

UbibusRoute: Usando Informações Contextuais de Redes Sociais para Sugestão de Rotas de Ônibus

Vanessa Gomes de Lima
Centro de Informática
Universidade Federal de Pernambuco
Recife, Brasil
vgl2@cin.ufpe.br

Ana Carolina Salgado
Centro de Informática
Universidade Federal de Pernambuco
Recife, Brasil
acs@cin.ufpe.br

RESUMO

Sistemas de Informação ao Usuário do transporte coletivo visam fornecer informações aos passageiros e apoiar suas decisões. A maioria dos sistemas com esse propósito utiliza informações estáticas ou auxiliadas por transmissores GPS instalados nos veículos. Os cidadãos podem contribuir com a melhoria dos serviços públicos e isto pode ser feito, por exemplo, com o uso das redes sociais. Este trabalho apresenta o UbibusRoute, um sistema móvel que considera informações contextuais dinâmicas do trânsito provenientes de redes sociais e apresenta informações sobre as rotas aos usuários, utilizando a API do Google Maps como fonte de geolocalização e a API do Twitter como provedora de informações dos cidadãos.

Palavras-chave

Sistemas Inteligentes de Transporte, Contexto, Aplicações Móveis, Redes Sociais.

1. INTRODUÇÃO

O cenário atual do trânsito nas grandes cidades brasileiras vem piorando a cada dia e os congestionamentos tornaram-se cada vez mais frequentes. Devido ao intenso tráfego, o horário em que os ônibus passam em seus respectivos pontos de parada pode se tornar imprevisível. Esse é um grande problema enfrentado pela população, pois de acordo com Cutolo [1], uma grande barreira para os passageiros é a ausência de informações relativas aos serviços e/ou sua baixa qualidade. Como opção de solução de alguns desses problemas surgiram os Sistemas Inteligentes de Transporte (SIT), que consistem na utilização da Tecnologia da Informação e Comunicação (TIC) para subsidiar a infraestrutura dos sistemas de transporte [2].

Como aplicações de SITs, existem os Sistemas Avançados de Transporte Público (SATP), que apoiam o aumento da eficiência e segurança dos sistemas de transporte público [3]. Dentro dos SATP, existem os Sistemas de Informações aos Usuários (SIU), que são as ferramentas de comunicação entre os gestores do transporte público e os usuários, fornecendo informações que satisfaçam necessidades específicas dos passageiros, tais como: horário de chegada dos ônibus nas paradas, tempo de espera e tempo de viagem.

Em paralelo, observa-se o aumento da popularidade dos dispositivos móveis com acesso à Internet, e o uso das redes sociais contribuindo para o surgimento de novos aplicativos que possam atender às diversas necessidades dos usuários. As redes sociais vêm, dessa forma, ajudando as pessoas a lidar com seus problemas diários, uma vez que facilitam a troca e o compartilhamento de informações dinâmicas, em tempo real,

relacionados aos mais diversos aspectos do cotidiano, inclusive as situações que envolvem o trânsito. Segundo Levy [4], a área de *Crowdsourcing* (ou inteligência coletiva) incentiva a combinação do conhecimento individual provido por um grupo de pessoas para produzir novas informações ou ideias mais úteis.

Existem vários tipos de aplicativos baseados em localização que fornecem informações aos usuários sobre o trânsito (e.g. *Google Maps*¹ [5]). Esses aplicativos provêm, em sua maioria, informações sobre rotas e estimativa de tempo de chegada, mas não são totalmente direcionadas aos passageiros de ônibus e não levam em consideração acontecimentos dinâmicos como engarrafamentos, acidentes, alagamentos, entre outros. Estas informações dinâmicas são chamadas informações contextuais [6], que tornam a aplicação desenvolvida mais adaptativa ao usuário, satisfazendo suas preferências e necessidades.

