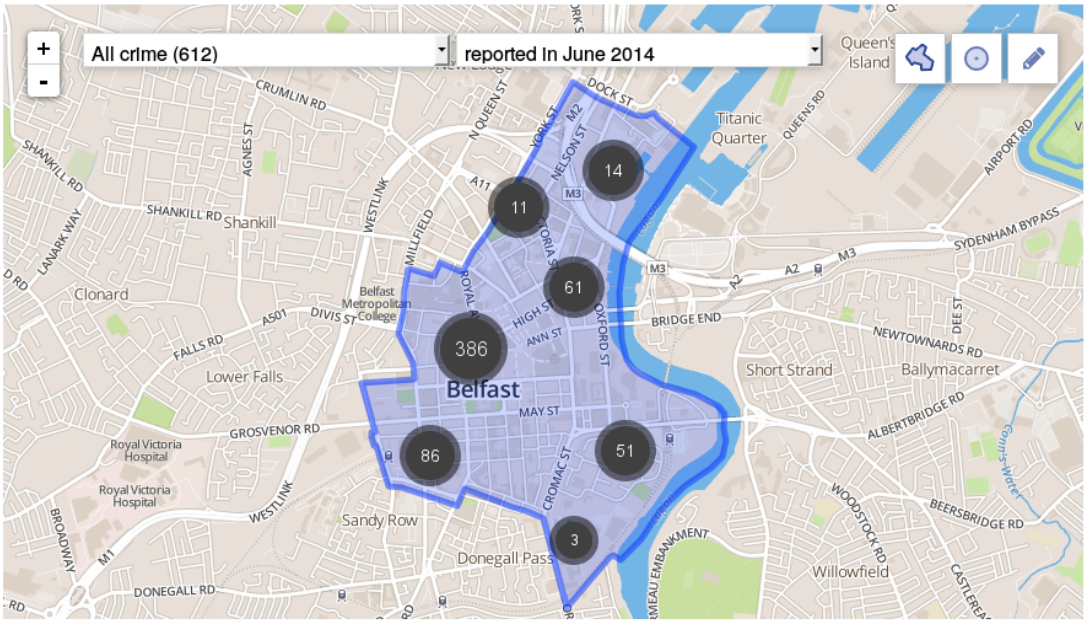


Figura 4.8: Mapa de crimes na cidade de Belfast disponível em `http://www.police.uk/northern-ireland/Central/crime/`

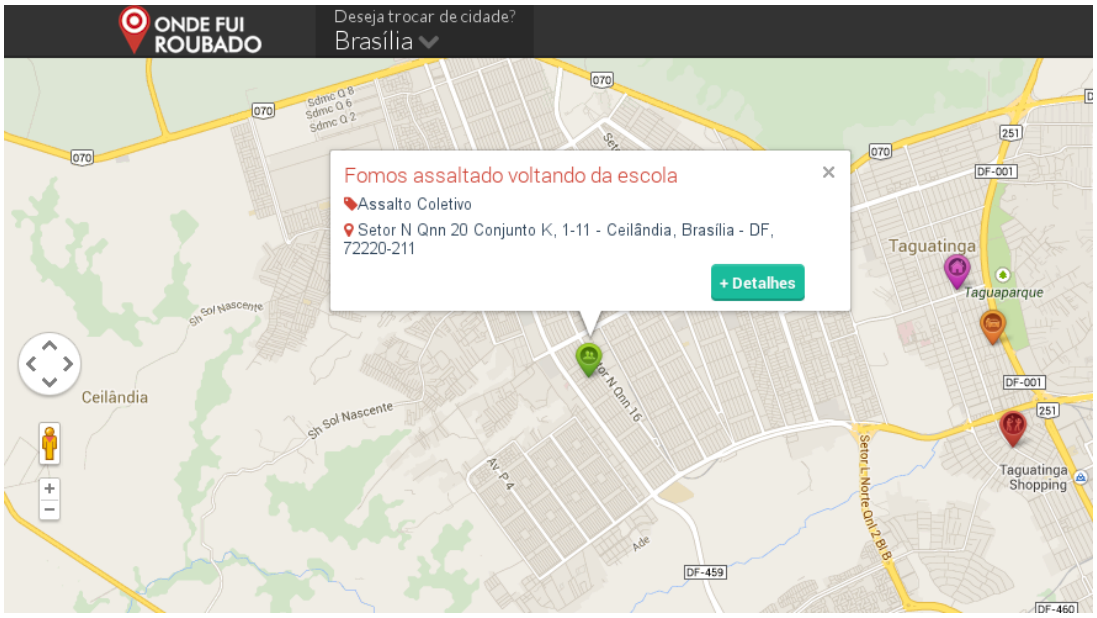


Figura 4.9: Mapa com crimes reportados pelos cidadãos

e traçados no Google Maps. É possível visualizar a distribuição de eventos pelo país observando a posição dos pontos.

Cada ponto identificado no mapa representa um status real obtido e processado. Ao clicar sobre o ponto é possível encontrar a fonte original e alguns metadados associados, conforme descrito na Figura 4.12. Na parte superior do lado esquerdo está apresentada a

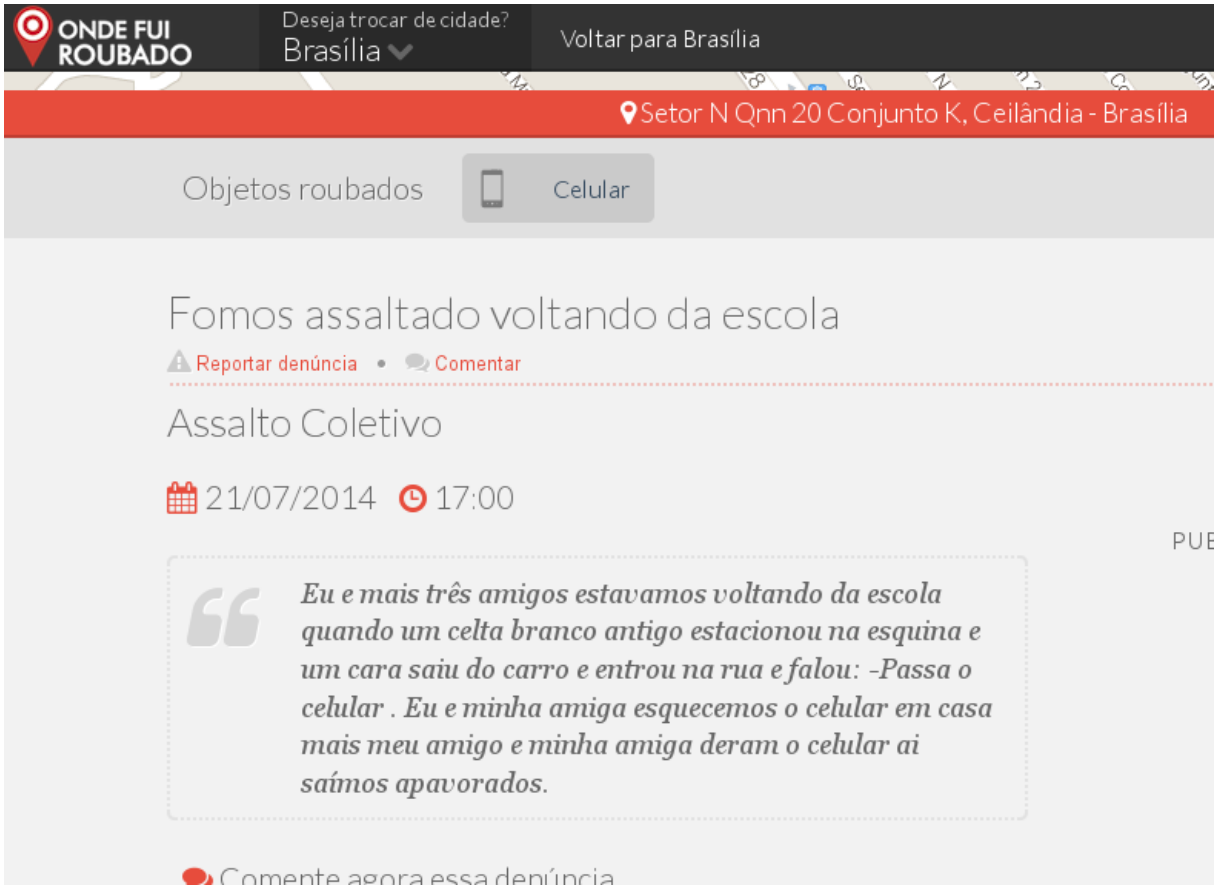


Figura 4.10: Exemplo de crime reportado pelos cidadãos

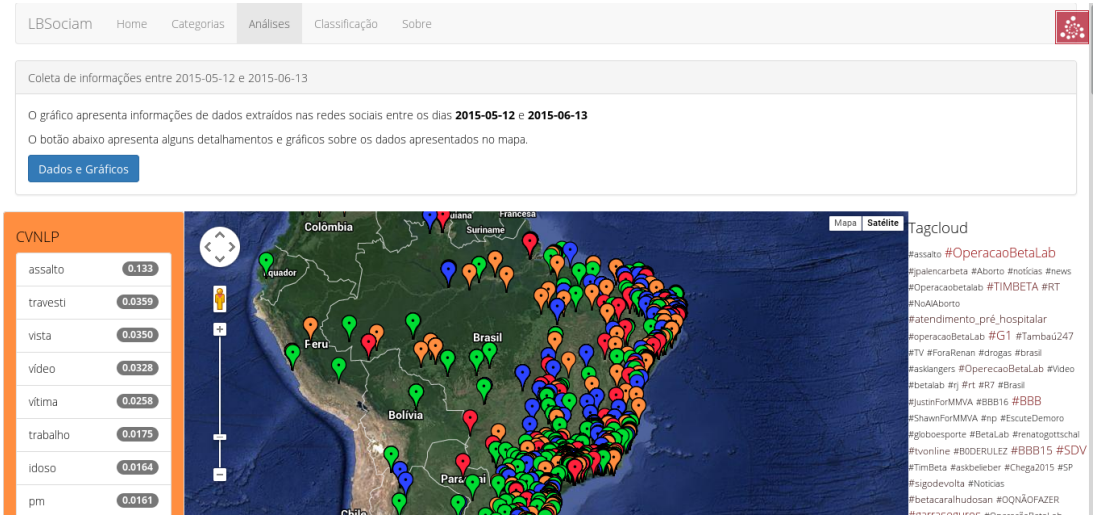


Figura 4.11: Mapa de eventos criminais identificados na base de treinamento

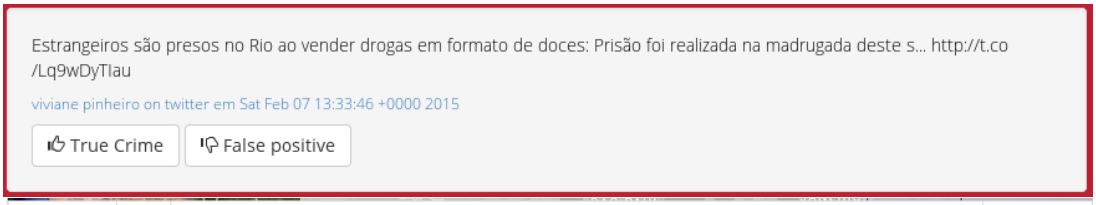
classificação relativa à categoria de crime identificada no status. No caso, trata-se de um crime de roubo.

A parte de coleta de *feedback* dos usuários está implementada com a interface da

Figura 4.12: Apresentação do status original

Figura 4.13. A interface de visualização apresenta a possibilidade de avaliação como positivo verdadeiro ou falso positivo, fornecendo números para o cálculo de *Precision*. A lista de status apresenta duas colunas, uma representando a quantidade de negativos e outra a quantidade de positivos. É de se esperar que um número maior de positivos que negativos indique que o status coletado trata de um evento relacionado a violência e criminalidade, fornecendo subsídios para o cálculo da taxa de *Precision*.

A classificação dos termos permite o cálculo do modelo LDA. Considerando os termos presentes na taxonomia reduzida da Figura 4.6, um conjunto de quatro categorias foi selecionado para criar a base de treinamento. Assim, o modelo de tópicos possui a identificação do tópico e a distribuição de probabilidade para cada termo dentro do tópico, como apresentado na Figura 4.14.



Date	+/-	Status	Origin
07/02/2015 10:17:57	0	#B0DERULEZ Segurança do governador morre em tentativa de assalto em Teresina: O homem trabalhava com a família... http://t.co/l71x1bms9p	twitter
07/02/2015 10:17:57	0	#B0DERULEZ Segurança do governador morre em tentativa de assalto em Teresina: O homem trabalhava com a família... http://t.co/l71x1bms9p	twitter
07/02/2015 10:47:07	0	Ação conjunta vai combater drogas, violência e doenças no carnaval de RO: Governo apresentou plano operacional... http://t.co/0fhDAlIJql	twitter
07/02/2015 10:17:57	0	#B0DERULEZ Segurança do governador morre em tentativa de assalto em Teresina: O homem trabalhava com a família... http://t.co/l71x1bms9p	twitter
07/02/2015 10:47:07	0	Ação conjunta vai combater drogas, violência e doenças no carnaval de RO: Governo apresentou plano operacional... http://t.co/0fhDAlIJql	twitter
07/02/2015 02:47:08	1	E lógico que o assassinato do Claudio ninguém nunca ia descobrir.. Ou melhor, ele podia ser preso e já era! :D #FelizesParaSempre	twitter
07/02/2015 10:17:57	0	#B0DERULEZ Segurança do governador morre em tentativa de assalto em Teresina: O homem trabalhava com a família... http://t.co/l71x1bms9p	twitter
07/02/2015 10:47:07	0	Ação conjunta vai combater drogas, violência e doenças no carnaval de RO: Governo apresentou plano operacional... http://t.co/0fhDAlIJql	twitter
07/02/2015 02:47:08	1	E lógico que o assassinato do Claudio ninguém nunca ia descobrir.. Ou melhor, ele podia ser preso e já era! :D #FelizesParaSempre	twitter
07/02/2015 04:46:19	1	E pro índio nada mais faz sentido Com tantas drogas porque só o seu cachimbo é proibido?	twitter
07/02/2015 10:17:57	0	#B0DERULEZ Segurança do governador morre em tentativa de assalto em Teresina: O homem trabalhava com a família... http://t.co/l71x1bms9p	twitter
07/02/2015 10:47:07	0	Ação conjunta vai combater drogas, violência e doenças no carnaval de RO: Governo apresentou plano operacional... http://t.co/0fhDAlIJql	twitter
07/02/2015 02:47:08	1	E lógico que o assassinato do Claudio ninguém nunca ia descobrir.. Ou melhor, ele podia ser preso e já era! :D #FelizesParaSempre	twitter
07/02/2015 04:46:19	1	E pro índio nada mais faz sentido Com tantas drogas porque só o seu cachimbo é proibido?	twitter

Figura 4.13: Classificação e apresentação dos resultados

Figura 4.14: Distribuição de probabilidade dos termos organizados na taxonomia

Capítulo 5

Outros resultados e validação

Este capítulo apresenta uma discussão sobre os resultados do trabalho, sumarizados na Tabela 1.3. A classificação da pesquisa apresentada na Tabela 1.1 construída a partir do modelo em [62] afirma que, para o modelo empírico, os resultados esperados podem ser classificados como "solução específica, protótipo, resposta ou julgamento". Assim, o principal resultado apresentado é a proposta de um modelo arquitetural para *social machines* no domínio de violência e criminalidade, já descrita e apresentada no capítulo 4. Durante as próximas seções o foco estará na apresentação dos subprodutos obtidos na construção do modelo arquitetural, que irão compor os outros resultados.

5.1 R_1 Protótipo funcional de Social Machine

É importante ressaltar que, como dissertação do Programa de Mestrado Profissional em Computação Aplicada da Universidade de Brasília, o produto desenvolvido é parte do portfólio de aplicações de software da empresa Lightbase, sendo também um componente do banco de dados documental Lightbase. O produto disponibiliza uma interface de consulta baseada em uma API REST, constituindo também um *backend* para a construção de aplicações que utilizam dados abertos.

