MÉTODOS PARA EXTRAÇÃO DE PADRÕES SEQUENCIAIS DE COMPORTAMENTO EM CASAS INTELIGENTES

GUSTAVO M. FIGUEIREDO E FLAVIO TONIDANDEL

CENTRO UNIVERSITÁRIO FEI,
AVENIDA HUMBERTO DE ALENCAR CASTELO BRANCO 3972, 09850-901, SBC, SP, BRASIL
E-MAIL: GUSTAVO.MOLINA17@YAHOO.COM.BR; FLAVIOT@FEI.EDU.BR

Abstract - A Smart Home has different types of systems that interact with the inhabitants that are looking for more comfort and a better quality within the residence. To increase the comfort and life quality of inhabitants, it is necessary to develop a system that adapts itself to the inhabitant's behavior. This article presents a study of the GSP, SPADE, PrefixSpan algorithms used for the extraction of sequential patterns. The fundamental concepts will be presented and later it will be explained how they are going to be implemented in order to extract the sequential patterns in smart homes.

Keywords – Sequential Patterns Extraction. Smart Home. Residential Automation. Algorithms.

Resumo - Uma casa inteligente possui diversos tipos de sistemas que interagem com os moradores que buscam mais conforto no interior da residência. Para aumentar o conforto e qualidade de vida torna-se necessário o desenvolvimento de um sistema que se adapte ao comportamento dos habitantes. Este artigo apresenta um estudo sobre os algoritmos (GSP, SPADE e PrefixSpan) utilizados para a extração de padrões sequenciais. Serão apresentados os conceitos fundamentais e posteriormente explicado como serão implementados para extrair os padrões sequenciais em casas inteligentes.

Palavras-chave - Extração de Padrões Sequenciais. Casa Inteligente. Automação Residencial. Algoritmos

1 Introdução

Atualmente, os habitantes das casas inteligentes, também conhecidas como *smart homes*, se interessam por mais conforto e uma melhor qualidade de vida no interior da residência. Visando o conforto é coerente afirmar que os sistemas mais eficientes são aqueles que se adaptam ao comportamento dos habitantes, ao invés de os habitantes terem de se adaptar ao sistema. Isto permite que no futuro, as casas inteligentes possam aumentar o conforto, economizar energia ou melhorar a segurança para idosos residentes em lares de vida assistida.

Segundo Weiser (1993, apud KING, 2003), uma casa inteligente é definida como uma "habitação incorporando uma rede de comunicações que conecta aparelhos elétricos e serviços que podem ser controlados, monitorados ou acessados remotamente".

Com as tecnologias que, nos dias atuais, já permitem um nível razoável de integração, a ideia de ter controle sobre certos equipamentos e algum nível de automação, visando comodidade e conforto do usuário, torna-se cada vez mais real (FUSSEK, 2010).

Baseado nas considerações acima torna-se desejável o desenvolvimento de sistemas, de fácil utilização pelos habitantes da residência (a complexidade dos controles dos sistemas presentes na casa com a automação não pode ser maior do que sem o uso dela) e, com o passar do tempo, cada vez mais adaptativos, evoluindo para o conceito de domótica inteligente, onde as regras de automação adaptam-se ao comportamento dos moradores.

Para aumentar o conforto dos habitantes e reduzir o consumo de energia em uma casa inteligente, é necessário que ela seja capaz de aprender o comportamento de seus habitantes e consiga prever suas necessidades futuras aplicando regras de automação. Para que as regras de automação possam ser aplicadas, é necessário que a casa inteligente saiba as preferências e os padrões de comportamento de cada habitante. Para reconhecer padrões, a casa inteligente analisa os dados de eventos advindos dos sensores (sensores de proximidade, vibração, movimento, temperatura, pressão, aceleração, luz e mudança de estado) instalados em objetos do dia a dia, tais como armários, gavetas, geladeira, portas, eletrodomésticos, conhecidos como *HVAC* (dispositivos de aquecimento e arrefecimento), sistemas de iluminação e das persianas (HOPPNER, 2002). Quando a análise dos dados termina, a mineração desses dados é feita a fim de encontrar padrões frequentes (padrões que ocorrem mais frequentemente do que outros) e padrões periódicos (padrões que ocorrem em um intervalo constante).