O problema, nos casos descritos, é que os passageiros do transporte público não têm um serviço que os apoie na decisão sobre que ônibus e rota tomarem para chegar aos seus destinos, contornando situações de tráfego intenso ou acidentes. Em nosso projeto de pesquisa, denominado Ubibus [7], investigamos o uso de tecnologias relacionadas a dispositivos móveis, web e mídias sociais como apoio aos passageiros de transporte público em grandes e médios centros urbanos. Neste contexto, propomos o UbibusRoute [8], uma solução direcionada aos usuários de transporte coletivo por ônibus, que usa informações provenientes de redes sociais para recomendar rotas a esses usuários apoiando-os em suas tomadas de decisão. O sistema utiliza como fonte de informações contextuais dinâmicas a rede social *Twitter*² e a API de geolocalização *Google Maps*.

O restante desse artigo está organizado da seguinte forma: a Seção 2 descreve alguns trabalhos correlatos ao proposto; a Seção 3 apresenta o sistema UbibusRoute, sua arquitetura e questões técnicas de implementação; a Seção 4 detalha um cenário de uso experimental realizado com o intuito de avaliar o sistema proposto; a Seção 5 discute as contribuições e conclusões alcançadas com essa pesquisa e as perspectivas de trabalhos futuros.

2. TRABALHOS RELACIONADOS

Durante pesquisas encontramos alguns trabalhos que recomendam rotas aos usuários, como Mazhelis [5], mas não são voltados ao transporte público. Hoar [9] e Ferris [10] apresentam sistemas que exibem rotas de ônibus aos usuários, mas não processam dados dinâmicos de redes sociais. A utilização de tipo

¹ <http://maps.google.com>

² <http://www.twitter.com>

de dados contextuais é vista em um SIT denominado *Waze*³, mas este não contempla o transporte público por ônibus. Existem ainda algumas propostas, como Alves [11], para desenvolvimento de um sistema de recomendação de rotas utilizando inteligência coletiva de redes sociais.

O presente trabalho apresenta um sistema de recomendação de rotas aos usuários de transporte público por ônibus, utilizando dados contextuais dinâmicos oriundos de redes sociais. Este sistema foi desenvolvido para uma plataforma móvel, usando *smartphones*, e será descrito na próxima seção.

3. UBIBUS ROUTE

Esta seção descreve a arquitetura e detalhes de implementação do UbibusRoute. O sistema que propõe a sugestão de rotas baseadas no conhecimento das informações dinâmicas.

4.1 Arquitetura

O UbibusRoute foi projetado em uma arquitetura cliente-servidor (Figura 1) onde o cliente contém um componente chamado Aplicativo Móvel, que pode ser nativo de qualquer sistema operacional móvel que se comunique com o servidor seguindo os padrões fornecidos pelo mesmo.

Existem duas bases de dados principais no sistema, a Base de Dados Contextuais de Trânsito, e a Base de Dados Estáticos. Os dados estáticos são paradas, linhas de ônibus e percursos. No módulo Cliente, o *Aplicativo Móvel* envia uma requisição de dados estáticos e de pedido de rota ao servidor, recebendo como resultado a rota e as instruções para percorrê-la. Já o módulo Servidor é dividido em três componentes principais: o *Identificador de Rotas*, responsável por identificar todas as rotas possíveis de acordo com a localização ou parada de ônibus selecionada; o *Indicador de Rotas*, responsável por selecionar a melhor rota de acordo com a preferência do usuário (tempo, preço ou distância); e o *Extrator de Informações Contextuais* que colhe informações de redes sociais e verifica como o trânsito está naquele momento.

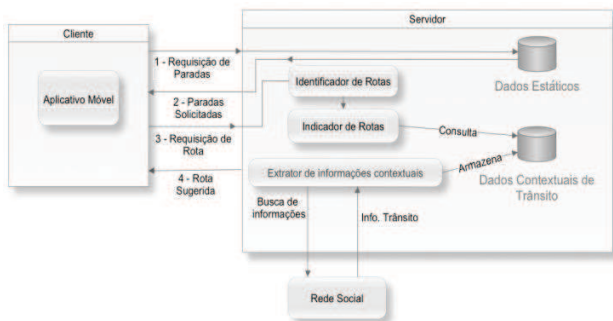


Figura 1 - Arquitetura Geral do UbibusRoute

4.1.1. Aplicativo Móvel

O Cliente, na arquitetura, contém um módulo chamado *Aplicativo Móvel*, que é a interface com o usuário. Na execução do aplicativo, é realizada inicialmente uma requisição automática de dados das paradas de ônibus, sendo fornecido ao usuário um mapa onde o mesmo pode visualizar todas as paradas de ônibus disponíveis e selecionar duas: uma parada de ônibus como origem e outra como destino. Além disso, o usuário deve informar se

prefere que a rota sugerida seja baseada em alguns fatores como: menor preço, viagem com menor tempo ou menor distância percorrida. Os dados das paradas de ônibus selecionadas, bem como o tipo de viagem escolhido, serão enviados para o *Identificador de Rotas* no servidor.