Na seção 3.3, ao introduzir o modelo de dados de Twitter, é ressaltada a importância para as bases de dados que armazenam metadados dos status dos *tweets* possuírem flexibilidade suficiente para lidar com as mudanças. As alterações podem ser frequentes, inclusive no que tange à dimensionalidade das informações. O modelo de dados apresentado no exemplo 4.2 traz a representação no formato JSON, sinalizando que o sistema gerenciador de banco de dados responsável pelo armazenamento deve ter facilidade em trabalhar com o formato. É de supor que o modelo de bancos de dados relacionais não é o mais adequado, uma vez que "a estrutura do modelo relacional, ainda que seja efetiva

para muitas aplicações tradicionais, é considerada rígida demais ou pouco útil em outros casos, principalmente no que tange à utilização de dados semi-estruturados" [1].

O banco de dados documental Lightbase [1], cuja primeira versão completa 25 anos em 2015, é desenvolvido com base em dois conceitos chave:

1. Tratamento de dados semi-estruturados;

2. Busca e recuperação textual em língua portuguesa.

Como o foco do trabalho está no armazenamento e disponibilização, o sistema desenvolvido tem o objetivo de prover tratamento de dados semi-estruturados oriundos de redes sociais. Ao iniciar a modelagem do sistema de armazenamento para o Lightbase é preciso definir dois importantes conceitos:

Bases Definição do domínio e estrutura de dados que se deseja armazenar;

Documentos Coleção de instâncias da base e conjuntos de metadados associados.

Para exemplificar a utilização dos conceitos simulamos a primeira etapa do processo de modelagem com a seguinte pergunta: qual a informação que será armazenada no banco de dados? A pesquisa trata de dados relativos à violência e criminalidade, o que até agora foi tratado como domínio no texto. Assim, aplicando o conceito às definições apresentadas, é possível dizer que se trata da criação de uma **base** de status oriundos de redes sociais relacionados a violência e criminalidade. Cada status armazenado é um **documento**, ou seja, uma instância do domínio definido para a base contendo os metadados associados.

Uma diferença conceitual importante em relação aos bancos de dados NoSQL tradicionais que trabalham com metadados é relativa ao caráter não totalmente agnóstico das bases definidas no Lightbase. Antes de iniciar a inserção dos documentos, é preciso definir a estrutura da base, ou seja, o conjunto de metadados a serem inseridos e seus tipos de dados. Para os casos em que não é possível determinar totalmente a estrutura ou que sofrerão alterações ao longo do tempo, existe o campo flexível de tipo JSON que recebe qualquer tipo de dado. O modelo estrutural da Figura 5.1 descreve as bases, documentos, metadados e suas relações.

Ainda que o armazenamento aconteça num formato próximo ao modelo NoSQL, uma das principais características do Lightbase é fornecer um modelo híbrido que permita ao desenvolvedor aproveitar funcionalidades disponíveis em ambos relacional e não relacional. "A proposta deste projeto é resolver esses dois problemas centrais e os problemas a eles correlatos, demonstrando por meio de uma interface de programação de aplicação (API)

[1] Mais informações pode ser obtidas no site do Projeto: `http://www.lightbase.com.br`

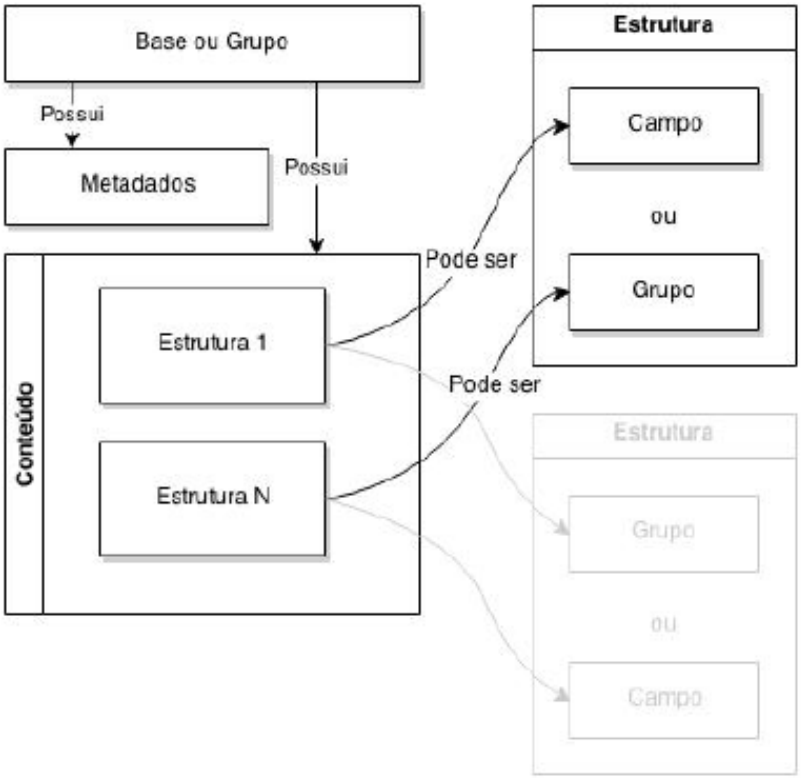


Figura 5.1: Modelo estrutural do esquema das bases [18]

com características de um modelo de dados não relacional ou NoSQL, integrada um banco de dados relacional" [18].

Para que seja possível implementar o modelo híbrido, internamente o sistema utiliza o modelo relacional simplificado descrito no diagrama de classes da Figura 5.2. É possível identificar os principais elementos do modelo: bases, campos e grupos. Enquanto as bases representam o domínio e possuem metadados específicos, como a data de criação e indexação, os campos representam o conjunto de metadados que poderão ser armazenados em um documento. Uma base seria então formada por campos com diferentes tipos de dados. Já o elemento de grupos permite a construção de conjuntos fixos de campos, que podem ser inseridos na base ou dentro de outros grupos, de forma a construir um modelo multi-dimensional. Utilizando a notação JSON, um grupo seria o equivalente à inserção de um JSON como campo dentro de outro, numa abstração similar à herança múltipla presente em linguagens orientadas a objeto.

Um outro princípio arquitetural bastante relevante é a independência de linguagem de programação. Para interagir com o banco de dados, utiliza-se uma API REST desenvolvida para trabalhar com os documentos do Lightbase, que segue os padrões de serviços Web [18]:

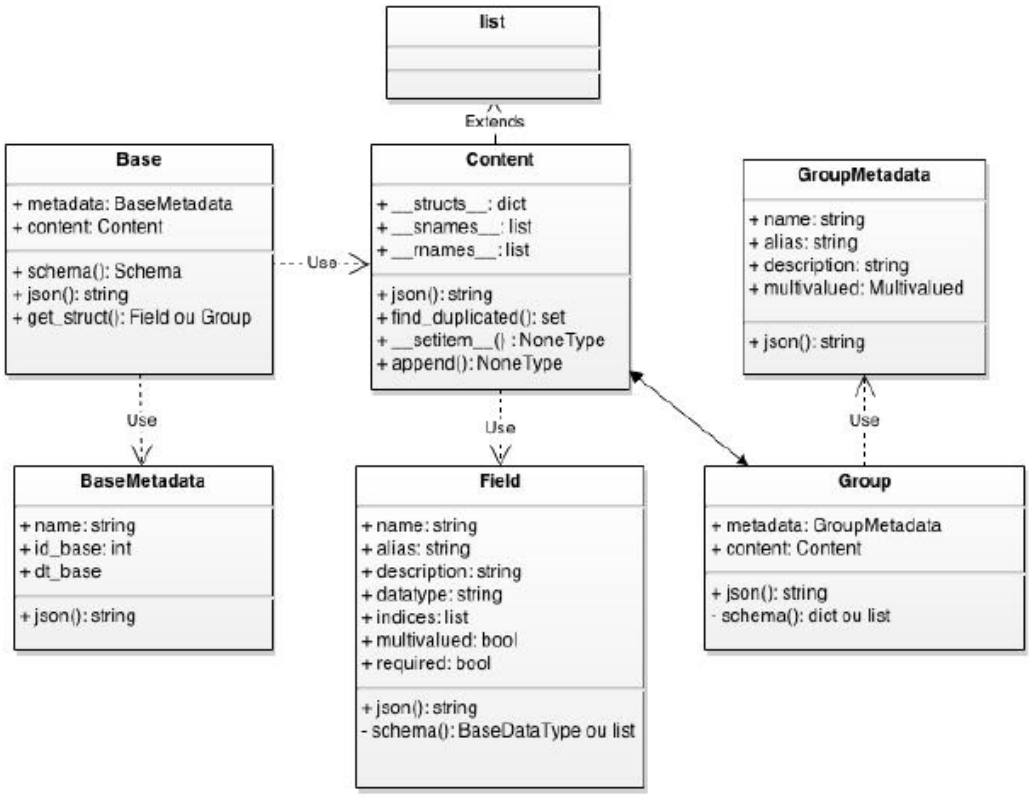


Figura 5.2: Diagrama de classes do modelo estrutural das bases de dados [18]

- Uma URI básica como http://example.com/;
- Um tipo de mídia da Internet para dados. Como já citado, a interface trabalha com JSON;
- Os métodos HTTP padronizados (GET, PUT, POST, DELETE).

A descrição da API pode ser melhor visualizada com a utilização do *Swagger*, cujo objetivo é "definir uma interface padrão, independente de linguagem de programação, que permita tanto humanos quanto computadores descobrir e entender as capacidades do serviço sem a necessidade de acessar código-fonte, documentação ou realizando análise de tráfego da rede" [63]. O exemplo das principais operações permitidas pela API e descritas com auxílio do *Swagger* está apresentado na Figura 5.3 [2].

O sistema de armazenamento e processamento de dados oriundos de redes sociais para identificação de eventos relacionados a violência e criminalidade, que será denominado **LBSociam**, representa a aplicação dos conceitos descritos no banco de dados documental Lightbase ao domínio de violência e criminalidade. O sistema obedece a premissa de, uma vez inserido na base, o status oriundo do Twitter sofrerá processamento para adição

[2]Uma versão da API para visualização no *Swagger* está disponível no endereço `http://api.brlight.net/api/api-docs`

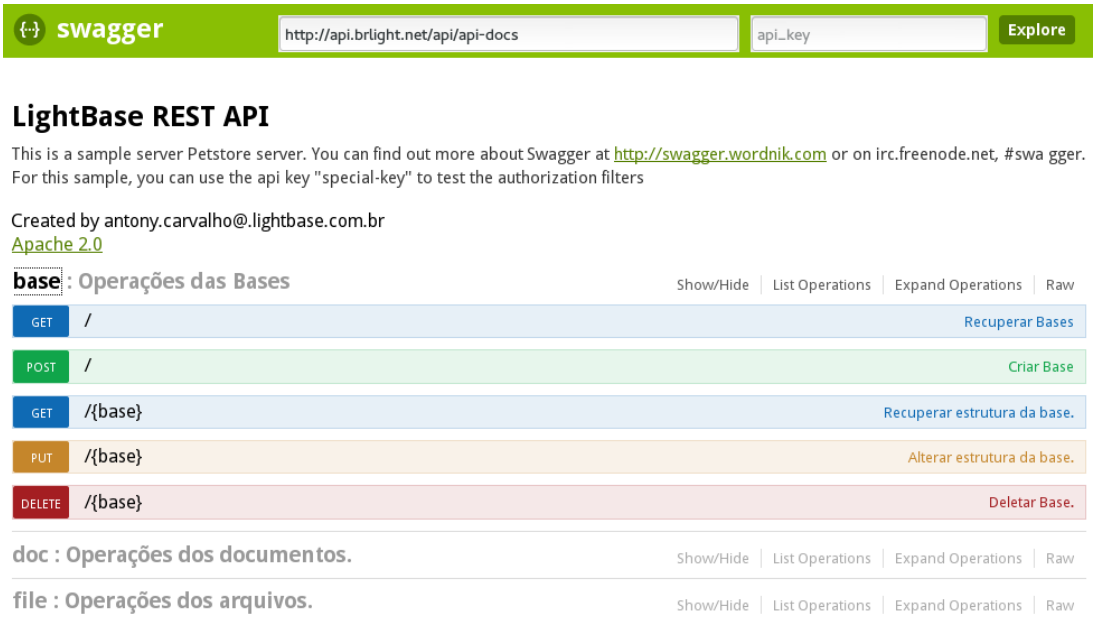


Figura 5.3: REST API do Lightbase descrita através do *Swagger*

de metadados até conter todas as informações necessárias para inclusão ou exclusão do domínio.

Obedecendo o sistema em camadas da seção 4.3, o módulo básico do LBSociam extrai as informações do Twitter utilizando a API de busca e insere numa base de dados ainda crua. No momento da inserção o status ainda não contém informações semânticas, conforme ilustrado no exemplo 5.1.

Exemplo 5.1: Primeira extração do Twitter

```
{
    "search_term": "drogas",
    "location": {},
    "events_tokens": [],
    "arg_structures": [],
    "origin": "twitter",
    "inclusion_datetime": "07/02/2015 13:46:04",
    "inclusion_date": "07/02/2015",
    "text": "\"@DecioNeves: Agentes da PF prendem jovens de classe émdia que vendiam
        drogas no Rio. http://t.co/uw2tjo4JB2 http://t.co/2hAdPadpJY\"",
    "hashtags": [ ],
    "tokens": [],
    "_metadata": {
        "dt_last_up": "25/03/2015 00:50:54",
        "dt_del": null,
        "dt_idx": null,
        "id_doc": 2364,
        "dt_doc": "07/02/2015 13:46:04"
    },
    "source": "[{(...)}]"
}
```

A segunda etapa do processamento envolve a aplicação de SRL para identificação de eventos e tokenização, gerando o documento do exemplo 5.2. Já é possível perceber a identificação do local pela aplicação do modelo da linguagem, onde observamos o conectivo AM-LOC na sintaxe do Propbank. Introduzimos então a conexão com mais um elemento da Web de dados, realizando a busca pelo local identificado na sentença utilizando a API de geolocalização do Google Maps [39].

Exemplo 5.2: Após a aplicação de SRL


```
{
    "search_term": "drogas",
    "location": {},
    "events_tokens": [
        "jovens",
        "agentes",
        "pf",
        "classe",
        "émdia",
        "vendiam",
        "drogas",
        "rio",
        "drogas"
    ],
    "arg_structures": [
        {
            "predicate": "prendem",
            "argument": [
                {
                    "argument_name": "A1",
                    "argument_value": [
                        "jovens"
                    ]
                },
                {
                    "argument_name": "A0",
                    "argument_value": [
                        "agentes",
                        "da",
                        "pf"
                    ]
                },
                {
                    "argument_name": "A2",
                    "argument_value": [
                        "de",
                        "classe",
                        "émdia",
                        "que",
                        "vendiam",
                        "drogas",
                        "no",
                        "rio"
                    ]
```

```
                },
                {
                    "argument_name": "V",
                    "argument_value": [
                        "prendem"
                    ]
                }
            ]
        },
        {
            "predicate": "vendiam",
            "argument": [
                {
                    "argument_name": "A1",
                    "argument_value": [
                        "drogas"
                    ]
                },
                {
                    "argument_name": "A0",
                    "argument_value": [
                        "que"
                    ]
                },
                {
                    "argument_name": "AM-LOC",
                    "argument_value": [
                        "no",
                        "rio"
                    ]
                },
                {
                    "argument_name": "V",
                    "argument_value": [
                        "vendiam"
                    ]
                }
            ]
        }
    ],
    "origin": "twitter",
    "inclusion_datetime": "07/02/2015 13:46:04",
    "inclusion_date": "07/02/2015",
    "text": "\"@DecioNeves: Agentes da PF prendem jovens de classe émdia que vendiam
        drogas no Rio. http://t.co/uw2tjo4JB2 http://t.co/2hAdPadpJY\"",
    "hashtags": [ ],
    "tokens": [
        "decioneves",
        "agentes",
        "prendem",
        "jovens",
        "classe",
        "émdia",
        "vendiam",
        "drogas",
        "rio",
        "uw2tjo4jb2",
```

```
            "hadpadpjy"
        ],
        "_metadata": {
            "dt_last_up": "25/03/2015 00:50:54",
            "dt_del": null,
            "dt_idx": null,
            "id_doc": 2364,
            "dt_doc": "07/02/2015 13:46:04"
        },
        "source": "[{(...)}]"
    }
```

A busca utiliza um procedimentos simples: envia como parâmetro as localizações identificadas pelo SRL e retorna somente a mais provável como resultado. Em seguida adiciona no metadado específico a latitude e longitude do local encontrado, além de identificar o nome e marcar a origem do dado. Após a aplicação do procedimento obtemos o dado de localização representado no exemplo 5.3.