Se a casa inteligente sabe todos os padrões periódicos e frequentes de seus habitantes, pode prever as próximas ações e obter os melhores resultados na automatização de tarefas. O objetivo desse artigo é comparar o desempenho dos algoritmos GSP, SPADE e PrefixSpan na extração de padrões sequenciais em casas inteligentes e verificar qual algoritmo é melhor para extrair autonomamente o comportamento de seus habitantes.

2 Trabalhos Relacionados

O rápido desenvolvimento das Tecnologias da Informação e Comunicação, as chamadas TIC, associadas à Automação (TICA), têm criado uma necessidade urgente de se definir teorias, métodos e conceitos para avaliar e analisar sua interação entre ambiente e usuários. Muitos trabalhos têm sido realizados pela comunidade científica em laboratórios experimentais para o desenvolvimento de ambientes inteligentes no mundo todo, graças ao progresso das tecnologias envolvendo redes domiciliares com e sem fio, sensores, métodos computacionais e dispositivos inteligentes (FUSSEK, 2010).

Na Suécia, *o Royal Institute of Technology* desenvolveu o projeto *comHOME*, com a finalidade de simular a casa do futuro (JUNESTRAND et al., 2001). Na *University of Tokyo*, foi elaborado outro projeto chamado *Ubiquitous Home*, com a finalidade de realizar testes para a criação de novos serviços destinados à ligação de aparelhos eletrodomésticos, sensores e atuadores em redes domiciliares (YAMAZAKI, 2006).

Diferente dos trabalhos desenvolvidos, neste trabalho o foco recai sobre a automação de residências privadas e formas de uma casa inteligente decidir o que é um comportamento normal do seu habitante pela mineração de padrões periódicos e frequentes. Além disso, o habitante não será obrigado a especificar os seus padrões de uso e preferências visto que os mesmos serão aprendidos pelo sistema.

3 Padrões Sequenciais em Casas Inteligentes

Um padrão é uma sequência de atividades realizadas pelo habitante. A sequência das atividades é muito importante quando os algoritmos estão analisando os dados gerados de cada habitante. Para aprender os hábitos individuais é fundamental saber se o eletrodoméstico X foi ligado antes ou depois do eletrodoméstico Y.

3.1 Conceitos Fundamentais

Mineração de padrões sequenciais representa uma importante classe de problemas de mineração de dados com grande variedade de aplicações. Minerar padrões sequenciais é uma atividade desafiadora, pois, se lida com uma verificação cuidadosa de um grande número de possíveis combinações (BOLZANI, 2010).

Usa-se a mineração de padrões em diferentes contextos, tais como as organizações empresariais para estudar o comportamento dos clientes, na mineração de uso da *web* para encontrar *web logs* distribuídos em vários servidores e para a identificação e análise dos comportamentos de habitantes de uma casa inteligente.

No contexto de mineração de dados e mineração de padrões de sequências em casas inteligentes, a identificação de ações frequentes, também chamadas de *itemsets* frequentes, é um dos conceitos mais importantes. A seguir, serão apresentados algumas definições importantes.

a) **Itemset:** Um *itemset* é um conjunto não vazio composto por uma ou mais ações realizadas pelo habitante representado por $I = \{i_1, i_2, ..., i_n\}$ para $1 \leq i \leq n$.

b) **Sequência:** Uma sequência é uma lista ordenada de ações dos habitantes (*itemsets*). Denota-se uma sequência S_i por $< s_i, s_j, ..., s_x > \in I$, onde s_i, s_j, e s_x são *itemsets* e $S \supset I$. Os *itemsets* s_i, s_j, e s_x também são considerados elementos da sequência onde $s_i \neq s_j \neq s_x \mid s_i \in S, s_j \in S, s_x \in S$.

c) **Frequência:** É a ocorrência de um *itemset* na base de dados. É denotada por σ.

d) **Suporte:** Denotado por s, o suporte corresponde à fração de σ considerando-se toda a base de dados.

e) **Suporte mínimo:** Denotado por minSup, é o valor mínimo que um *itemset* deve ter para ser considerado frequente.

3.2 Algoritmos de Descoberta de Padrões Sequenciais

Um algoritmo adequado permite que uma casa inteligente aprenda de forma autônoma os padrões de utilização (comportamento) dos seus habitantes.