A resposta obtida do servidor tem características visuais e textuais, onde consta uma rota traçada em um mapa, enfatizando o ponto de origem, o ponto de destino, os pontos de troca de ônibus (caso existam), o preço de cada um deles e uma mensagem textual. Tal visualização segue de acordo a opção selecionada pelo usuário inicialmente, contendo a distância total informada, o preço total a ser gasto, ou ainda o tempo total de viagem. Na resposta textual, o usuário recebe um texto explicando todas as etapas que o passageiro deve seguir para chegar ao seu destino.

4.1.2. Identificador de Rotas

O processo de identificar rotas, que acontece no componente *Identificador de Rotas*, é iniciado a partir de uma requisição por parte do *Aplicativo Móvel*. Desenvolveu-se um algoritmo de busca de rotas modificando o algoritmo guloso de Dijkstra⁴ para que seja retornado o menor caminho, levando em consideração o parâmetro vindo do aplicativo móvel, ou seja, o *menor caminho* não é sempre pela distância; a depender do parâmetro, o peso das arestas do grafo que o algoritmo Dijkstra gerencia muda de acordo com a *preferência do usuário*.

A função do algoritmo desenvolvido é, portanto, procurar os possíveis caminhos que possam ser traçados entre a origem e o destino, oferecendo como saída essas rotas com parâmetros que indicam preço da viagem, distância a ser percorrida e tempo total. Este módulo classifica, então, cada possível rota com três pontuações numéricas: uma para a distância, uma para o tempo e uma para o preço a ser gasto. A saída desse módulo é enviada ao *Indicador de Rotas*.

4.1.3 Indicador de Rotas

O componente *Indicador de Rotas*, por sua vez, utiliza todas as rotas possíveis, recuperadas a partir do processo de identificação, e seleciona a melhor, baseada na preferência inicial definida pelo usuário (menor preço, menor distância ou menor tempo). Esta seleção é feita a partir da saída das rotas repassadas do módulo *Identificador de Rotas*. Como cada rota possui uma pontuação de tempo, custo de preço e distância, é escolhida a que contém o menor valor do parâmetro indicado pelo usuário. Este componente consulta, também, informações contextuais armazenadas pelo *Extrator de Informações Contextuais*, que estão guardadas na base de *Dados Contextuais*, provenientes de redes sociais. Sendo assim, o *Indicador de Rotas* é responsável por cruzar as preferências do usuário com as informações contextuais capturadas das redes sociais, atribuindo uma nova pontuação relativa ao tempo para as possíveis rotas.

4.1.4. Extrator de Informações Contextuais, Dados Contextuais e Redes Sociais

O objetivo do *Extrator de Informações Contextuais* é capturar informações dinâmicas de redes sociais. É possível obter mensagens recentes relacionadas aos trechos do trânsito por onde passam as linhas de ônibus cadastradas no UbibusRoute.

A rede social explorada no UbibusRoute foi o *Twitter*, devido à natureza curta de suas mensagens, com no máximo 140

³ <http://www.waze.com/>

⁴ <http://www.cs.auckland.ac.nz/~jmor159/PLDS210/dijkstra.html>

caracteres. O módulo das Redes Sociais está intrinsecamente ligado ao *Extrator de Informações Contextuais*. Os *tweets* são extraídos em intervalos de tempo periódicos. Na versão inicial deste sistema, a informação dos *tweets* foi interpretada com o uso de técnicas de expressões regulares. Por fim, esses *tweets* armazenados em uma *Base de Dados Contextuais de Trânsito*, localizada no servidor. Tais informações são utilizadas para conhecer a situação atual do trânsito.