Exemplo 5.3: Identificação do local através da API de Geocoding do Google Maps

```
{
    "search_term": "drogas",
    "location": {
        "id_location": 18,
        "longitude": -43.1728965,
        "city": "Rio de Janeiro",
        "loc_origin": "srl",
        "latitude": -22.9068467
    },
    "events_tokens": [
        "jovens",
        "agentes",
        "pf",
        "classe",
        "émdia",
        "vendiam",
        "drogas",
        "rio",
        "drogas"
    ],
    "arg_structures": [
        {
            "predicate": "prendem",
            "argument": [
                {
                    "argument_name": "A1",
                    "argument_value": [
                        "jovens"
                    ]
                },
                {
                    "argument_name": "A0",
                    "argument_value": [
                        "agentes",
                        "da",
```

```
                        "pf"
                    ]
                },
                {
                    "argument_name": "A2",
                    "argument_value": [
                        "de",
                        "classe",
                        "émdia",
                        "que",
                        "vendiam",
                        "drogas",
                        "no",
                        "rio"
                    ]
                },
                {
                    "argument_name": "V",
                    "argument_value": [
                        "prendem"
                    ]
                }
            ]
        },
        {
            "predicate": "vendiam",
            "argument": [
                {
                    "argument_name": "A1",
                    "argument_value": [
                        "drogas"
                    ]
                },
                {
                    "argument_name": "A0",
                    "argument_value": [
                        "que"
                    ]
                },
                {
                    "argument_name": "AM-LOC",
                    "argument_value": [
                        "no",
                        "rio"
                    ]
                },
                {
                    "argument_name": "V",
                    "argument_value": [
                        "vendiam"
                    ]
                }
            ]
        }
    ],
    "origin": "twitter",
    "inclusion_datetime": "07/02/2015 13:46:04",
```

```
    "inclusion_date": "07/02/2015",
    "text": "\"@DecioNeves: Agentes da PF prendem jovens de classe émdia que vendiam
        drogas no Rio. http://t.co/uw2tjo4JB2 http://t.co/2hAdPadpJY\"",
    "hashtags": [ ],
    "tokens": [
        "decioneves",
        "agentes",
        "prendem",
        "jovens",
        "classe",
        "émdia",
        "vendiam",
        "drogas",
        "rio",
        "uw2tjo4jb2",
        "hadpadpjy"
    ],
    "_metadata": {
        "dt_last_up": "25/03/2015 00:50:54",
        "dt_del": null,
        "dt_idx": null,
        "id_doc": 2364,
        "dt_doc": "07/02/2015 13:46:04"
    },
    "source": "[{(...)}]"
}
```


A última etapa de processamento envolve a classificação do status na taxonomia correta. A construção do modelo LDA é feita utilizando um corpus que contém todos os `events_tokens` identificados através da aplicação do SRL. O número de tópicos utilizado como entrada no modelo deve ser o mesmo número de elementos da taxonomia escolhidos para execução da busca:

LE trafico, drogas;

CVLI homicidio;

CVNLP furto, assalto, roubo;

CNLNICP agressao.

Importante observar que, para cálculo do modelo, são utilizados somente os status já avaliados descritos na seção 5.2. Com o conjunto de status avaliados construímos a **base de treinamento** para o modelo LDA.

A construção é feita a partir da execução dos seguintes passos:

1. Extração dos eventos utilizando SRL (Exemplo 5.4);

2. Construção do corpus de eventos (Exemplo 5.5);

3. Cálculo do modelo LDA utilizando o software gensim (Exemplo 5.6);

4. Identificação do status na categoria mais provável (Exemplo 5.7).

Exemplo 5.4: Extração de eventos

```
def get_events_tokens(self):
        """
        Get events corpus
        :return: Events corpus
        """
        orderby = OrderBy(asc=['id_doc'])
        select = ['events_tokens']
        search = Search(
            select=select,
            limit=None,
            order_by=orderby,
            offset=0
        )
        url = self.documentrest.rest_url
        url += "/" + self.lbbase._metadata.name + "/doc"
        vars = {
            '$$': search._asjson()
        }

        # Envia çãrequisio para o REST
        response = requests.get(url, params=vars)
        collection = response.json()
        saida = list()

        # Cria uma lista de resultados como ID
        if collection.get('results') is None:
            return None

        for results in collection['results']:
            if results is not None:
                saida.append(results['events_tokens'])

        return saida
```

Exemplo 5.5: Construção do corpus de eventos

```
@property
def corpus(self):
        """
        Get corpus
        :return: Formatted corpus
        """
        return [self.dic.doc2bow(text) for text in self.events_tokens]
```

Exemplo 5.6: Cálculo do modelo LDA

```
def crime_topics(
        status_base,
        crimes_base,
```

```
        n_topics=4):
    """
    Generate crime topics
    :return: dict with term frequency calculated by LDA
    """
    t0 = time.clock()
    c = corpus.get_events_corpus(status_base)
    t1 = time.clock() - t0
    log.debug("Time to generate Corpus: %s seconds", t1)

    t0 = time.clock()
    lda = ldamodel.LdaModel(c.corpus, id2word=c.dic, num_topics=n_topics)
    t1 = time.clock() - t0
    log.debug("Time to generate LDA Model for %s topics: %s seconds", n_topics, t1)

    topics_list = lda.show_topics(num_topics=n_topics, formatted=False)
    base_info = status_base.get_base()
    total_status = int(base_info['result_count'])

    saida = dict()
    i = 0

    # Now we allow the status to be in only one category
    # Consider the highest probability
    found_categories = list()
    for elm in topics_list:
        saida[i] = dict()
        saida[i]['tokens'] = list()
        for token in elm:
            probability = token[0]
            word = token[1]
            if total_status is not None:
                token_dict = dict(
                    word=word,
                    probability=probability,
                    frequency=probability*total_status
                )
            else:
                token_dict = dict(
                    word=word,
                    probability=probability,
                    frequency=None
                )
            saida[i]['tokens'].append(token_dict)

            # Get category if we didn't find it yet
            if saida[i].get('category') is None:
                category = crimes_base.get_token_by_name(word)
                if category is None:
                    continue
                if category['category_name'] not in found_categories:
                    found_categories.append(category['category_name'])
                    saida[i]['category'] = crimes_base.get_token_by_name(word)

                    # Finish searching
                    continue
```

```
        i += 1

    return saida
```

Exemplo 5.7: Identificação na taxonomia

```
def get_category(status,
                 status_base,
                 crimes_base,
                 n_topics=4):
    t0 = time.clock()
    c = corpus.get_events_corpus(status_base)
    t1 = time.clock() - t0
    log.debug("Time to generate Corpus: %s seconds", t1)

    t0 = time.clock()
    lda = ldamodel.LdaModel(c.corpus, id2word=c.dic, num_topics=n_topics)
    t1 = time.clock() - t0
    log.debug("Time to generate LDA Model for %s topics: %s seconds", n_topics, t1)

    # Produce sorted list of probabilities
    vec_bow = c.dic.doc2bow(status['events_tokens'])
    vec_lda = lda[vec_bow]
    sorted_vec_lda = sorted(vec_lda, key=operator.itemgetter(1), reverse=True)

    # Get categories
    category_list = crime_topics(
        status_base,
        crimes_base,
        n_topics
    )

    # This will the topic with highest probability
    category_index = sorted_vec_lda[0][0]
    category = category_list[category_index]

    # Add this category back to status
    status['category'] = {
        'category_id_doc': category['category']['_metadata']['id_doc'],
        'category_probability': sorted_vec_lda[0][1]
    }

    return status
```

Após o cálculo da probabilidade, a categoria mais provável no modelo LDA construído a partir da base de treinamento é inserida de volta como um metadado, obtendo o modelo completo do Exemplo 5.8. O identificador está relacionado a uma base de categorias associada, que será útil para construção da interface descrita na próxima seção.

Exemplo 5.8: Identificação do local através da API de Geocoding do Google Maps

```
{
"category": {

    "category_id_doc": 4,
```

```
    "category_probability": 0.49975645161659965

},
    "search_term": "drogas",
    "location": {
        "id_location": 18,
        "longitude": -43.1728965,
        "city": "Rio de Janeiro",
        "loc_origin": "srl",
        "latitude": -22.9068467
    },
    "events_tokens": [
        "jovens",
        "agentes",
        "pf",
        "classe",
        "émdia",
        "vendiam",
        "drogas",
        "rio",
        "drogas"
    ],
    "arg_structures": [
        {
            "predicate": "prendem",
            "argument": [
                {
                    "argument_name": "A1",
                    "argument_value": [
                        "jovens"
                    ]
                },
                {
                    "argument_name": "A0",
                    "argument_value": [
                        "agentes",
                        "da",
                        "pf"
                    ]
                },
                {
                    "argument_name": "A2",
                    "argument_value": [
                        "de",
                        "classe",
                        "émdia",
                        "que",
                        "vendiam",
                        "drogas",
                        "no",
                        "rio"
                    ]
                },
                {
                    "argument_name": "V",
                    "argument_value": [
                        "prendem"
```

```
                    ]
                }
            ]
        },
        {
            "predicate": "vendiam",
            "argument": [
                {
                    "argument_name": "A1",
                    "argument_value": [
                        "drogas"
                    ]
                },
                {
                    "argument_name": "A0",
                    "argument_value": [
                        "que"
                    ]
                },
                {
                    "argument_name": "AM-LOC",
                    "argument_value": [
                        "no",
                        "rio"
                    ]
                },
                {
                    "argument_name": "V",
                    "argument_value": [
                        "vendiam"
                    ]
                }
            ]
        }
    ],
    "origin": "twitter",
    "inclusion_datetime": "07/02/2015 13:46:04",
    "inclusion_date": "07/02/2015",
    "text": "\"@DecioNeves: Agentes da PF prendem jovens de classe émdia que vendiam
        drogas no Rio. http://t.co/uw2tjo4JB2 http://t.co/2hAdPadpJY\"",
    "hashtags": [ ],
    "tokens": [
        "decioneves",
        "agentes",
        "prendem",
        "jovens",
        "classe",
        "émdia",
        "vendiam",
        "drogas",
        "rio",
        "uw2tjo4jb2",
        "hadpadpjy"
    ],
    "_metadata": {
        "dt_last_up": "25/03/2015 00:50:54",
        "dt_del": null,
```

```
            "dt_idx": null,
            "id_doc": 2364,
            "dt_doc": "07/02/2015 13:46:04"
        },
        "source": "[{(...)}]"
    }
```

Ao final da execução o modelo LDA fornece o conjunto de tópicos e os elementos associados a cada tópico. Em uma base de treinamento bem escolhida é de se esperar que os termos da taxonomia "flutuem" para o topo da distribuição de probabilidade em cada tópico, de forma que seja possível identificar facilmente a qual termo está associado o tópico. A Tabela 5.1 mostra os cinco elementos mais prováveis em cada tópico, associando-os também aos termos da taxonomia reduzida descrita na seção 4.3.2.

LE – Leis Especiais

Termo	**Probabilidade**
drogas	0.116
homem	0.0152
assaltos	0.0138
brasil	0.0121
leves	0.0108

CVNLP – Crimes Violentos Não Letais contra o Patrimônio

Termo	**Probabilidade**
assalto	0.137
vista	0.0315
boa	0.0297
suposto	0.0296
vídeo	0.0280

CVLI – Crimes Violentos Letais Intencionais

Termo	**Probabilidade**
homicídio	0.0743
após	0.0490
pm	0.0256
desta	0.0207
agredido	0.0199

CNLNICP – Crimes Não Letais Intencionais Contra a Pessoa

Termo	**Probabilidade**
agressão	0.0777
travesti	0.0523
c.	0.0406
drogas	0.0322
morto	0.0213

Tabela 5.1: Distribuição de probabilidade para os termos da taxonomia

5.2 R_2 "Crowdsourcing system" construído a partir de modelos de identificação de atividades criminais

O último subproduto, apresentado no resultado R_2, tenta resolver a lacuna identificada em [25]:

> Esperamos que a utilização de *crowdsourcing* para construção bancos de dados e serviços estruturados (Web services com entradas e saídas formais) recebam cada vez mais atenção.

O resultado supõe o desenvolvimento de uma arquitetura de colaboração que seja capaz de utilizar os dados fornecidos pelos usuários como um elemento computacional. Processar as informações fornecidas pelos usuários nas redes sociais para criar o conjunto de eventos a serem encaixados na taxonomia representam um tipo de *crowdsourcing*. O procedimento é descrito na seção 4.3.3 como a camada social que permite tanto a interação direta, ao manipular os dados através da interface, como a indireta, ao fornecer dados para a base de informações criminais.

A criação de um serviço estruturado deve levar em consideração a proposta de criação de um *framework* para identificação de atividades criminais utilizando *Data Mining* [19]. Os autores realizam uma análise sobre a performance das diferentes técnicas, apresentando a extração de entidades (*entity extraction*) como a mais eficiente em relação ao espectro mais amplo dos tipos de crime. Ou seja, é capaz de identificar crimes com performance aceitável em diferentes categorias.

Para testar a performance da identificação de crimes de acordo com o termo da taxonomia a ser utilizado para realizar buscas no Twitter, uma coleta preliminar foi realizada entre os dias 23/11/2014 e 12/12/2014 utilizando a API de busca, onde foram coletados um total de **33997** *tweets*. Os seguintes termos foram utilizados:

- assassino;
- crime;
- homicídio;
- roubo;
- furto;
- assalto;
- tráfico;
- arma;
- estupro;
- agressão.

Após a execução da coleta, o modelo SRL foi aplicado para testar a identificação de eventos, conforme modelo proposta na seção 4.3.1. Os resultados obtidos estão sumarizados na Tabela 5.2. Para a análise da acurácia utilizamos a taxa de *recall*, uma vez que temos apenas duas informações: total de *tweets* corretamente identificados (positivos verdadeiros) e total de *tweets* onde os eventos não foram identificados, mas deveriam (falso

negativo). Os autores do módulo nlpnet [35] reportam um taxa de *recall* entre 58,10% e 62,20% dependendo do tipo de texto a ser fornecido. Para a identificação do primeiro evento, reportam ainda uma taxa de 71,71%.

Positivos verdadeiros	**Falsos negativos**	**Recall**
26634	7363	0,78
Total de tweets obtidos: **33997**		

Tabela 5.2: Identificação de eventos para primeira coleta

Para calcular a taxa de *precision* das coletas realizadas a estratégia manual foi selecionada: escolhe-se uma amostra dos dados, realiza-se a classificação manual e calcula-se os resultados. Para facilitar o trabalho de classificação foi utilizada a interface da Figura 4.13, considerando somente os dados da coleta realizada. O objetivo era agilizar o trabalho de determinação da taxa de *precision* avaliando múltiplos *tweets* ao mesmo tempo.

Contudo, ao navegar pelos dados coletados para realizar a classificação, foi detectado um problema na seleção dos termos de pesquisa. O termo **crime** foi utilizado na busca e, como podemos ver na taxonomia da Figura 4.6 trata-se de um termo que está no topo da taxonomia. Seguindo os conceitos de ontologia [53], se há uma relação de classe e subclasse entre os termos da taxonomia pressupõe-se que os termos de nível inferior herdem os conceitos de nível superior. Assim, é possível inferir que todo assalto é um crime, por exemplo.

A partir da identificação da questão, a decisão tomada foi por buscar somente termos que estivessem no mesmo nível da taxonomia. Para definir os termos foi necessário realizar uma série de coletas para encontrar o conjunto de palavras associadas que obtivesse melhor performance. Para efeito de avaliação foi adotada a taxa de *precision* obedecendo os seguintes critérios:

Positivo verdadeiro *tweet* classificado na categoria correta;

Falso positivo *tweet* classificado na categoria, mas que não está realmente relacionado ao tema.