Diversos algoritmos foram implementados para identificar sequências frequentes em bancos de dados de sequências. Uma das abordagens mais utilizadas para identificar sequências frequentes é a abordagem *Apriori*. De acordo com Agrawal e Srikant (1994), o algoritmo *Apriori* foi originalmente concebido com o objetivo de associar regras de mineração de dados e é baseado na propriedade *apriori* que diz que para um item ser frequente, todos os seus subconjuntos também devem ser.

Existem três algoritmos considerados bem-sucedidos na extração de padrões sequenciais: GSP (*Generalized Sequential Patterns*) (AGRAWAL; SRIKANT, 1996), SPADE (*Sequential Pattern Discovery using Equivalence classes*) (ZAKI, 2001) e PrefixSpan (*Prefix-projected Sequential pattern mining*) (PEI *et al*., 2001).

3.2.1 Algoritmo GSP

O algoritmo GSP, criado em 1996, utiliza a estratégia de Geração e Teste de candidatos (consiste basicamente na repetição de duas etapas, Geração e Teste, até não ser mais possível a geração de sequências frequentes). O algoritmo utiliza o suporte mínimo minSup e a base de dados como parâmetros ao ser simulado e foi escolhido por se basear na técnica *Apriori* que é o método mais utilizado na mineração de padrões frequentes em bases de dados.

Uma vantagem do algoritmo GSP é que o suporte mínimo de cada item é obtido na primeira busca,

dessa forma obtêm-se os itens frequentes que cumprem o requisito de suporte mínimo.

Como desvantagem, o algoritmo apresenta a geração de uma grande quantidade de sequências candidatas, o que influencia na velocidade de processamento dos dados dependendo do tamanho da base de dados (AGRAWAL; SRIKANT, 1996).

3.2.2 Algoritmo SPADE

Uma grande vantagem do algoritmo SPADE é a utilização de propriedades combinatórias para decompor o problema original em subproblemas menores, que podem ser resolvidos de forma independente (ZAKI, 2001). O algoritmo utiliza o suporte mínimo minSup e a base de dados como parâmetros ao ser simulado e foi escolhido por não se basear na técnica *Apriori*, pois, encontra as sequências frequentes com apenas três varreduras no banco de dados.

O SPADE é baseado em listas que contêm informações sobre a localização dos padrões nas sequências e as consequências consecutivas desses padrões fazem com que a lista cresça rapidamente, podendo dessa forma, afetar o desempenho do algoritmo (ZAKI, 2001).

3.2.3 Algoritmo PrefixSpan

O algoritmo PrefixSpan utiliza uma técnica diferente da empregada no *Apriori*. Este algoritmo aplica o princípio de *Pattern-Growth* (Crescimento de Padrão) por meio da projeção de prefixos para geração de padrões sequenciais. Estas projeções dividem a base de dados em porções menores, dessa forma, o espaço de busca é menor e a identificação e extração das sequências frequentes são realizadas de forma mais rápida e eficaz. (PEI *et al*., 2001). Assim como os algoritmos anteriores, o suporte mínimo minSup e a base de dados são os parâmetros utilizados na simulação do algoritmo que foi escolhido por aplicar uma técnica diferente dos anteriores.

O algoritmo PrefixSpan é mais eficiente do que os algoritmos baseados no algoritmo *Apriori*. O consumo de memória é estável ao se executar o algoritmo.

O principal problema deste algoritmo é o tempo, que é consumido para se explorar as bases de dados projetadas, que podem ser muito grandes, dependendo do tamanho da base original.

4 Métodos

Considerando-se que, nos dias atuais a tarefa de mineração de padrões sequenciais em casas inteligentes é uma tarefa desafiadora, já que a busca pode exigir a análise de um alto número de padrões, torna-se muito complexa a simulação dos algoritmos GSP, SPADE e PrefixSpan em uma residência e situações reais, pois, uma simulação real dependeria de meses, talvez anos, de coletas de dados.

Além disto, há muitas variáveis a serem testadas, visando encontrar os critérios que tornarão a análise dos algoritmos eficientes para a extração dos padrões sequenciais. O tempo necessário para a coleta de dados suficientes, para avaliação de cada conjunto de critérios considerados, também seria impraticável.