As mensagens provenientes das redes sociais precisariam estar em um formato pré-estabelecido, a fim de identificar a informação. Essa limitação acabava perdendo outras mensagens que contenham informações importantes e relevantes sobre o trânsito. No entanto, tendo as mensagens nesse formato pré-definido, há um ganho de velocidade no processo de identificação textual das diversas situações de trânsito, melhorando o desempenho e garantido correteza na informação identificada.

No primeiro protótipo, para experimentação, foi desenvolvida uma gramática de reconhecimento e padronização das mensagens. As mensagens deveriam ser compostas por um número que identifica a hora, <HORA>, os minutos <MIN>, um fato agravante <AGRAVANTE>, que é uma palavra que intensifica uma situação de trânsito (a presença de um agravante na mensagem a ser identificada é facultativa). Uma situação de trânsito <SITUAÇÃO>, por sua vez, é uma palavra que indica congestionamento ou fluidez no trânsito, em diferentes níveis. Uma preposição <PREPOSIÇÃO> é apenas um conector entre a situação de trânsito e a localidade da ocorrência, composta por <LOCAL> e <TRECHO>. A gramática proposta é ilustrada na Figura 2.

<HORA> h <MIN> min Trânsito <AGRAVANTE> <SITUAÇÃO>
<PREPOSIÇÃO> <LOCAL> <TRECHO>

Onde:

<HORA> → número de 00 a 23;
<MIN> → número de 00 a 59;
<AGRAVANTE> → muito | pouco | quase | bastante | meio
<SITUAÇÃO> → parado | lento | ruim | bom | livre | fácil | difícil | moderado | péssimo | complicado | devagar | rápido | congestionado | engarrafado | fluindo | normal | tranquilo | leve | pesado
<PREPOSIÇÃO> → em | na | no
<LOCAL> → nomes de ruas da base dados
<TRECHO> → trechos cadastrados na base de dados.

Figura 2 - Gramática para Extração de Mensagens de Trânsito de Redes Sociais.

As informações contextuais extraídas do *Twitter* são armazenadas na base de dados como um tipo composto [localização geográfica, dia, horário, *pontuação*]. A *pontuação*, que será detalhada na próxima seção, indica o grau de congestionamento de determinado local.

4.2. Aspectos de Implementação

O aplicativo móvel foi desenvolvido sobre a plataforma *Android*, em *Java* e usando alguns recursos da API do *Google Maps*. A comunicação com o Servidor é feita seguindo os princípios REST (*Representational State Transfer*), onde o módulo Cliente faz uma requisição HTTP, contendo *origem*, *destino* e *tipo de busca*, e o módulo Servidor responde com um objeto JSON (*JavaScript Object Notation*).

O servidor do UbibusRoute foi implementado com o framework *Django* e todos os seus módulos foram desenvolvidos

em *Python*. A resposta do servidor ao aplicativo móvel é dada a partir da interação entre os componentes: *Identificador de Rotas* e *Indicador de Rota*, consultando os Dados Contextuais de Trânsito. Tal resposta possui vários parâmetros como: pontos geográficos da origem e destino, dados geográficos das paradas onde houver troca de ônibus; preço de cada troca de ônibus, caso exista; mensagem de resposta textual; e a mensagem final da requisição, que contém: a distância a ser percorrida, o custo de tempo total da viagem, ou o custo relativo ao preço.

A comunicação com o *Twitter* é realizada pelo *Extrator de Informações Contextuais* por meio da API REST⁵ do *Twitter*. Isto permite acesso a dados essenciais como: *tweets* (mensagens postadas pelos usuários), prazos de atualização e informações dos usuários. A atualização dos dados contextuais de trânsito ocorre de acordo com a atual situação de trânsito informada pelos *tweets*.