Foi utilizado como fator de corte uma taxa de *precision* menor do que a prevista no módulo nlpnet, que variam entre 66,95% e 69,01% dependendo do tipo de texto a ser analisado. As coletas foram realizadas no dia 06 de fevereiro de 2015 e os resultados estão sumarizados na Tabela 5.3. Apesar de se tratar de uma amostra pequena, o objetivo era executar apenas um teste simplificado para verificar possíveis inconsistências na escolha de termos, não se tratando do resultado final da pesquisa.

CVLI – Crimes Violentos Letais e Intencionais			
Termos buscados	**Número de tweets**	**Falsos positivos**	**Precision**
homicidio, assassinato	53	41	0,56
homicidio	30	12	0,71

CVNLP – Crimes Violentos Não Letais contra o Patrimônio			
Termos buscados	**Número de tweets**	**Falsos positivos**	**Precision**
furto, roubo, assalto	43	18	0,71

CNLNICP – Crimes Não Letais Intencionais Contra a Pessoa			
Termos buscados	**Número de tweets**	**Falsos positivos**	**Precision**
agressao	32	13	0,71

CCLS – Crimes Contra a Liberdade Sexual			
Termos buscados	**Número de tweets**	**Falsos positivos**	**Precision**
estupro	42	12	0,77

LE – Leis Especiais			
Termos buscados	**Número de tweets**	**Falsos positivos**	**Precision**
trafico, drogas, arma	66	48	0,58
trafico, drogas	42	13	0,76

Tabela 5.3: Cálculo de *precision* para os termos da taxonomia

A definição da base de dados para extração de entidades considerou a coleta de informações utilizando a API de busca do Twitter realizada entre os dias 07 e 09 de fevereiro de 2015. Foram coletados no total **11456** tweets utilizando os seguintes termos da taxonomia descrita na seção 4.3.2:

LE trafico, drogas;

CVLI homicidio;

CVNLP furto, assalto, roubo;

CNLNICP agressao.

A acurácia na identificação dos eventos está apresentada na Tabela 5.4. Será utilizada novamente a taxa de *recall* para efeito de comparação, utilizando os mesmos critérios descritos para a Tabela 5.2. O valor obtido de **0,81** para a taxa de *recall* está próximo do relatado pelos autores do módulo nlpnet.

Agora é possível analisar a acurácia dos termos da taxonomia de maneira geral. Seguindo a premissa de que os termos de um nível inferior são subclasses de níveis inferiores,

Positivos verdadeiros	**Falsos negativos**	**Recall**
9275	2181	0,81
Total de tweets obtidos: **11456**		

Tabela 5.4: Identificação dos eventos na segunda coleta

e partindo da hipótese de que não há sobreposição de termos, os dados da segunda coleta foram utilizados para cálculo da acurácia na identificação de atividades criminais. O procedimento é o mesmo descrito na Figura 4.13 e os dados estão apresentados na Tabela 5.5. Para tornar a análise mais precisa estão considerados somente o total de *tweets* classificados manualmente, desconsiderando do cálculo os que não foram classificados em nenhum momento. Para o cálculo foi considerada novamente a taxa de *precision*, que leva em conta somente os positivos e falsos positivos.

Positivos verdadeiros	**Falsos positivos**	**Precision**
792	158	0,83
Total de tweets considerados: **958**		

Tabela 5.5: Acurácia na identificação de atividades criminais

Os dados apontam para uma taxa de *precision* de 0,83 ou 83% na identificação de atividades criminais. O modelo de referência SRL + LDA do trabalho [72] utiliza a técnica com um conjunto limitado de dados para prever a probabilidade da ocorrência de acidentes com carros. Os autores sustentam que a utilização da busca por eventos aumenta a performance do modelo preditivo.

Como o objetivo do trabalho não é realizar um modelo preditivo, e sim apontar a relação entre o que está sendo comentado nas redes sociais com atividade criminal, não é possível estabelecer uma comparação direta. Já o módulo nlpnet reporta uma taxa de *precision* de 76,72% na identificação de eventos.

Capítulo 6

Experiência e Avaliação

Como a validação dos resultados será feita com foco na aplicação protótipo, é possível classificá-la como uma pesquisa empírica. A Tabela 1.1, que apresenta a estratégia de pesquisa, traz o desenvolvimento do protótipo como instrumento de validação. Uma revisão sistemática dos trabalhos em Engenharia de Software [62] explica como a validação de uma técnica ou procedimento através da estratégia *slice of life* pode ser realizada com a implementação de um protótipo:

> O protótipo do tipo *slice of life* tende a ser convincente, especialmente se acompanhado de uma explicação de porque o exemplo desenvolvido contém a essência do problema a ser resolvido.

O capítulo descreve as estratégias de validação, discorre sobre a avaliação dos resultados e sobre como estão relacionados com a pergunta de pesquisa.

6.1 V_1. Descrição da arquitetura a partir dos modelos

A regra de validação V_1 tem como objetivo validar a seguinte hipótese: é possível definir um modelo arquitetural para social machines. A validação pode ser feita em duas partes: desenvolvendo uma proposta para utilização no domínio de violência e criminalidade e testando sua aderência em relação aos modelos existentes.

O primeiro modelo abordado diz respeito à utilização da EBSE para o planejamento de pesquisas em Engenharia de Software. O planejamento da implementação descrito na seção 4.2 mostra como as diretrizes da EBSE auxiliaram o processo, principalmente no que diz respeito à escolha da população e teste dos modelos de extração. Assim, a execução foi realizada nas seguintes etapas:

1. **Escolha da população** Selecionar corretamente a população influencia fortemente o trabalho de extração das informações, uma vez que é preciso conhecer os dados que

estão disponíveis para serem tratados. No caso, o Twitter fornece várias informações que são relevantes, como o texto original, data de inclusão, imagens, URL e outros [67].

2. **Seleção do domínio** Trabalho de escolha dos termos e suas relações dentro do domínio. Ainda que não seja necessário construir uma ontologia completa e expressiva, uma taxonomia contendo até o segundo ou terceiro nível é importante;

3. **Processamento/armazenamento** Antes de implementar o modelo, é preciso saber se as técnicas selecionadas são válidas e possuem acurácia aceitável para os termos do domínio;

4. **Interface** Como os dados serão visualizados e/ou disponibilizados?

Antes de entrar na etapa 4, de planejamento da interface, é importante a realização de um pré-experimento, a fim de garantir que o modelo proposto é viável. Via de regra as etapas de 1 a 3 devem ser repetidas quantas vezes forem necessárias, até obter números relativos às taxas de *precision* e *recall* que sejam suficientes para prosseguir. No caso, alteramos os parâmetros de busca até garantir que os termos obtidos possuíssem performance igual ou superior às fontes originais.

A definição e validação dos modelos de *social machines* necessitam de uma análise mais detalhada. Alguns trabalhos apresentam casos de estudo [26] sobre a aplicação dos conceitos de *Social Machines* no desenvolvimento real de aplicações, principalmente no que tange à validação de modelos formais [16]. Ainda que tenha sido possível desenvolver uma aplicação real no modelo sugerido, os autores reforçam a necessidade de construção de mais estudos de caso. O objetivo final seria "construir um *framework* arquitetural para definição e desenvolvimento de sistemas baseados em *Social Machines* tratando questões como segurança, cobrança, monitoramento e tolerância a falhas, entre outros".

O trabalho é complementado na definição do estilo arquitetural SoMar [17], construído como uma "combinação de princípios diferentes oriundos de estilos arquiteturais existentes, restritos pela visão unificada de Social Machines". Como diretriz de desenvolvimento, o estilo apresenta um modelo em quatro passos:

1. **Definição dos componentes sociáveis** Escolha das partes do sistema que devem ser sociáveis, ou seja, quais componentes podem ser encapsulados em uma API para serem definidos como uma entidade passível de conexão;

2. **Especificação dos serviços fornecidos** Quais os serviços a serem fornecidos pela *Social Machine*? Podem ser especificados a partir dos tipos de requisição (GET, POST, PUT, etc) e seus parâmetros;

3. **Projeto de integrações** Como os dados serão abstraídos e mapeados entre os diferentes serviços?

4. **Projeto de modelos de interação** Como os serviços se integram? Quais os protocolos de comunicação?

Analisando ambos os modelos em quatro passos, é possível identificar importantes similaridades. A população ou fonte de dados pode ser vista também como um componente sociável, já que redes sociais podem ser consideradas a primeira geração de Social Machines [16]. O segundo passo apresenta uma diferença relevante no que diz respeito à necessidade de estabelecimento do domínio. Do ponto de vista do programador é importante saber, logo no início do processo de desenvolvimento, quais as operações que o sistema está sendo desenhado para executar. Contudo, do ponto de vista do cidadão, a questão central é o problema social que o sistema está sendo desenhado para abordar. Embora ambos pontos de vista não sejam excludentes, trata-se de uma diferença importante na concepção da aplicação, uma vez que a escolha de um domínio de informação envolve uma análise ontológica. Ainda que simplificada, a análise apresentada na seção 4.3.2 reforça a inserção dos resultados em um domínio social.

A etapa de processamento e armazenamento das informações possui um paralelo direto com o projeto da integração proposto no estilo SoMar (Figura 6.1). A utilização de fontes diferentes de dados está no núcleo da proposta com a diferença sutil, mas importante, da presença de um banco de dados único contendo o resultado de todas as técnicas de processamento. Tal banco tem como objetivo fornecer a informação por si mesma como um serviço, dentro dos conceitos de dados abertos descritos na seção 2.2.1.

Já a definição das interfaces de visualização/apresentação possuem conceitos similares aos defendidos nos projetos de modelos de interação descritos no SoMar. O objetivo é disponibilizar formas de apresentação dos dados de forma a atender diferentes perfis de usuários e/ou dispositivos computacionais.

No que tange à arquitetura do sistema, o estilo arquitetural SoMar [17] prevê a aplicação do modelo MVC, pressupondo a existência das seguintes camadas:

Model Camada de encapsulamento dos dados, responsável por receber e processar dados oriundos de diversas unidades computacionais;

Controller Camada que contém os serviços fornecidos pela Social Machine. O serviço Gestor de Relacionamentos (*Relationship Manager*) é responsável por prover a integração entre as diferentes camadas e/ou Social Machines;

View Camada que expõe os dados construídos pela Social Machine para diferentes interfaces e/ou dispositivos, como interfaces WWW e outras Social Machines.

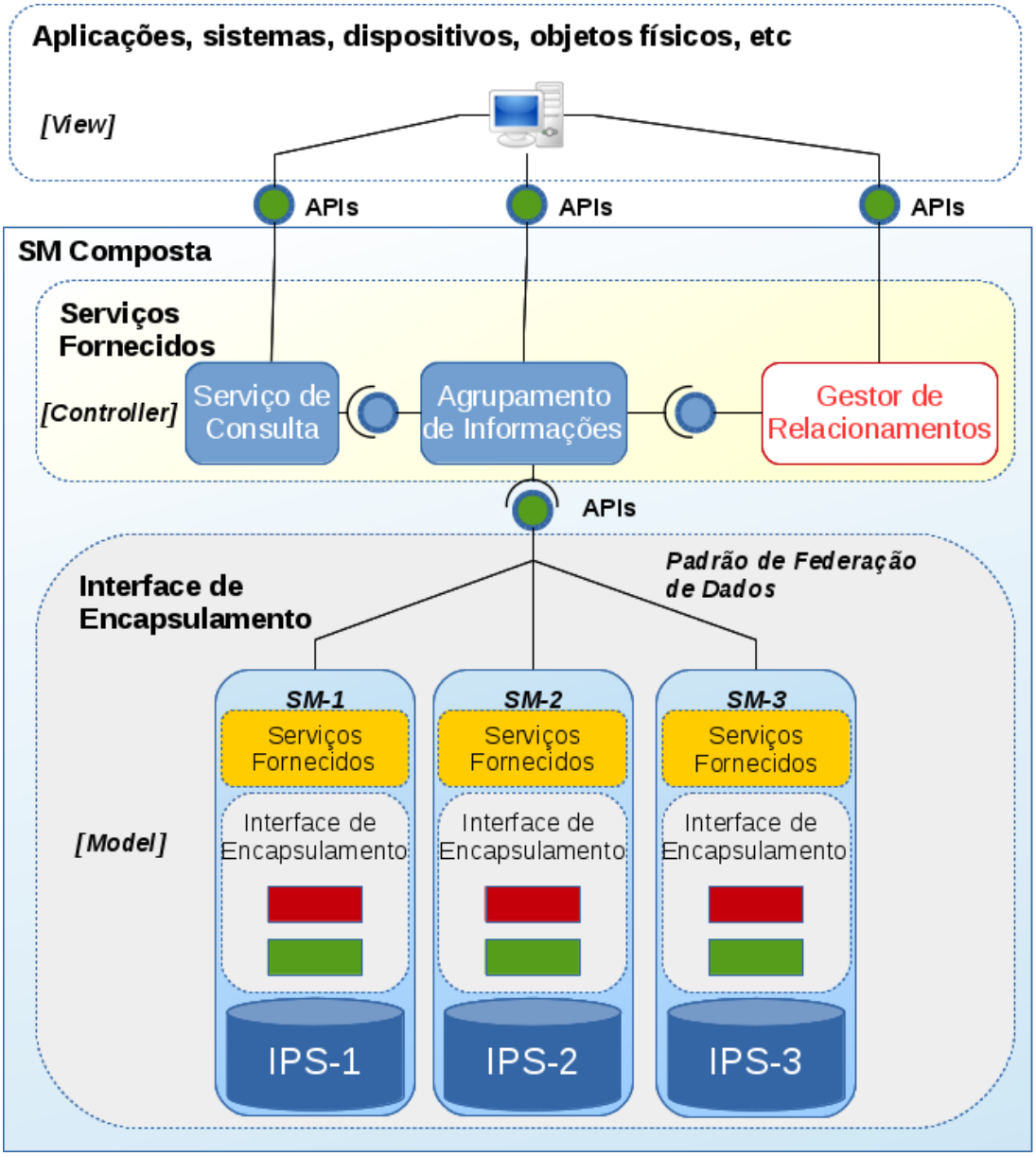


Figura 6.1: Modelo MVC SoMar [17]

A camada de encapsulamento representa o núcleo do modelo de Social Machine construído na pesquisa. A arquitetura proposta na seção 4.3 prevê o desenvolvimento de um sistema em três camadas, sendo elas:

L1 Camada de extração;

L2 Banco de dados de informações criminais;

L3 Camada social.

Realizando uma comparação com o estilo arquitetural SoMar é possível construir uma relação de sobreposição de camadas, construindo o quadro de similaridades da Tabela 6.1, contendo os seguintes elementos:

L1/Controller Implementação do gestor de relacionamentos através da API do Twitter e disponibilização de um serviço de consulta aos dados;

L2/Model Extração e processamento dos dados utilizando SRL + LDA em uma interface de encapsulamento. Gestor de persistência no Lightbase;

L3/View Disponibilização dos dados pela REST API do Lightbase e desenvolvimento de interface WWW consumindo o serviço.

O quadro comparativo da Tabela 6.2 apresenta os modelos conceituais abordados, como foram executados no Projeto e o resultado relativo ao nível de atendimento do modelo, acompanhado das respectivas evidências. A primeira coluna da tabela descreve o que foi executado, ou seja, a implementação do Projeto para atender o modelo abordado. A escala de atendimento indica se os requisitos necessários para a compatibilidade foram cumpridos, acompanhada da referência comprobatória no texto. Com base no quadro comparativo, é possível dizer que os modelos formais para definição e desenvolvimento de Social Machines foram validados com sucesso.

Como as definições formais foram quase totalmente atendidas, as diretrizes aqui descritas representam um resumo do modelo arquitetural para o desenvolvimento de social machines. Aplicá-lo a outro domínio envolveria apenas uma nova análise ontológica para descobrir os termos a serem utilizados na busca, mas a mudança dos termos não deve afetar fundamentalmente as etapas de desenvolvimento e a arquitetura de comunicação.