Desta forma, a solução encontrada foi trabalhar com uma base (*dataset*) já existente de dados reais. A base de dados utilizada neste trabalho é do *Massachussets Institute of Technology* (MIT) de 2007. O *dataset* utilizado refere-se a uma casa inteligente (denominada *PlaceLab*) com um único habitante vivendo nela por um período de catorze dias. O *dataset* é público e está disponível para download na página do projeto House_n do MIT (MIT, 2016).

A casa está equipada com 77 sensores, de diferentes tipos: sensores de proximidade, sensores de aceleração (acelerômetros), sensores de vibração, sensores de movimentos, sensores de temperatura, sensores de pressão, sensores de luz e sensores de mudança de estado.

Todos os sensores foram instalados em objetos do dia a dia, tais como armários, gavetas, geladeira, portas, entre outros com o intuito de gravar eventos do tipo abrir/fechar (eventos de ativação/desativação dos sensores) enquanto o habitante executar suas tarefas diárias, dessa forma é possível obter a sequência de ações do mesmo.

Visando testar o comportamento, eficiência e desempenho dos algoritmos, para extrair padrões sequenciais, foi criado um *software* em computador denominado *Smart Home Pattern Miner*. Os algoritmos GSP, SPADE e PrefixSpan foram implementados na linguagem C++.

Devido ao propósito do uso dos algoritmos GSP, SPADE e PrefixSpan para a extração de sequencias em um sistema de automação residencial real, os testes efetuados com o *dataset* focam os seguintes pontos principais: precisão da extração dos padrões e qualidade das extrações.

A precisão da extração de padrões se refere ao grau de variação dos resultados de uma determinada medição.

A qualidade das extrações se refere à porcentagem de padrões que o algoritmo consegue identificar corretamente.

Quando o *software* recebe os dados advindos dos sensores, ele cria um arquivo.CSV com as ações realizadas pelos habitantes e as mesmas são ordenadas temporalmente, ou seja, o comportamento sequencial nesta base é definido pelos horários de início e término dos eventos.

No contexto deste trabalho, extrair corretamente um padrão significa extrair sequencialmente as ações realizadas pelo habitante na ordem em que as mesmas ocorreram, ou seja, serão mapeadas todas as ações realizadas diariamente pelo habitante.

O funcionamento ideal do sistema prevê a extração correta dos padrões sequenciais, de forma a possibilitar que seja possível identificar qual dos algo-

ritmos (GSP, SPADE ou PrefixSpan) é mais interessante para ser aplicado em casas inteligentes.

Para a validação dos resultados obtidos pelos algoritmos, os mesmos serão comparados com as ações do dataset, dessa forma é possível ter certeza que os padrões extraídos são corretos.

Para melhor análise do comportamento de cada algoritmo perante a quantidade de dados disponíveis, o teste foi divido em quatro etapas:

Etapa 1: Foi gerado o arquivo 01_09.CSV com todas as ações realizadas pelo habitante. Essa data foi escolhida, pois é o dia que possui menos eventos no *dataset*, ou seja, foi o dia em que o habitante realizou menos ações dentro do *PlaceLab*.

Etapa 2: Foi gerado o arquivo 05_09.CSV com todas as ações realizadas pelo habitante. Essa data foi escolhida, pois é o dia que possui mais eventos no *dataset*, ou seja, foi o dia em que o habitante realizou mais ações dentro do *PlaceLab*.

Etapa 3: Foi gerado o arquivo 50_base.CSV que corresponde a 50% do total de todas as ações realizadas pelo habitante no período de 01/09/2007 a 14/09/2007. Ao utilizar 50% da base, foram totalizados 109 eventos.

Etapa 4: A última etapa do teste consistiu na geração do arquivo 100_base.CSV que corresponde as ações realizadas pelo habitante no período de 01/09/2007 a 14/09/2007. A base possui 212 eventos no total.

Em um primeiro momento, o desempenho dos três algoritmos foi comparado utilizando-se o suporte mínimo minSup = 0,01. Para os testes iniciais foram escolhidos padrões (sequências de ações) com o comprimento mínimo de 2 eventos (ações) e comprimento máximo de 7 eventos (ações). Posteriormente o comprimento máximo passa a ser de 10 eventos.