Para manter os dados atualizados sobre o estado atual do trânsito, foi desenvolvido um *crawler* (componente que busca continuamente os *tweets* relativos ao trânsito), que se encontra dentro do *Extrator de Informações Contextuais*. Este *crawler* utiliza a API REST do *Twitter* e captura automaticamente as informações do perfil @Ubibus_PE de acordo com dois parâmetros: quantidade máxima de *tweets* por captura - definida como 20, para efeitos de desempenho, e a diferença entre os instantes de tempo atual e de criação do *tweet* - definido como 20 minutos na versão atual (podendo ser parametrizado em versões futuras). Com isso, é possível obter os 20 últimos *tweets* que foram publicados no perfil.

Também foi construído um *script* em *Python* que formata o conteúdo capturado do perfil do *Twitter*, extraindo o trecho e a situação atual do trânsito. Este *script* contém um algoritmo de interpretação de informações de trânsito que opera considerando a Gramática definida anteriormente (Seção 3.1.4). A situação do trânsito é calculada de acordo com algumas variáveis que classificam o adjetivo <SITUAÇÃO> combinado com o seu <AGRAVANTE>. De acordo com a *situação* e a presença ou não de um *agravante*, é calculada uma pontuação para aquele endereço. Essa pontuação é válida por 20 minutos, (tempo considerado suficiente para mudança na fluidez do tráfego). Para facilitar a extração das informações do trânsito, foi criado um conjunto de classificação das *situações* e dos *agravantes*. A situação pode estar classificada como [*bom*, *médio* e *ruim*] e os agravantes como [*alto*, *baixo*].

Toda rua que é representada na base de dados possui, estaticamente, um indicativo de velocidade da via. O tempo de trânsito (a pontuação) é calculado em função da mudança que pode ocorrer nessa velocidade. Uma vez que o trânsito está livre (sem interrupções ou congestionamentos), a velocidade não é alterada e, portanto, a pontuação fica zerada. Na primeira versão deste sistema, o retorno desse algoritmo consiste de uma pontuação que oscila de 0 a 3 onde: 0 (zero) indica que o trânsito está livre, como descrito anteriormente; 1 (um) indica que o trânsito está moderado, quando houve detecção dos classificadores de situação *médio*; quando da detecção do classificador *ruim*, a pontuação atribuída é 2 (dois) e indica que o trânsito está lento; e, por fim, 3 (três) indica que o trânsito está muito lento, quando da junção do classificador *ruim* com o agravante *alto*. Posteriormente, estes números são utilizados no algoritmo de busca de rotas para influenciar na decisão de qual rota sugerir e qual tempo informar ao usuário.

⁵ <http://dev.twitter.com/docs/api>

5. CENÁRIO DE USO DO UBIBUS ROUTE

Foi desenvolvido um cenário de experimentação do sistema, que teve como objetivo verificar o comportamento das recomendações de acordo com as informações contextuais extraídas do *Twitter*. As bases de dados foram simuladas com a criação de um perfil no *Twitter* (@Ubibus_PE), e alguns endereços da cidade do Recife, simulando neles localizações de pontos de ônibus com latitude, longitude e identificadores. A partir daí, foram simuladas rotas de linhas de ônibus, preços e intervalo de tempo de saída dos ônibus dos terminais.

Ao usuário, foi disponibilizada uma aplicação móvel (como descrita anteriormente), onde ele poderia escolher origem e destino. À medida que determinado endereço fosse comentado, e sua pontuação depreciada, de acordo com as informações contextuais, a sugestão das rotas baseadas no tempo de viagem ia mudando. Na Figura 3, é possível constatar um resultado de consulta de rotas baseada em menor tempo (custo).

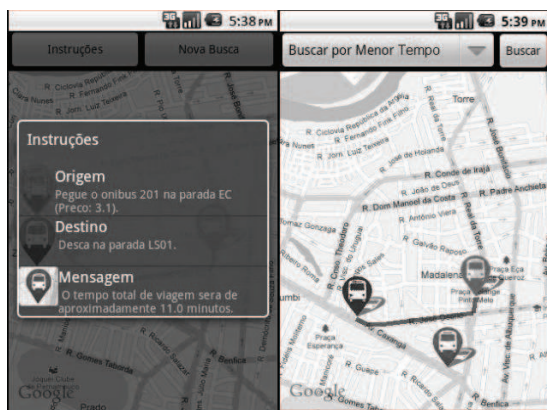


Figura 3 - Mapa e informações da rota indicada

À esquerda, a resposta textual dada ao usuário e à direita o mapa com origem e destino marcados e o trajeto traçado.