6.2 V_2. Comparação com modelos e com os relatórios do SUSP

A regra de validação V_2 diz respeito à hipótese H_2: se existir um modelo arquitetural para o desenvolvimento de *social machines*, o mesmo pode ser usado na implementação de um sistema que sirva para identificar atividades criminais. Como discorrido ao longo do capítulo 3 no que tange às estratégias de extração de informações em dados de redes sociais, as próprias características da comunicação e natureza dos dados sugerem ser difícil a identificação da atividade como um crime que está ocorrendo no momento. Assim, a informação extraída reflete o sentimento das pessoas, que escolhem voluntariamente falar sobre os tipos de crime identificados.

Camada	SoMar	Similaridades
L1	Controller	• Módulo LBSociam implementado como um gestor de relacionamentos executando as seguintes integrações: – Twitter *Search API*; – Google Maps *Geocoding API*; • Execução da Modelagem de Tópicos utilizando modelo LDA disponível na base de treinamentos; • Disponibilização de um serviço de consulta (*Query Service*) pela API descrita na tabela 4.4.
L2	Model	• Modelagem de tópicos dos eventos extraídos pela SRL utilizando LDA, conforme previsto na interface de encapsulamento do SoMar; • Gestor de persistência flexível que permite a inserção de quantidade variável de metadados; • Utilização da API REST para inserção de dados.
L3	View	• Construção de interface WWW para visualização das informações; • Disponibilização de um serviço de consulta (*Query Service*) pela API descrita na tabela 4.4;

Tabela 6.1: Quadro comparativo com o modelo MVC do SoMar [17]

É preciso então determinar até que ponto os dados obtidos refletem o sentimento dos usuários nas redes sociais. A análise dos modelos se inicia pelo cálculo de acurácia das técnicas apresentadas com os dados obtidos na pesquisa. O procedimento é similar ao proposto em [3]: extrair uma quantidade definida de dados, aplicar o modelo de classificação e verificar manualmente os resultados.

Executado	**Modelo abordado**	**Fonte**	**Resultado**	**Evidência**
Pré-experimento	Diretrizes de Planejamento	EBSE [44]	Totalmente atendido	Solução teórica da seção 4.1
Modelo de Operação	Definição de Social Machine	Modelo Simplificado [16]	Totalmente atendido	Tabela 4.4
Planejamento da Execução	Diretrizes de Desenvolvimento	SoMar [17]	Parcialmente atendido	Resultado R_1 descrito na seção 5.1
Arquitetura em três camadas	Modelo MVC	SoMar [17]	Totalmente atendido	Resultado R_1 descrito na seção 5.1

Tabela 6.2: Quadro comparativo para validação dos modelos formais

O primeiro dado cuja performance pode ser comparada é a identificação de eventos utilizando SRL. O quadro comparativo da Tabela 6.3 foi construído com dados da primeira coleta, realizada entre os dias 23/11/2014 e 12/12/2014 e descrita na seção 5.2. Os dados apontam para uma performance similar a reportada pelo módulo, tornando possível dizer que a identificação dos eventos aconteceu de maneira satisfatória.

Positivos verdadeiros	**Falsos negativos**	**Recall**	**Fonte**	**Diferença**
26634	7363	0,78	0,71	+0,07
Total de tweets obtidos: **33997**				

Tabela 6.3: Performance da primeira coleta

Para construir o modelo de identificação de atividades criminais é preciso construir um modelo LDA bem treinado, onde os termos da taxonomia "fluem" naturalmente para o topo dos tópicos, ou seja, são os mais prováveis em cada um. Definir a quantidade correta de tópicos é, por si só, um desafio complexo, uma vez que o modelo matemático depende do número fornecido. A figura 6.2 mostra um recorte da interface que contém as distribuições de probabilidade para os tópicos do modelo LDA que constitui a base de treinamento. A representação de cores visa a diferenciação dos elementos além de trazer, por padrão, sempre os dez termos mais frequentes. Da figura é possível observar que o modelo cumpre o objetivo de manter os termos mais importantes da taxonomia como os mais prováveis.

Após a execução dos procedimentos descritos na seção 5.2 para definição dos melhores termos e número de tópicos, foi possível obter os dados da Tabela 6.4, definindo um índice de *precision* de **0,83** para atividades criminais. O cálculo levou em consideração apenas o conjunto de dados da segunda coleta classificados manualmente.

Em relação aos modelos de identificação de atividades criminais citados na pesquisa, tratam-se de tentativas de estabelecer padrões preditivos capazes de identificar futuras

Figura 6.2: Representação da interface de identificação dos tópicos

Positivos verdadeiros	**Falsos positivos**	**Precision**
792	158	0,83
Total de tweets considerados: **958**		

Tabela 6.4: Acurácia na identificação de atividades criminais

ocorrências. A metodologia de pesquisa não estabelece a necessidade de construção de modelos probabilísticos que envolvam o futuro. Encontrar um valor de acurácia elevado também não é o objetivo principal do trabalho, sendo importante apenas para saber se os resultados podem ser considerados. Identificar a validade dos dados para posterior disponibilização na interface agrega mais valor, acreditando que a computação social pode melhorar a qualidade da informação ao permitir que os usuários tirem suas próprias conclusões da observação.

Contudo, uma pergunta surge naturalmente ao realizar a análise dos dados: estão eles próximos da realidade? Ou seja, se as pessoas escolhem voluntariamente falar sobre crimes seria equivalente a dizer que mais crimes estão ocorrendo? Alguns autores levantam problemas relativos à confiabilidade [30] ao tentar substituir dados oficiais por uma aplicação com foco no usuário. Por esse motivo a escolha da pesquisa é identificar e mapear a sensação de insegurança no país, partindo da hipótese que a ocorrência de um evento criminal gera repercussão nas redes sociais.

O primeiro passo é a identificação das áreas de comparação e seleção dos dados. Para realizar a análise foi selecionada uma amostra contendo os *tweets* coletados e classificados entre os dias 12 de maio de 2015 e 13 de junho de 2015, totalizando **42704** status. O gráfico da Figura 6.3 apresenta a distribuição de ocorrências por estado brasileiro e categoria, construindo visão similar à fornecida pelos dados oficiais do Anuário Brasileiro de Segurança Pública [33] nas categorias selecionadas.

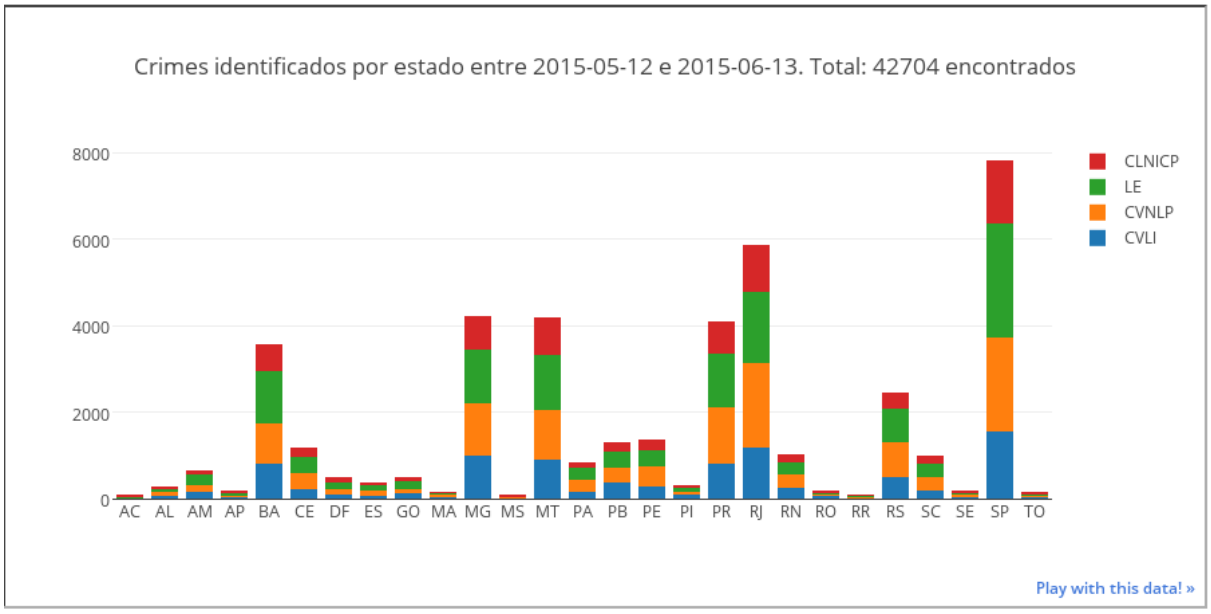


Figura 6.3: Distribuição de eventos por estado e categoria

Para efeito de comparação selecionamos os dois estados com o maior número de ocorrências: São Paulo e Rio de Janeiro. Os dados oficiais do Anuário estão apresentados na Tabela 6.5, utilizando os termos da taxonomia reduzida descrita na seção 4.3.2 e identificados na coluna categoria. A taxa se refere ao número de ocorrências a cada 100.000 habitantes, índice bastante utilizado pelo anuário para fazer comparação entre estados com tamanhos populacionais diferentes. A coluna ocorrências traz o total de ocorrências registradas no ano de 2013. Para a categoria LE foram considerados somente ocorrências dos crimes de tráfico de drogas, enquanto na categoria CNLNICP somente lesão corporal dolosa foi contabilizada. Como os dados são referentes ao ano, a coluna (1/12) apresenta o valor total dividido pelos 12 meses do ano para facilitar a comparação com o período coletado que foi de um mês.

Estado	**Categoria**	**Taxa**	**Ocorrências**	**Ocorrências (1/12)**
SP	CVLI	11,7	5119	427
	CVNLP	812,9	355792	29649
	LE	99,5	43556	3630
	CNLNICP	394,5	172665	14389
RJ	CVLI	30,1	4928	411
	CVNLP	768,6	126045	10504
	LE	79,1	12976	1081
	CNLNICP	461,3	75642	6304

Tabela 6.5: Dados selecionados do Anuário de Segurança Pública [33]

Na extração realizada foram obtidos os dados da Tabela 6.6, contendo o total de eventos identificados por categoria. Como não temos acesso aos números referentes à

população total do Twitter por estado brasileiro, não é possível obter um equivalente da taxa por habitantes como apresentada no Anuário.

Estado	**Categoria**	**Ocorrências**
SP	CVLI	1563
	CVNLP	2160
	LE	2630
	CNLNICP	1477
RJ	CVLI	4928
	CVNLP	126045
	LE	12976
	CNLNICP	75642

Tabela 6.6: Dados coletados

Conforme o esperado, os dados estão em diferentes escalas de grandeza e não podem ser diretamente comparados. Porém, é possível adotar uma outra abordagem, que está apresentada na Tabela 6.7: existe uma relação entre a ocorrência dos crimes e o que as pessoas falam? A comparação considera o percentual de ocorrências nas categorias selecionadas. O percentual extraído indica o percentual na categorial em relação o total de *tweets* do estado. Já a coluna percentual oficial adota uma abordagem similar, considerando o percentual de ocorrências em relação ao total de ocorrências consideradas.

Estado	**Categoria**	**Percentual extraído**	**Percentual oficial**
SP	CVLI	19,96%	0,89%
	CVNLP	27,59%	61,65%
	LE	33,59%	7,55%
	CNLNICP	18,86%	29,92%
RJ	CVLI	20,03%	2,24%
	CVNLP	33,76%	57,40%
	LE	27,98%	5,91%
	CNLNICP	18,24%	34,45%

Tabela 6.7: Comparativo entre os dados extraídos e as ocorrências

Embora os dados reforcem a tese de que não é possível utilizar os dados da pesquisa como fonte de comparação para ocorrências criminais, o objetivo principal de mapear a percepção foi cumprido com sucesso. Em ambos os estados, chama atenção a diferença proporcional entre o total de ocorrências e o percentual de *tweets* identificados na categoria CVLI. É de imaginar que um crime mais violento aumente a sensação de insegurança da população, reforçando a ocorrência de comentários nas redes sociais. Assim, pode-se inferir que a ocorrência de homicídios reforça a sensação de insegurança da população.

Capítulo 7

Considerações Finais

Foi possível visualizar ao longo da pesquisa como a evolução da Web tem criado um conjunto de serviços que podem ou não ser conectados, e será cada vez mais relevante desenvolver tecnologias de interação, principalmente na Web de dados. O desenvolvimento de uma *social machine* no domínio de violência e criminalidade tem o objetivo de responder a seguinte sentença: seriam tais tecnologias capazes de ajudar o cidadão comum?

O problema é abordado em duas diferentes visões: cidadãos comuns, produzindo e consumindo dados sociais, e governos, elaborando leis e políticas públicas para atender os anseios dos cidadãos. Redes sociais são capazes de fornecer uma visão restrita à vizinhança dos usuários, normalmente amigos próximos e conhecidos que leem a *timeline* do Twitter ou compartilham atividade através do facebook. Contudo, cada canal de comunicação possui um limite para o número de nós conectados que a mensagem é capaz de alcançar. A construção de uma tecnologia integrada está diretamente relacionada à capacidade de encontrar "nós latentes", ou dados que não poderiam ser coletados através dos canais oficiais.

Diferente dos outros exemplos que trabalham com dados estáticos coletados, a proposta é acompanhar as tendências à medida que "aparecem" nas redes sociais. Dentre os desafios superados durante a pesquisa é possível destacar a construção de uma arquitetura integrada, delimitação do referencial teórico necessário e elaboração de protótipo de interface para visualização das informações. Considerando a necessidade de validação empírica, esse estudo de viabilidade é parte do processo em andamento para definição de um arcabouço prático e teórico em *social machines*.

7.1 Contribuições

O principal objetivo da pesquisa, como apresentado na seção 1.2, é investigar um modelo arquitetural para o desenvolvimento de uma *social machine* no domínio de violência e

criminalidade que utilize dados oriundos de redes sociais. Para atendê-lo é construído o resultado descrito na seção 4, organizado em um modelo de três camadas e quatro passos. A solução teórica apresentada se baseia na EBSE no que tange às diretrizes de desenvolvimento e organização da implementação. Assim, é preciso definir até que ponto é possível generalizar as descobertas para fora do escopo da pesquisa.

O foco da validação do objetivo da proposta se deu em dois diferentes aspectos: na definição de *social machines* como uma tecnologia social e no escopo de sistemas de aquisição do conhecimento. O objetivo foi mostrar que havia similaridades suficientes em ambos os arcabouços teóricos, permitindo situar em definições tanto de um quanto de outro. Assim, é possível dizer que *social machines* são sistemas de aquisição de conhecimento que utilizam computação social, e podem ser descritos em diferentes modelos. Para exemplificar a sobreposição de conceitos arquiteturais, a Figura 7.1 apresenta a *social machine* desenvolvida na pesquisa descrita através do modelo MVC proposto no estilo arquitetural SoMar.

Como exposto ao longo do texto, o objetivo não é construir um *framework* genérico que englobe todos os aspectos envolvidos no processo de desenvolvimento de *social machines*. Contudo, uma abordagem mais completa e mais genérica é abordada na descrição do estilo arquitetural SoMar. Ao determinar a compatibilidade de ambas as propostas é possível definir o procedimento aqui descrito como uma aplicação do modelo arquitetural a um protótipo no domínio de violência e criminalidade. Trata-se ainda de uma extensão do modelo, na medida em que introduz um módulo Gestor de Relacionamentos como um banco de dados com foco em metadados.

Também é possível afirmar, com base nos dados obtidos na pesquisa, que a *social machine* é capaz identificar atividades relacionadas a violência e criminalidade com acurácia conhecida. Como o protótipo instancia e estende um modelo arquitetural conhecido, identifica eventos relacionados à criminalidade utilizando dados fornecidos pelos próprios usuários, e ainda trata de um problema relevante para a sociedade, é possível afirmar que o objetivo foi atingido com sucesso.

Durante o processo de desenvolvimento foram obtidos ainda alguns subprodutos, identificados nos objetivos específicos a seguir:

G_1 Construir um protótipo funcional de *social machine* para identificar atividade criminal em dados oriundos de redes sociais, desenvolvendo um banco de dados como uma aplicação de dados abertos;

G_2 Validar a aplicação da análise semântica em dados de redes sociais e a utilização de *crowdsourcing* para a identificação de atividades criminais.

As próximas seções discutem os objetivos específicos atingidos na pesquisa.