Essas configurações de parâmetros são consideradas como representativas no *dataset* utilizado porque o suporte mínimo minSup de 0,01 (ou 1%) permite que os algoritmos sejam executados mais rapidamente do que quando um suporte mínimo baixo (por exemplo 0,001 ou menos) é escolhido. Valores mais altos para o suporte mínimo (0,1 ou acima) não produzem padrões para o *dataset* utilizado e, portanto, não podem ser utilizados nos testes. Padrões frequentes constituídos por menos de dois eventos não são interessantes ao tentar descobrir o que constitui um comportamento normal. Padrões constituídos por mais de dez eventos sequenciais são improváveis de ocorrer no dia a dia.

5 Resultados

Conforme citado anteriormente, a qualidade das extrações se refere a porcentagem de padrões que o algoritmo consegue identificar e extrair corretamente. Em todos os testes as configurações de parâmetros foram mantidas. Os resultados obtidos estão apresentados a seguir.

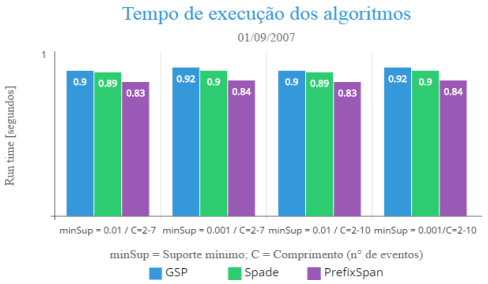

Figura 1. Tempo de execução dos algoritmos GSP, Spade e PrefixSpan em 01/09/2007.

Figura 2. Qualidade da extração dos algoritmos GSP, Spade e PrefixSpan em 01/09/2007.

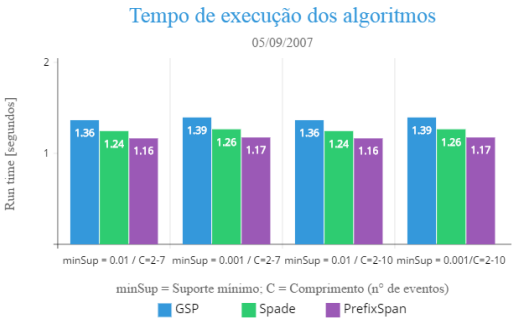

Figura 3. Tempo de execução dos algoritmos GSP, Spade e PrefixSpan em 05/09/2007.

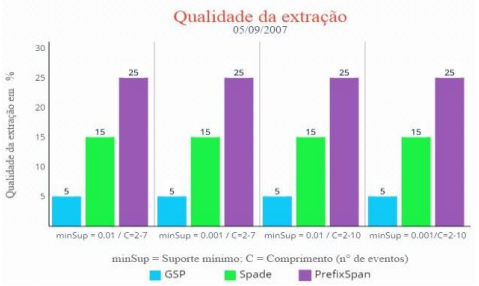

Figura 4. Qualidade da extração dos algoritmos GSP, Spade e PrefixSpan em 05/09/2007.

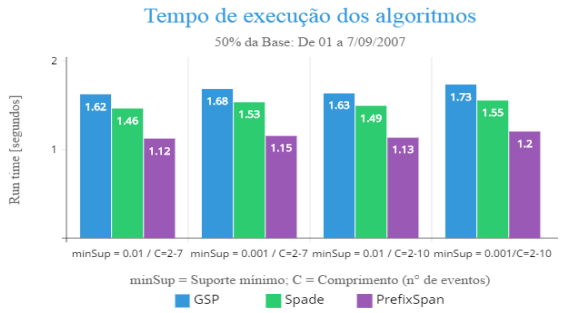

Figura 5. Tempo de execução dos algoritmos GSP, Spade e PrefixSpan em 50% da base.

Figura 6. Qualidade da extração dos algoritmos GSP, Spade e PrefixSpan em 50% da base.

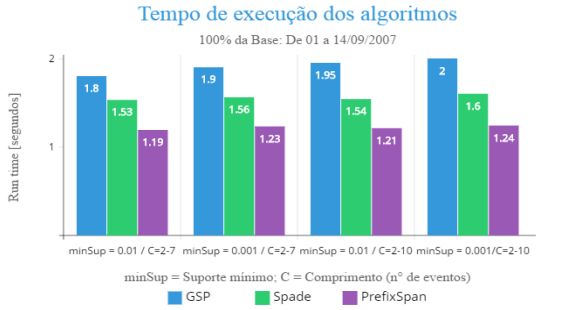

Figura 7. Tempo de execução dos algoritmos GSP, Spade e PrefixSpan em 100% da base.