6. CONCLUSÃO

Este trabalho teve por objetivo desenvolver uma aplicação móvel para indicar e sugerir rotas aos usuários de transporte público, capaz de obter informações contextuais dinâmicas da rede social *Twitter*. Tal recomendação é baseada em pontos de origem, destino e tipo de busca, fornecidos pelos usuários. Para implementação do UbibusRoute [8] foram utilizadas ferramentas externas como a API do *Google Maps* e *Twitter*. O aplicativo mostrou que é possível recomendar rotas de ônibus aos usuários utilizando informações dinâmicas extraídas de redes sociais, apesar de demandar algumas melhorias que serão abordadas em pesquisas futuras.

Alguns trabalhos futuros serão investigados com o intuito de contribuir para a melhoria do sistema desenvolvido, entre eles podemos citar: a implementação de outros algoritmos para a busca

por menor preço, a utilização de uma base de dados reais, o uso de informações provenientes de outras Redes Sociais, como *Facebook*, *Instagram* e *Foursquare*, além de mudanças na forma de busca e interpretação dessas informações. Estão também sendo desenvolvidas técnicas baseadas em Análise de Sentimentos, a fim de descobrir o *sentimento* em mensagens de usuários sobre o trânsito, para então extrair informação necessária sobre a pontuação das rotas, como funciona atualmente no UbibusRoute.

REFERÊNCIAS

- [1] Cutolo, F. A. (2003) "Diretrizes para sistema de informação ao usuário". In: III Seminário Internacional PROMOTEO, Porto Alegre-RS.
- [2] Gómez, A., Diaz, G. and Bousetta, K. (2009) "ITS Forecast: GIS Integration with Active Sensory System". In: Global Information Infrastructure Symposium, GIJS'09. Hammamet, Tunisia, pp. 1-6.
- [3] Sussman, J. (2005), "Perspectives on Intelligent Transportation Systems". New York, USA: Springer.
- [4] Levy, P. (2003), "A Inteligência Coletiva". São Paulo: Loyola.
- [5] Mazhelis, O., Zliobaite, I. and Pechenizkiy, M. (2011) "Context-aware personal route recognition". In: The Fourteenth International Conference on Discovery Science (DS 2011), Espoo, Finland, pp. 365-379.
- [6] Brézillon, P. (1999) "Context in Artificial Intelligence: IA Survey of the Literature", *Computer & Artificial Intelligence*, v. 18, pp. 321-340.
- [7] Vieira, V. Salgado, A.C., Tedesco, P., Times, V. C., Ferraz, C., Huzita, E., Chaves, A. P., Steinmacher, I. The UbiBus Project: Using Context and Ubiquitous Computing to build Advanced Public Transportation Systems to Support Bus Passengers. In: Anais do VIII Simpósio Brasileiro de Sistemas de Informação, 2012, São Paulo, pp. 55-60.
- [8] Lima, V., Magalhães, F., Tito, A., Santos, R., Ristar, A., Santos, L., Vieira, V., Salgado, A. C. (2012) "UbibusRoute : Um Sistema de Identificação e Sugestão de Rotas de Ônibus Baseado em Informações de Redes Sociais". In: VIII Simpósio Brasileiro de Sistemas de Informação (SBSI) – Cidades Inteligentes, São Paulo-SP, pp. 516-527.
- [9] Hoar, R. (2010) "A Personalized Web Based Public Transit Information System with User Feedback". In: 13th International IEEE Conference on Intelligent Transportation Systems, Ilha da Madeira, Portugal.
- [10] Ferris, B., Watkins, K. and Borning, A. (2010) "Location-Aware Tools for Improving Public Transit Usability", *IEEE Pervasive Computing*, v. 9 n. 1, pp. 13-19.
- [11] Alves, L. P. S., Chaves, A. P e Steinmacher, I. F. (2011) "Um aplicativo baseado em inteligência coletiva para compartilhamento de rotas em redes sociais". In: VIII Simpósio Brasileiro de Sistemas Colaborativos (SBSC), Paraty-RJ.