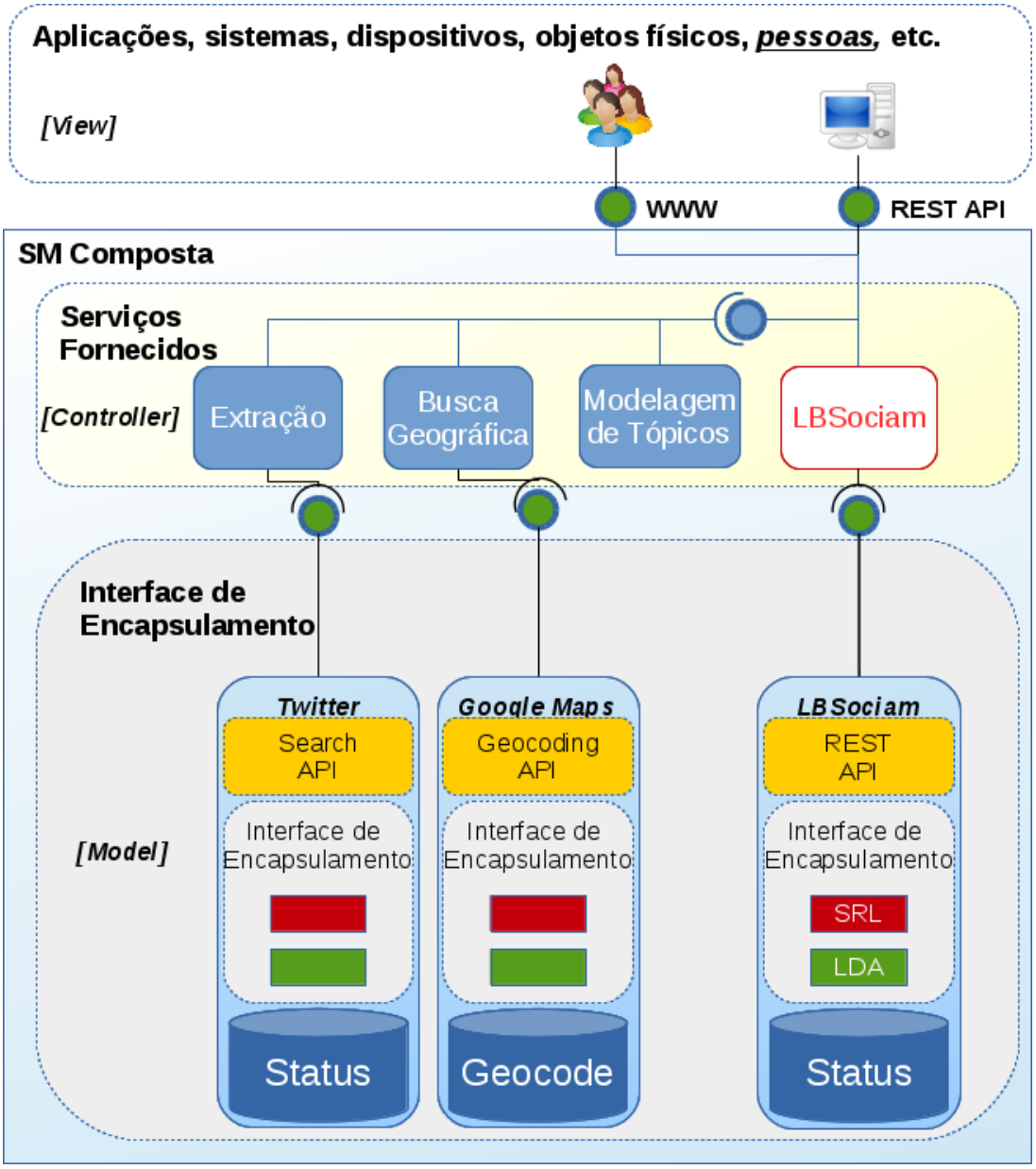


Figura 7.1: Modelo SoMar alterado para o LBSociam

7.1.1 G_1. Protótipo funcional de Social Machine

O desenvolvimento do módulo LBSociam descrito na seção 5.1 traz uma importante contribuição no sentido de estender o estilo arquitetural SoMar, incluindo um componente de armazenamento das informações oriundas de diferentes interfaces. Ao fornecer uma API única de comunicação com as camadas superiores e permitir visualização tanto dos dados brutos quanto disponibilizar um protótipo de interface, constrói-se uma base tangível para integração de diferentes tipos de *social machine*. É possível estender a aplicação para outros domínios, substituindo a camada de extração das informações, da mesma maneira

que permite ainda a construção de diferentes camadas de visualização.

Ainda em comparação com os trabalhos relacionados descritos na seção 2.4, o módulo LBSociam traz uma grande diferença: é o único que permite interagir sobre os dados oriundos de diferentes origens, fornecendo uma camada extra e novas possibilidades de *crowdsourcing*. Ao interagir com os dados consolidados em diferentes fontes, agrega valor ao permitir o desenvolvimento de interfaces de *feedback* por parte dos usuários em uma operação simplificada, como descrito na Tabela 4.4.

A API REST traz ainda integração com padrões e formatos populares junto aos desenvolvedores de dados abertos, como formato JSON, documentação *Swagger* e operações REST. Permite ainda a inclusão de novos metadados, com o diferencial de não trabalhar com um modelo totalmente agnóstico, facilitando a análise e integração com outras aplicações.

Por se tratar de uma dissertação de Mestrado Profissional em Computação Aplicada, trata-se ainda de um produto que agrega bastante valor à empresa Lightbase. Pode tanto ser disponibilizado como um serviço, principalmente para clientes governamentais, como demonstra a aplicação do banco de dados para dados de redes sociais. Como o escopo de atuação da empresa tem foco no desenvolvimento de sistemas de protocolo eletrônico, o sistema desenvolvido expande o protocolo de serviços ao tratamento de dados de redes sociais.

A descrição do processo de desenvolvimento para o domínio de violência e criminalidade não apresentou detalhes que foram identificados como específicos do domínio. Ainda que o modelo SRL+LDA descrito no capítulo 3 possua maior respaldo na identificação de atividades criminais, não há nada na definição dos algoritmos que impeça a utilização em outros domínios. Se levarmos em consideração que o objetivo do procedimento é preservar um certo nível de abstração em relação aos detalhes de desenvolvimento [17], basta utilizar outro modelo de extração na mesma camada e o resultado permanece o mesmo.

7.1.2 G_2. Validar a utilização de análise semântica e crowdsourcing em dados de redes sociais para identificar atividade criminal

O objetivo G_2 traz a necessidade de validação dos modelos de análise semântica na identificação de atividade criminal, cuja discussão e análise está descrita na seção 5.2. A técnica de extração de entidades é o primeiro princípio a ser observado em relação ao atendimento dos objetivos. Os quadros das Tabelas 5.2 e 5.4 mostram que os resultados obtidos estão próximos dos que são estimados para os softwares utilizados. Assim, é pos-

sível afirmar que a extração de entidades foi realizada com sucesso para identificação de eventos relacionados à violência e criminalidade.

Já a identificação dos termos da taxonomia, cuja taxa de *precision* está apresentada na Tabela 5.5, também está dentro do que se espera com base nas mesmas taxas dos softwares utilizados como base. Não é possível afirmar, com um conjunto tão pequeno de dados, que os termos considerados são válidos em qualquer cenário, mas a observação da performance mostra uma diferença clara em relação a outros conjuntos de termos testados. Pode-se concluir então que o conjunto de termos selecionado é melhor do que outros testados na pesquisa, imaginando que a validação se dará na observação da distribuição de probabilidade para os termos dentro dos tópicos.

Na análise manual da taxa de *precision* para a base de treinamento, o valor obtido foi de **0,83** ou **83%** de sucesso na identificação de crimes. Comparando com a performance estimada em modelos LDA [74] e identificação de eventos utilizando SRL [35], é possível considerar os valores aceitáveis dentro do que se propõe. Uma importante ressalva é relativa à impossibilidade de comparação entre os dados obtidos e os modelos preditivos descritos na seção 3.2.1. Como o objetivo da pesquisa não é prever o futuro, e sim identificar tendências atuais, não foi possível realizar uma comparação direta com os resultados dos modelos. Ainda assim, levando em consideração somente a distribuição de probabilidade relativa aos tópicos, as performances são similares, ainda que considerando que o número de tópicos aqui definido é significativamente menor.

Um outro importante modelo a ser considerado diz respeito à utilização de *crowdsourcing* para identificação de atividades criminais. Em vários momentos do texto é apresentada uma diferença clara entre o sistema de registro de ocorrências criminais e a proposta de pesquisa, destacando-se a análise dos dados criminais na seção 3.5. Assim, é possível imaginar que os dados apresentados não tratam somente de crimes que estejam ocorrendo no momento da coleta. O objetivo é captar a sensação de insegurança, imaginando que nos locais onde as pessoas falam mais sobre crimes, existe de fato uma maior ocorrência de atividades criminais.

A descoberta de tal informação só é possível se os usuários estiverem efetivamente escrevendo sobre crimes nos seus *tweets*. A definição de *crowdsourcing* apresentada na seção 2.3.1 delimita a necessidade de compartilhamento de informações relevantes entre os usuários nas redes sociais. Já a definição de Computação Social apresentada na mesma seção aponta a necessidade de inserir novamente em uma *social machine* a colaboração realizada voluntariamente. A seção 5.2 afirma que o desenvolvimento de uma *social machine* que contenha dados oriundos de redes sociais é suficiente para concluir que o *crowdsourcing* da informação foi realizado, com a ressalva de que é necessário provar se o dado coletado se refere de fato a atividades criminais.

Tal afirmação pode ser melhor entendida pelo exemplo da Figura 7.2, que trata de um crime que chocou a cidade do Rio de Janeiro. Os usuários, dadas as circunstâncias violentas da situação, passaram algum tempo falando sobre o assunto, aumentando a sensação de insegurança da população. A Figura 7.3 apresenta um exemplo onde a insatisfação apareceu nas coletas executadas, e foi corretamente identificada como na categoria CVNLP, ou assalto. O usuário revela um sentimento crescente de insatisfação, ao afirmar que "virou moda" esfaquear as pessoas no Rio de Janeiro.

Figura 7.2: *Tweets* falando sobre um crime bárbaro ocorrido na cidade do Rio de Janeiro

Coletar tal insatisfação é o que foi mencionado há pouco como "identificação de nós latentes". Não se trata de um usuário que vai à delegacia realizar um registro de ocorrência, mas certamente é alguém que entende que a cidade está mais insegura.

Figura 7.3: Status coletado na interface da *social machine*

7.2 Restrições

Ainda que a tecnologia seja capaz de reduzir a lacuna de informações em violência e criminalidade, não é uma tarefa simples provar que outras demandas podem ser atendidas. É necessário, além de realizar um estudo de taxonomia inicial para o domínio desejado, expandir o escopo do desenvolvimento na identificação de outros problemas, com o objetivo de considerar a solução aqui apresentada como um *framework* genérico para o desenvolvimento de social machines.

A pergunta de pesquisa sugere ainda a necessidade de atendimento de demandas sociais utilizando tecnologias Web modernas, supondo que sejam úteis aos cidadãos. A utilidade é um conceito abstrato, mas tecnologias Web possuem diferentes formas de medir a aceitação da sociedade através de estatísticas de visualização e acesso. O aplicativo Onde Fui Roubado [1] foi considerado bem sucedido por causa do grande impacto que teve na mídia: as maiores agências de notícia do Brasil apresentaram diferentes níveis de interesse na ideia original de dois estudantes universitários. Contudo, em São Paulo, os dados do site do Projeto apresentam um total de **8395** relatos de crime [58][2]. Dados oficiais do ano de 2013 [32] mostram, somente na categoria assalto [3], uma taxa de **568,6** roubos por cem mil habitantes, totalizando aproximadamente **68000** ocorrências na cidade de São Paulo [4]. É possível perceber que os dados estão em ordens de grandeza diferentes, não

[1] `http://www.ondefuiroubado.com.br/`

[2] Dados mais atuais estão disponíveis no endereço `http://www.ondefuiroubado.com.br/sao-paulo/SP/estatisticas`

[3] Na taxonomia da Figura 4.5 assalto é uma subcategoria CVNLP – Crimes Violentos Não Letais contra o Patrimônio

[4] Para o cálculo foi considerada a população de 11,89 milhões de habitantes na cidade de São Paulo

permitindo a comparação dos dados oficiais com os apresentados pelo aplicativo.

A implementação da social machine tem um escopo limitado no que se refere às políticas públicas. Como discutido ao longo do texto, não é possível substituir as ferramentas e métodos de coleta de dados oficiais do governo. Tal mudança seria possível apenas com um suporte oficial massivo para evitar todas as restrições apresentadas em [30]. O objetivo é estender as teorias computacionais para permitir novas visões no domínio de violência e criminalidade.

Ainda que não seja considerada uma aplicação oficial, não significa que os governos não possam utilizá-la para melhorar as políticas públicas que já existem. Da coleta nas estações de polícia até atingir o Governo Federal, a informação tem que seguir um caminho longo demais. Enquanto isso, o serviço de dados desenvolvido na pesquisa pode ser utilizado pelas autoridades como uma ferramenta de auxílio em decisões momentâneas, principalmente no que se refere à mobilização online.

Uma outra restrição importante diz respeito ao aspecto econômico da população, que precisa estar conectada à Internet para interagir com as redes sociais. Nos experimentos realizados é possível perceber claramente um viés de concentração dos incidentes nas regiões mais ricas do país, que também não por acaso são melhor atendidas por conexão de Internet. É de se imaginar que resultados interessantes podem ser obtidos ao analisar as principais metrópoles do país, enquanto em outras a massa de dados não é relevante o suficiente para tirar conclusões ou fazer observações pontuais.

Do lado dos cidadãos, a maior parte da distribuição das forças de segurança tem como base dados estatísticos relativos à incidentes criminais em determinada área geográfica. O exemplo do projeto Ushahidi [61] mostra como as pessoas utilizaram as redes sociais para fugir de áreas violentas enquanto aconteciam os protestos. Quando há algum tipo de "onda de violência" se aproximando, os cidadãos podem utilizar a informação para sair do caminho e se proteger.

7.3 Trabalhos futuros

Por se tratar de uma área de estudo em seu estágio inicial, ainda há muito trabalho a ser feito no que tange à validação do modelo e identificação das fronteiras de aplicação. Na seção 2.4 foi possível ver que já existem provas de conceito para pontos específicos do modelo, como a aplicação *WhatHere* [26]. Ao propor uma social machine no domínio de violência e criminalidade [30], os autores apontaram restrições que supõem a execução de um projeto maior para atender também governos e substituir os sistemas tradicionais. Contudo, ainda é necessário identificar outros domínios e verificar se o procedimento desenvolvido durante a pesquisa pode ser facilmente estendido.

No que tange aos modelos de extração e armazenamento, a aplicação por si só é uma importante prova de conceito, uma vez que aplica o banco de dados documental Lightbase como fonte primária de armazenamento das informações. Já os modelos preditivos utilizados para classificar as atividades criminais precisam de um trabalho mais longo de verificação da acurácia. Será que, ao longo do tempo, com um volume realmente significativo de dados, a performance dos modelos se mantém? A introdução do *crowdsourcing* massivo através da disponibilização na Internet teria capacidade para melhorar os resultados? E como se comportaria a arquitetura proposta em tal estrutura? Estes são apenas alguns dos possíveis desdobramentos identificados durante a execução da pesquisa que se referem à infra-estrutura tecnológica desenvolvida.

Há também muito trabalho a ser feito no campo da Interação Humano Computador – IHC –, uma vez que os protótipos desenvolvidos não seguiram padrões relativos à usabilidade e acessibilidade do sistema. Como o próprio movimento de dados abertos vem ensinando ao longo do tempo, não se trata de apenas disponibilizar os dados, e sim de torná-los disponíveis em interfaces que os usuários sejam capazes de entender.