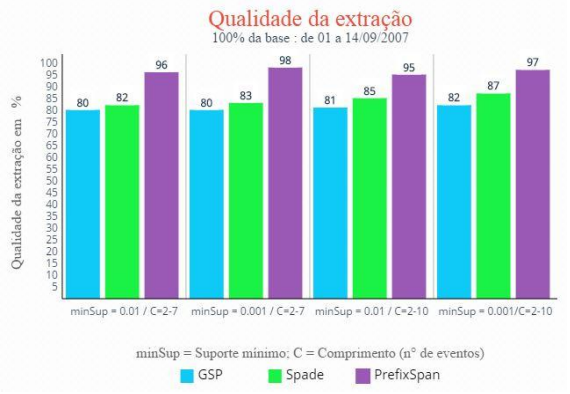

Figura 8. Qualidade da extração dos algoritmos GSP, Spade e PrefixSpan em 100% da base.

6 Discussões

Para o *dataset* utilizado neste trabalho, o melhor método para extração de padrões sequenciais de comportamento em casas inteligentes é o algoritmo PrefixSpan, pois o mesmo apresentou: melhores tempos de execução, além de ter obtido a melhor qualidade de extração (independentemente do valor do suporte mínimo e do comprimento utilizado).

O algoritmo Spade utiliza propriedades combinatórias para decompor o problema original em subproblemas menores, que podem ser resolvidos de forma independente. Durante os testes, o Spade obteve resultados medianos: apresentou tempos de execução intermediários, e qualidade de extração média de 85%.

Por ser baseado no algoritmo *Apriori*, possuir etapas de geração e contagem de candidatos e poda baseada no suporte, o GSP apresentou pior desempenho nos testes com os piores tempos de execução. Com relação à qualidade de extração, pode-se dizer que o GSP teve bons resultados, pois o mesmo apresentou qualidade de extração média de 80% no *dataset* utilizado neste trabalho.

A quantidade de dados analisada influencia nos resultados dos algoritmos. Quando poucos dados foram analisados, o sistema aprendeu pouco sobre o comportamento do habitante. Com 109 eventos (50% da base), já foi possível extrair bem o comportamento do habitante.

Após os testes, foi gerado o arquivo activities.CSV para identificar quais foram as sequências de ações que os algoritmos não conseguiram identificar e extrair corretamente. O algoritmo GSP teve problemas para identificar e extrair as ações que ocorreram no banheiro (ações que utilizavam o sensor de temperatura e o sensor de proximidade). Os sensores de temperatura e de proximidade instalados no *PlaceLab*, são de alta precisão e apresentam valores contínuos que afetaram a qualidade de extração do GSP pois, o mesmo foi projetado para tratar valores numéricos. O Spade também teve problemas para identificar e extrair as ações que ocorreram no banheiro, porém ao contrário do GSP, ele apenas teve problemas com o sensor de temperatura, ou seja, conseguiu identificar e extrair as ações que utilizaram o sensor de proximidade, pois, apresentou uma precisão maior do que o GSP. O PrefixSpan só teve problemas para identificar e extrair poucas ações que ocorreram na cozinha (utilizando o sensor de mudança de estado que utiliza valores binários).

7 Conclusão

Sistemas de automação residencial devem ter como objetivo principal tornar mais simples a interação entre os habitantes e os recursos da residência gerenciados por ela.

Este trabalho consistiu na análise de algoritmos em um sistema de extração de padrões sequenciais de

comportamentos. Um sistema deste tipo possibilita que a casa inteligente aprenda de forma autônoma o comportamento de seus habitantes.

Para os testes do sistema proposto, uma base já existente de dados reais foi utilizada, pois, a mesma satisfaz as características de uma residência real.

Foram testados diversos parâmetros de configurações sobre o *dataset* utilizado com o objetivo de simular situações que poderiam ocorrer em situações reais. O sistema se comportou conforme o esperado, extraindo os padrões sequenciais de comportamento do habitante, apresentando resultados diferentes para cada um dos algoritmos utilizados.

A variação do suporte mínimo e do comprimento, aplicados nos testes, mostrou que estes possibilitam aumentar ou reduzir a quantidade e qualidade das extrações. Foi verificado que os parâmetros ótimos no sistema dependem das características do *dataset* testado, ou seja, para a implantação do sistema em residências reais, seria necessário conhecer os hábitos dos moradores, a fim de se calibrar as variáveis do sistema.

Foi verificado que ajustes dos valores de suporte mínimo e comprimentos utilizados permitem obter uma otimização nos resultados. Neste trabalho, não foi implementada uma política de ajustes automáticos destes parâmetros, mas há indícios que outros dados presentes no *software* de extração poderiam ser utilizados para ajustar automaticamente parâmetros do sistema, de forma a otimizar seus resultados.

Todos os dados utilizados nos testes foram simulados. Desta forma, particularidades que poderiam ser observadas em testes efetuados em residências reais não puderam ser verificadas no sistema.

Foi utilizado um *dataset* único, nos testes, com apenas um habitante. Para ser possível uma avaliação melhor do sistema, seria interessante a realização de testes em outros *datasets*, com dois ou mais habitantes, e a execução de novos testes para observar modificações no comportamento das extrações e dos algoritmos.

Depois de avaliados os resultados obtidos, novos trabalhos podem ser desenvolvidos, com o intuito de aperfeiçoar o sistema e resolver seus pontos negativos.

Neste trabalho, foi observado o comportamento dos algoritmos utilizando-se um *dataset* único, contendo as ações de um habitante de uma casa inteligente por um período de catorze dias. Foi verificado, nos testes onde os valores dos parâmetros (suporte mínimo e comprimento) foram modificados, que os resultados das extrações são alterados conforme o número de eventos (ações) ocorridos.

Como trabalho futuro, seria interessante realizar testes utilizando outros *datasets*, alterando a quantidade de habitantes, as quantidades de ações de cada habitante e seus comportamentos, avaliando agora as alterações nas extrações dos padrões sequenciais conforme o *dataset* utilizado.

Também seria importante verificar comportamentos particulares do sistema, em situações específicas, que não foram geradas pelo *software*, que cria as situações a partir de parâmetros médios informados. Por exemplo, como o sistema se comportará quando houver a presença de visitantes na residência.

Referências Bibliográficas

AGRAWAL, R. and Srikant, R. “Fast algorithms for mining association rules.” Proc. 20th Int. Conf. Very Large Data Bases, VLDB, 1994, Volume 1215, pp. 487-499.

AGRAWAL, R. and Srikant, R. Mining sequential patterns: Generalizations and performance improvements. Proceedings of the 5th International Conference on Extending Database Technology, pages 3_17, 1996.

BOLZANI, Caio Augustus Morais. **Análise de arquiteturas e desenvolvimento de uma plataforma para residências inteligentes**. Tese de Doutorado em Engenharia Elétrica. Universidade de São Paulo, 2010.

FUSSEK, R. A. **Desenvolvimento de um classificador automático de pessoas para sistemas de automação residencial inteligente**. 2010. Dissertação (Mestrado em Engenharia Elétrica) – Centro Universitário da FEI, São Bernardo do Campo.

HOPPNER, F., 2002. “Finding informative rules in interval sequences,” *Intelligent Data Analysis*, vol.6, no. 3, pp. 237-255.

JUNESTRAND *et al.* Private and public digital domestic spaces. In: **International Journal of Human-Computer Studies.** 2001, Stockholm, Sweden, p. 753-778, 2001.

KING, N. “Smart home. A definition” Intertek Research and Testing Center, pp. 1-6, 2003.

MIT, 2016. Disponível em: <http://web.mit.edu/cron/group/house_n/intro.html>. Acesso em: 15 de dez. 2016.

PEI, J et al. Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth. In: proceedings of the 17th international conference on data engineering. 2001. p. 215-224.

YAMAZAKI, T. Beyond the Smart Home. **International Conference on Hybrid Information Technology (ICHIT'06)**, IEEE, 2006.

ZAKI, M. MACHINE LEARNING: An Efficient Algorithm for Mining Frequent Sequences. January 2001, Volume 42, Issue 1, p. 31–60.