Um outro desdobramento, que tem capacidade para afetar radicalmente a maneira com a qual os cidadãos se relacionam com os serviços públicos no Brasil, seria uma tentativa de trazer o governo para o centro do problema. Como apresentado na seção 3.5, o processo de coleta de estatísticas é caro e demorado, além de muito burocrático para o cidadão comum. E se existisse uma outra forma de realizar o mesmo procedimento, utilizando os mesmos recursos de interface? É possível imaginar o cenário onde, pela simples adição de um metadado ao Twitter, seja capaz de identificar o usuário como uma pessoa verdadeira e permita o registro de uma ocorrência real. Ainda que pareça um cenário impossível a curto prazo, vislumbrar a possibilidade pode abrir os horizontes e diminuir a burocracia no Estado.

Apêndice A

Dicionário dos termos da taxonomia

A seção contém a descrição detalhada dos termos da taxonomia da Figura 4.5.

- Crimes

  -> NT1. CNL Crimes não Letais

  -> NT2. CL (Crimes Letais)

- T7. LE

  Leis Especiais

  -> NT1. CNL Crimes não Letais

  - ET

    Entorpecentes - Tráfico
  - EPU

    Entorpecentes - Posse e Uso
  - PIAF

    Porte Ilegal de Arma de Fogo

- T9. CNLNICP

  Crimes Não Letais Intencionais Contra a Pessoa

  -> NT1. CNL Crimes não Letais

  - TH

    Tentativa de Homicídio
  - LCD

    Lesão Corporal Dolosa

  - OLCD

    Outras Lesões Corporais Dolosas
  - LCCT

    Lesão Corporal Culposa de Trânsito
  - OCRLC

    Outros Crimes Resultantes em Lesão Corporal

- T5. OCL

  (Outros Crimes Letais)

  -> NT2. CL (Crimes Letais)

  - MAT

    Mortes Acidentais no Trânsito
  - S

    Suicídio
  - OMA

    Outras Mortes Acidentais
  - OHC

    Outros Homicídios Culposos
  - OCRM

    Outros Crimes Resultantes em Morte
  - HCT

    Homicídio Culposo de Trânsito
  - MAE

    Mortes a Esclarecer

- T4. CVLI

  (Crimes Violentos Letais Intencionais)

  -> NT2. CL (Crimes Letais)

- T3. CLI

  (Crimes Letais Intencionais)

  -> T4. CVLI (Crimes Violentos Letais Intencionais)

  - LCSM

    Lesão Corporal Seguida de Morte
  - LT

    Latrocínio

- T8. CCLS

  Crimes Contra a Liberdade Sexual

  -> NT1. CNL Crimes não Letais

  - E

    Estupro
  - TE

    Tentativa de Estupro

- T2. HD

  (Homicídios Dolosos)

  -> T4. CVLI (Crimes Violentos Letais Intencionais)

- T6. CVNLCP

  Crimes Violentos Não Letais Contra o Patrimônio

  -> NT1. CNL Crimes não Letais

  - R

    Roubo
  - RV

    Roubo de Veículo
  - RC

    Roubo de Carga
  - RIF

    Roubo a Instituição Financeira

- NT1. CNL

  Crimes não Letais

Descrição gerada automaticamente através do serviço `https://bubbl.us/`

Apêndice B

Telas do Sistema

A seção apresenta as telas do protótipo funcional desenvolvido.

Figura B.1: Página Inicial

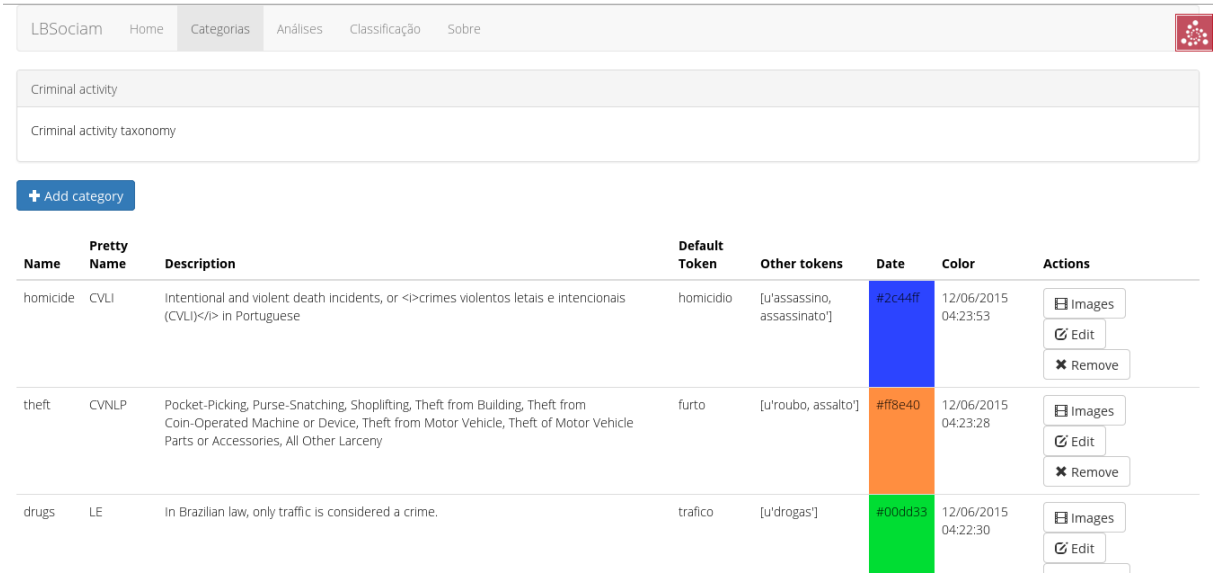


Figura B.2: Lista de termos da taxonomia

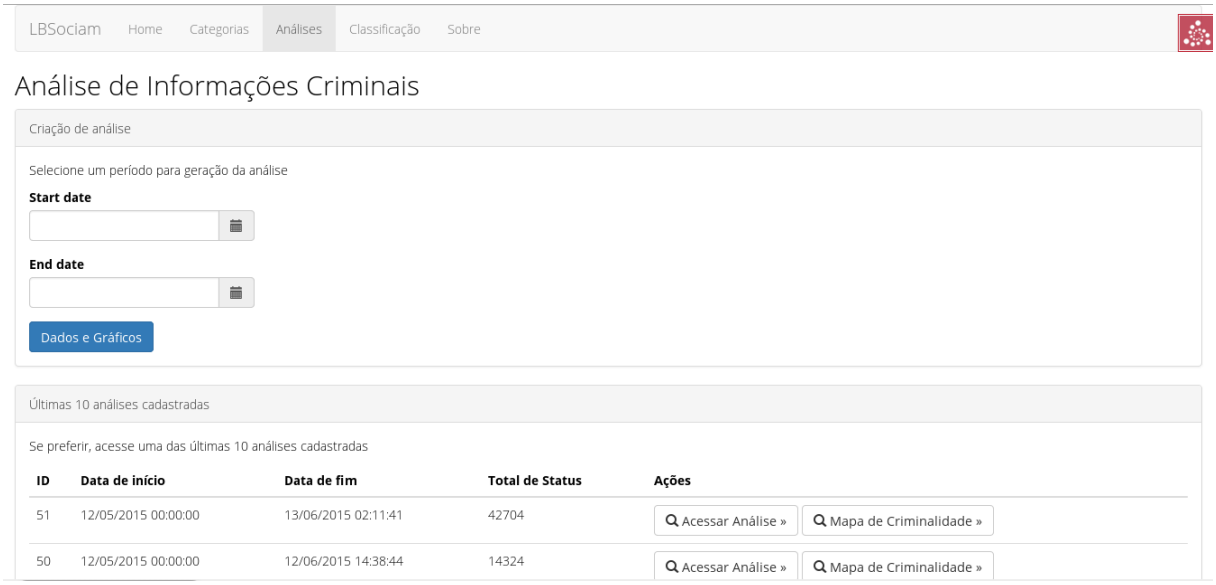


Figura B.3: Lista de análises disponíveis

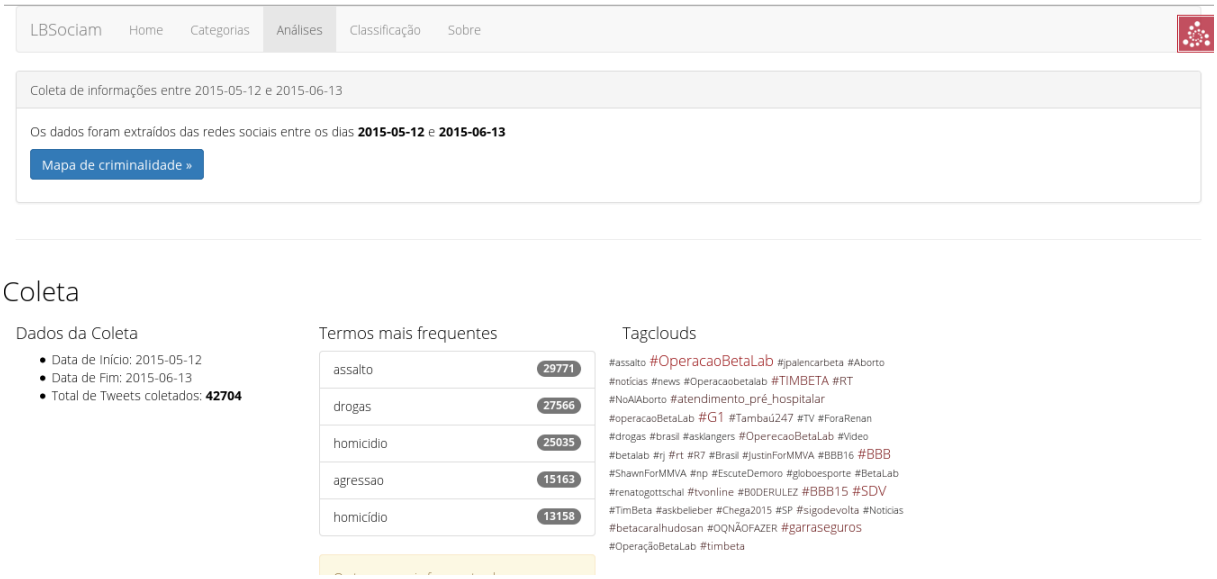


Figura B.4: Página de resumo da análise

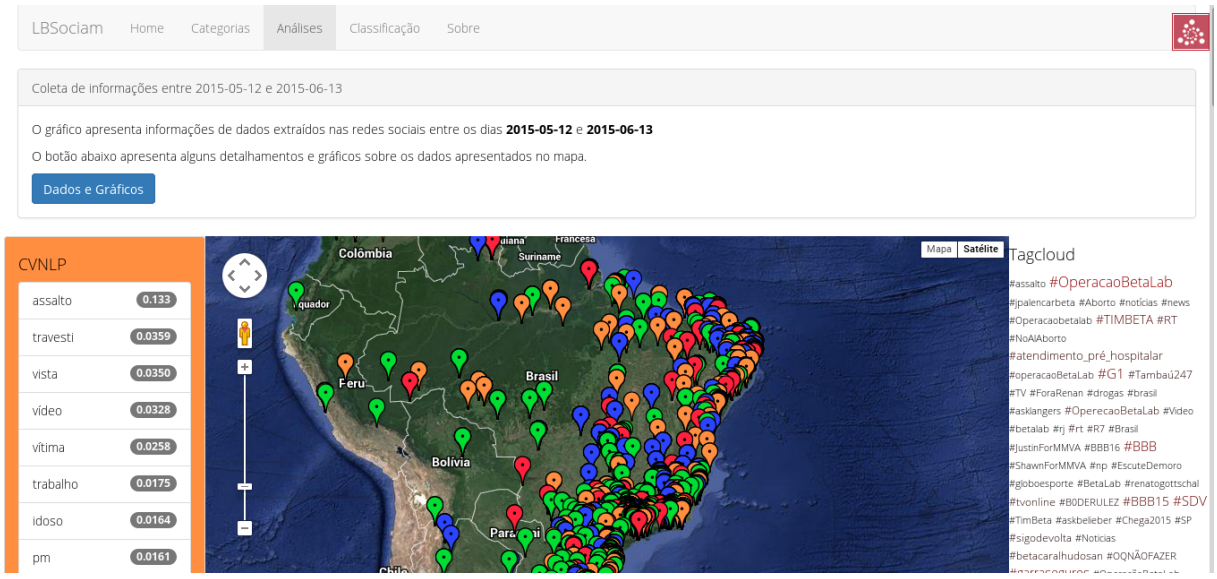


Figura B.5: Mapa de criminalidade

Figura B.6: Apresentação do status

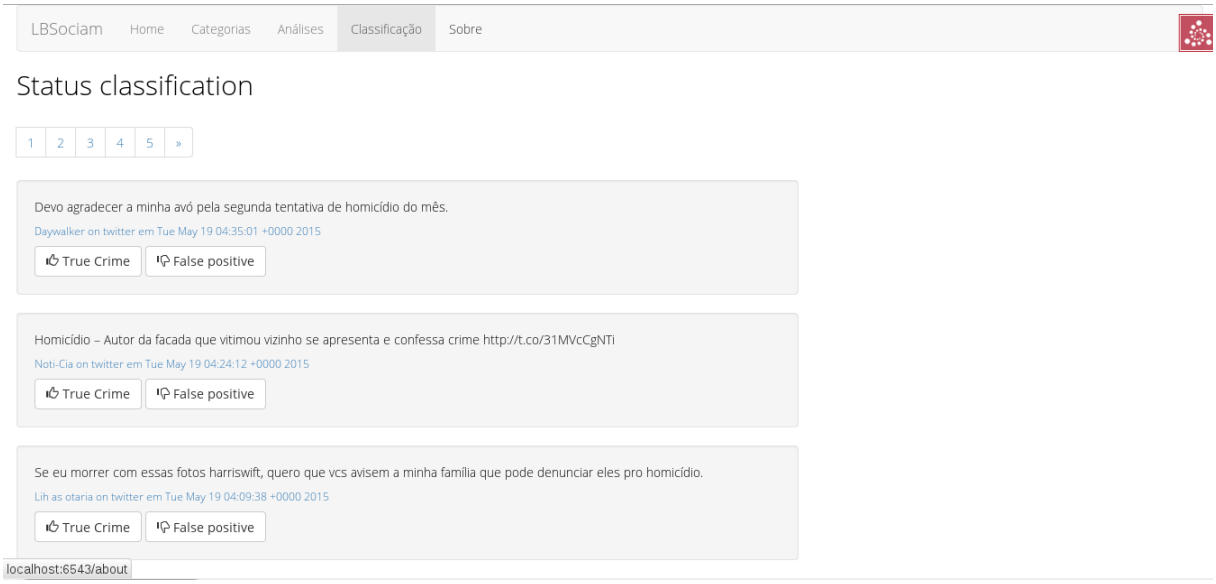


Figura B.7: Página de classificação de status

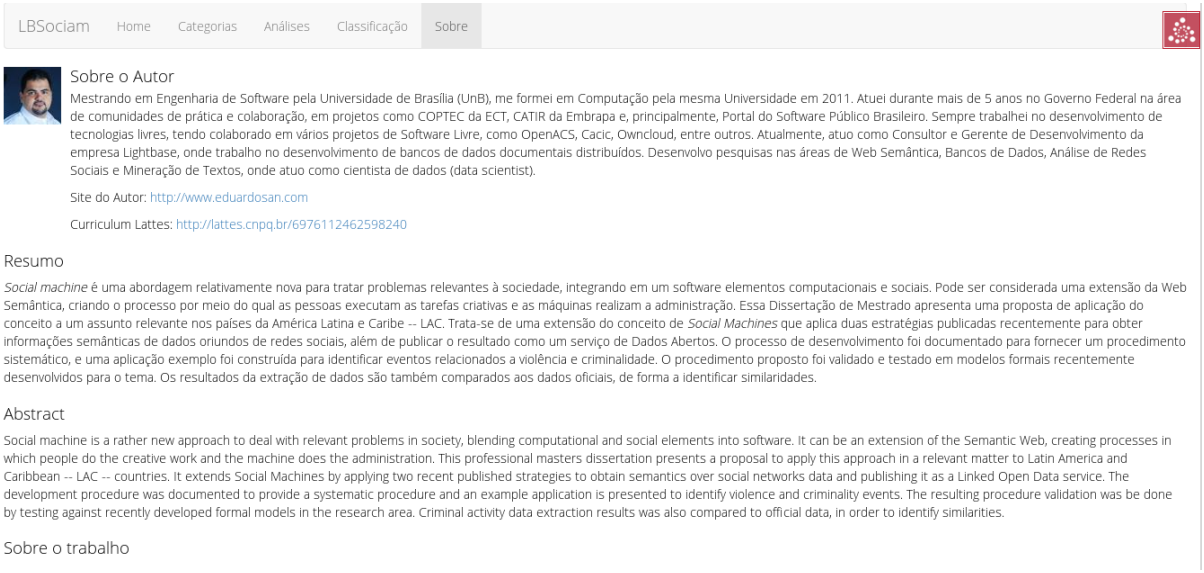


Figura B.8: Sobre o autor

Apêndice C

Fontes

Todos os fontes utilizados no desenvolvimento do sistema podem ser encontrados no github:

- Motor de extração e armazenamento LBSociam: `https://github.com/lightbase/LBSociam`
- Interface de visualização das informações lbsociamgame: `https://github.com/lightbase/lbsociamgame`
- Módulo de descoberta de informações geográficas LBGeo: `https://github.com/lightbase/LBGeo`

O projeto pode ser acessado no endereço `http://lbsociam.eduardosan.com`

Referências

[1] Paolo Atzeni, Francesca Bugiotti, and Luca Rossi. Uniform access to nosql systems. *Information Systems*, 43(0):117 – 133, 2014. URL: `http://www.sciencedirect.com/science/article/pii/S0306437913000719`, `doi:10.1016/j.is.2013.05.002`. 61

[2] Ricardo Baeza-Yates, Berthier Ribeiro-Neto, et al. *Modern information retrieval*, volume 463. ACM press New York, 1999. 27

[3] Marco Bastos, Raquel Recuero, and Gabriela Zago. Taking tweets to the streets: A spatial analysis of the vinegar protests in brazil. *First Monday*, 19(3), 2014. URL: `http://firstmonday.org/ojs/index.php/fm/article/view/5227`. 6, 16, 43, 85

[4] Adam Bermingham and Alan F Smeaton. On using twitter to monitor political sentiment and predict election results. *SAAIP*, page 2, 2011. 3

[5] Tim Berners-Lee. Linked data, 2006. URL: `http://www.w3.org/DesignIssues/LinkedData.html`. 11, 14

[6] Tim Berners-Lee and Robert Cailliau. Worldwideweb: Proposal for a hypertext project. *Retrieved on February*, 26:2008, 1990. URL: `http://www.w3.org/Proposal`. 10

[7] Tim Berners-Lee, Mark Fischetti, and Michael L Foreword By-Dertouzos. *Weaving the Web: The original design and ultimate destiny of the World Wide Web by its inventor*. HarperInformation, 2000. 16

[8] Tim Berners-Lee, James Hendler, Ora Lassila, et al. The semantic web. *Scientific American*, 2001. URL: `http://www.scientificamerican.com/article/the-semantic-web/`. 11, 15

[9] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python*. "O'Reilly Media, Inc.", 2009. 23, 32

[10] Christian Bizer, Tom Heath, and Tim Berners-Lee. Linked data-the story so far. *IJSWIS*, 5(3):1–22, 2009. 11, 12, 14

[11] David M Blei. Probabilistic topic models. *Communications of the ACM*, 55(4):77–84, 2012. 27, 28, 46

[12] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003. 28

[13] BRASIL. Crime records map, 2009. URL: http://migre.me/kHPO5. x, 34, 35, 37

[14] Rob Brennan, Kevin C Feeney, and Odhrán Gavin. Publishing social sciences datasets as linked data: a political violence case study. *ENRICH 2013 Conference Proceedings*, 2013. 4, 20, 31, 50

[15] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1):107–117, 1998. 11, 27

[16] Vanilson Burégio, Silvio Meira, and Nelson Rosa. Social machines: a unified paradigm to describe social web-oriented systems. In *Proceedings of the 22nd international conference on World Wide Web companion*, pages 885–890. International World Wide Web Conferences Steering Committee, 2013. 3, 4, 6, 8, 17, 19, 81, 82, 86

[17] Vanilson André de Arruda Buregio. *Social Machines: A Unified Paradigm to Describe, Design and Implement Emerging Social Systems*. PhD thesis, UFPE, 2014. xi, xii, 16, 18, 81, 82, 83, 85, 86, 93

[18] Antony Gonçalves Carvalho. Interface NoSQL integrada a banco relacional para gerenciamento de dados em nuvem privada. Technical report, UniCEUB, Março 2015. URL: http://hdl.handle.net/235/5939. x, 62, 63

[19] Hsinchun Chen, Wingyan Chung, Jennifer Jie Xu, Gang Wang, Yi Qin, and Michael Chau. Crime data mining: a general framework and some examples. *Computer*, 37(4):50–56, 2004. 3, 33, 76

[20] Tony et all Coates. URIs, URLs, and URNs: Clarifications and recommendations 1.0. Technical report, W3C, September 2001. URL: http://www.w3.org/TR/uri-clarification/. 12

[21] D. et all Crockford. The application/json media type for javascript object notation (JSON). Technical report, W3C, July 2006. URL: http://tools.ietf.org/rfc/rfc4627.txt. 50, 51

[22] Richard Cyganiak and Anja Jentzsch. Linking open data cloud diagram. *LOD Community (http://lod-cloud. net/)*, 2011. 12

[23] DBpedia. DBpedia place definition, 2014. URL: http://dbpedia.org/ontology/Place. 50

[24] Daniel Dietrich, J Gray, T McNamara, A Poikola, P Pollock, J Tait, and T Zijlstra. Open data handbook, 2009. 4, 8, 13, 14

[25] Anhai Doan, Raghu Ramakrishnan, and Alon Y. Halevy. Crowdsourcing systems on the world-wide web. *Commun. ACM*, 54(4):86–96, April 2011. URL: http://doi.acm.org/10.1145/1924421.1924442, doi:10.1145/1924421.1924442. 3, 6, 8, 75

[26] Kellyton dos Santos Brito, Lenin Ernesto Abadie Otero, Patrícia Fontinele Muniz, Leandro Marques Nascimento, Vanilson André de Arruda Burégio, Vinicius Cardoso Garcia, and Silvio Romero de Lemos Meira. Implementing web applications as social machines composition: A case study. In *SEKE*, pages 311–314, 2012. 4, 8, 16, 17, 18, 19, 81, 97

[27] Magali Sanches Duran and Sandra Maria Aluísio. Propbank-br: a brazilian treebank annotated with semantic role labels. In *LREC*, pages 1862–1867, 2012. 25

[28] Marcelo Ottoni Durante. Avanços e desafios na implantação do sistema nacional de estatísticas de segurança pública e justiça criminal (SINESPJC). *Anuário de Segurança Pública*, 2008. 2

[29] EUCLID. Chapter 1: Introduction and application scenarios, 2014. URL: `http://www.euclid-project.eu/modules/chapter1`. x, 44

[30] Maire Byrne Evans, Kieron O'Hara, Thanassis Tiropanis, and Craig Webber. Crime applications and social machines: crowdsourcing sensitive data. In *SOCIAM: The Theory and Practice of Social Machines*, May 2013. URL: `http://eprints.soton.ac.uk/351275/`. 2, 3, 4, 19, 36, 87, 97

[31] FBI. National incident-based reporting system (NIBRS) – General Information, 2014. URL: `http://www2.fbi.gov/ucr/faqs.htm`. 33, 36, 49

[32] FBSP. Segurança pública em números. *Anuário Brasileiro de Segurança Pública 2013*, 2013. 96

[33] FBSP. Segurança pública em números. *Anuário Brasileiro de Segurança Pública 2014*, 2014. xii, 36, 47, 49, 87, 88

[34] Lee Feigenbaum, Ivan Herman, Tonya Hongsermeier, Eric Neumann, and Susie Stephens. The semantic web in action. *Scientific American*, 297(6):90–97, 2007. URL: `http://www.scientificamerican.com/article/semantic-web-in-actio/`. 12

[35] Erick R Fonseca and Joao Luis G Rosa. A two-step convolutional neural network approach for semantic role labeling. In *Neural Networks (IJCNN), The 2013 International Joint Conference on*, pages 1–7. IEEE, 2013. 24, 26, 32, 77, 94

[36] Brian R. Gaines. Knowledge acquisition: past, present and future. *IJHCS*, 71:135–156, 2013. `doi:10.1016/j.ijhcs.2012.10.010`. 5, 10, 11, 16

[37] Matthew S Gerber. Predicting crime using twitter and kernel density estimation. *Decision Support Systems*, 61:115–125, 2014. 43

[38] Daniel Gildea and Daniel Jurafsky. Automatic labeling of semantic roles. *Computational linguistics*, 28(3):245–288, 2002. 24

[39] Google. *A Google Geocoding API*. Google, 2015. URL: `https://developers.google.com/maps/documentation/geocoding/`. x, 41, 65

[40] Tom Heath and Christian Bizer. Linked data: Evolving the web into a global data space. *Synthesis lectures on the semantic web: theory and technology*, 1(1):1–136, 2011. 14

[41] Jim Hendler and Tim Berners-Lee. From the semantic web to social machines: A research challenge for ai on the world wide web. *ARTINT*, 174:156–161, 2010. doi:10.1016/j.artint.2009.11.010. 5, 11, 15, 16, 31

[42] Paul Kingsbury and Martha Palmer. From treebank to propbank. In *LREC*. Citeseer, 2002. 25, 45

[43] Barbara A. Kitchenham, Hiyam Al-Kilidar, Muhammad Ali Babar, Mike Berry, Karl Cox, Jacky Keung, Felicia Kurniawati, Mark Staples, He Zhang, and Liming Zhu. Evaluating guidelines for reporting empirical software engineering studies. *Empirical Software Engineering*, 13(1):97–121, 2008. doi:10.1007/s10664-007-9053-5. 37

[44] Barbara A Kitchenham, Tore Dyba, and Magne Jorgensen. Evidence-based software engineering. In *Software Engineering, 2004. ICSE 2004. Proceedings. 26th International Conference on*, pages 273–281. IEEE, 2004. 37, 86

[45] Barbara A Kitchenham, Shari Lawrence Pfleeger, Lesley M Pickard, Peter W Jones, David C. Hoaglin, Khaled El Emam, and Jarrett Rosenberg. Preliminary guidelines for empirical research in software engineering. *Software Engineering, IEEE Transactions on*, 28(8):721–734, 2002. xii, 37, 39

[46] Graham Klyne, Jeremy J. Carroll, and Brian McBride. RDF 1.1 concepts and abstract syntax. Technical report, W3C, February 2014. URL: http://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/. 11

[47] Marcelo Leite. Datafolha aponta saúde como principal problema dos brasileiros, 2014. URL: http://folha.com/no1432478. 2

[48] Julita LEMGRUBER et al. Arquitetura institucional do sistema único de segurança pública. *Acordo de Cooperação Técnica: Ministério da Justiça, Secretaria Nacional de Segurança Pública, Federação das Indústrias do Rio de Janeiro, Serviço Social da Indústria e Programa das Nações Unidas para o Desenvolvimento. Distrito Federal*, 2004. 33

[49] Debora MacKenzie. Brazil's uprising points to rise of leaderless networks. *New Scientist*, 218(2923):9, 2013. 16

[50] Silvio RL Meira, Vanilson AA Buregio, Leandro Marques Nascimento, Elaine Figueiredo, Misael Neto, Bruno Encarnação, and Vinícius Cardoso Garcia. The emerging web of social machines. In *Computer Software and Applications Conference (COMPSAC), 2011 IEEE 35th Annual*, pages 26–27. IEEE, 2011. 3, 4, 15, 16, 17

[51] MJ. Dicionário de dados do SINESJPC, 2014. URL: http://migre.me/kHPG4. 47

[52] Leandro Marques do Nascimento, Vanilson A.A. Burégio, Vinicius C. Garcia, and Silvio R.L. Meira. A new architecture description language for social machines. In *Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion*, WWW Companion '14, pages 873–874, Republic and Canton of Geneva, Switzerland, 2014. International World Wide Web Conferences Steering Committee. URL: http://dx.doi.org/10.1145/2567948.2578831, doi:10.1145/2567948.2578831. 4, 8

[53] Natalya F Noy, Deborah L McGuinness, et al. Ontology development 101: A guide to creating your first ontology, 2001. URL: http://liris.cnrs.fr/~amille/enseignements/Ecole_Centrale/What%20is%20an%20ontology%20and%20why%20we%20need%20it.htm. 12, 77

[54] OGP. Open government declaration, 2011. URL: http://www.opengovpartnership.org/about/open-government-declaration. 15

[55] ONU-Habitat. *Estado de las ciudades de América Latina y el Caribe 2012*. ONU-Habitat, 2012. URL: http://www.onuhabitat.org/index.php?option=com_docman&task=cat_view&gid=362&Itemid=538. 2

[56] Tim O'reilly. What is web 2.0: Design patterns and business models for the next generation of software. *Communications & strategies*, 65(1):17–37, 2007. 11

[57] Vasin Punyakanok, Dan Roth, and Wen-tau Yih. The importance of syntactic parsing and inference in semantic role labeling. *Computational Linguistics*, 34(2):257–287, 2008. 43

[58] Onde Fui Roubado. Estatísticas para São Paulo do aplicativo Onde Fui Roubado, 2014. URL acessed on 2014-08-20. URL: http://www.ondefuiroubado.com.br/sao-paulo/SP/estatisticas. 96

[59] Matthew A. Russell. *Mining the Social Web*. O'Reilly Media, first edition, January 2011. 31

[60] Max Schmachtenberg, Christian Bizer, and Heiko Paulheim. Adoption of the linked data best practices in different topical domains. In *The Semantic Web–ISWC 2014*, pages 245–260. Springer, 2014. xii, 14, 15

[61] Nigel Shadbolt. Knowledge acquisition and the rise of social machines. *IJHCS*, 71:200–205, 2013. doi:10.1016/j.ijhcs.2012.10.008. 2, 3, 4, 5, 8, 15, 16, 20, 97

[62] Mary Shaw. Writing good software engineering research papers: minitutorial. In *Proceedings of the 25th international conference on software engineering*, pages 726–736. IEEE Computer Society, 2003. xii, 6, 60, 80

[63] SmartBear. *The Swagger Specification*, 2015. URL: https://github.com/swagger-api/swagger-spec. 63

[64] Marina Sokolova and Guy Lapalme. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4):427–437, 2009. xii, 31, 32

[65] Mike Taylor. Python twitter, 2015. URL: https://github.com/bear/python-twitter. 43

[66] Twitter. Public streams, 2015. URL: https://dev.twitter.com/streaming/public. 22

[67] Twitter. The search api, 2015. URL: https://dev.twitter.com/rest/public/search. 22, 81

[68] Twitter. Termos de uso – twitter developers, 2015. URL: https://dev.twitter.com/pt/overview/terms. 23

[69] Twitter. Tweets, 2015. URL: https://dev.twitter.com/overview/api/tweets. 22

[70] W3C. Semantic web, 2015. URL: http://www.w3.org/standards/semanticweb/. 11

[71] Xiaofeng Wang, Donald E Brown, and Matthew S Gerber. Spatio-temporal modeling of criminal incidents using geographic, demographic, and twitter-derived information. In *Intelligence and Security Informatics (ISI), 2012 IEEE International Conference on*, pages 36–41. IEEE, 2012. 9, 39, 46

[72] Xiaofeng Wang, Matthew S Gerber, and Donald E Brown. Automatic crime prediction using events extracted from twitter posts. In *Social Computing, Behavioral-Cultural Modeling and Prediction*, pages 231–238. Springer, 2012. 3, 5, 9, 25, 30, 37, 43, 45, 79

[73] Radim Řehůřek. Gensim: Tutorials, 2015. URL: https://radimrehurek.com/gensim/tutorial.html. xiii, 28, 29

[74] Radim Řehůřek and Petr Sojka. Software framework for topic modelling with large corpora. In *Proceedings of LREC 2010 workshop New Challenges for NLP Frameworks*, pages 46–50, Valletta, Malta, 2010. University of Malta. URL: http://www.fi.muni.cz/usr/sojka/presentations/lrec2010-poster-rehurek-sojka.pdf. 28, 94