UNISANTOS
UNIVERSIDADE CATÓLICA DE SANTOS

FERNANDO KAUFFMANN BARBOSA

UM AGENTE FACILITADOR DA NAVEGAÇÃO NA WEB

SANTOS
2007

FERNANDO KAUFFMANN BARBOSA

UM AGENTE FACILITADOR DA NAVEGAÇÃO NA WEB

Dissertação apresentada à Banca Examinadora do Programa de Pós-Graduação em Informática da Universidade Católica de Santos, como exigência parcial para a obtenção do título de Mestre em Informática, sob a orientação da Profa. Dra. Marta Costa Rosatelli.

SANTOS

2007

UM AGENTE FACILITADOR DA NAVEGAÇÃO NA WEB

FERNANDO KAUFFMANN BARBOSA

Esta dissertação foi julgada adequada para a obtenção do título de Mestre em Informática área de concentração em Ciência da Computação e aprovado em sua forma final pelo Programa de Mestrado em Informática.

DATA DE APROVAÇÃO: ____ / ____ / _____

Profª. Drª Marta Costa Rosatelli
Coordenadora do Curso

BANCA EXAMINADORA:

Profª. Drª Marta Costa Rosatelli
Orientadora - Membro Nato

Prof. Dr. Eduardo Raul Hruschka
Membro Titular

Prof. Dr. Juan Manuel Adán Coello
Membro Titular

RESUMO

Esta dissertação apresenta um agente, o FNA (*Fast Navigation Agent*), que tem como objetivo aumentar a facilidade e rapidez de acesso às páginas *Web* de maior interesse de um visitante, em *sites* que possuem uma grande quantidade de informações e serviços. O FNA faz parte do conteúdo das páginas *Web* do *site*, interage com o visitante e traça o seu perfil de navegação. O FNA utiliza técnicas de mineração de uso na *Web* e um algoritmo de Regras de Associação.

Palavras-chave: Agentes, Mineração de dados na *Web*, Regras de associação

ABSTRACT

This dissertation presents an agent that has as its main goal increasing the easiness and speed of access to Web pages that are of interest to visitors of sites that have a great amount of information and services. The agent, named Fast Navigation Agent (FNA), is part of a Web page content. It, interacts with the site's visitor and identify his or her navigation profile. In order to accomplish this it makes use of the Web Usage Mining technique and an Association Rules algorithm.

Keywords: Agents, Web mining, Association rules

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE ABREVIATURAS E SIGLAS

FNA	*Fast Navigation Agent*
FNAc	*Fast Navigation Agent* - Coletor
FNAm	*Fast Navigation Agent* - Minerador
IP	*Internet Protocol*
SQL	*Structured Query Language*
URL	*Uniform Resource Locator*
WWW	*World Wide Web*

SUMÁRIO

1 INTRODUÇÃO

A evolução e revolução tecnológica no ambiente *Web* resultam em um grande aumento de vários tipos de informações e serviços disponíveis e/ou oferecidos aos usuários da *Internet*.

O comportamento de pessoas que navegam na *Internet* é de grande interesse para os pesquisadores que trabalham com sistemas baseados na *Web*. Em particular, em Sistemas Inteligentes, uma das áreas de aplicação mais relevantes tem por objetivo descobrir padrões de navegação dos visitantes e a partir deste fazer a personalização dos *sites*, melhorar a navegabilidade do usuário nos mesmos, etc.

Os sistemas que fazem a personalização dos *sites* a partir de um perfil do usuário podem ser caracterizados como sistemas de Hipermídia Adaptativa (ROSATELLI; TEDESCO, 2003). Estes têm a finalidade de oferecer informações e *hyperlinks* adaptados e personalizados de acordo com o perfil do visitante no *site* ou dar apoio ao usuário oferecendo *hyperlinks* de maior interesse baseados em suas navegações sem realizar adaptações nas páginas *Web* do *site*, utilizando a técnica de Suporte à Navegação Adaptativa com Orientação Direta a *Links* Não Contextuais (BRUSILOVSKY, 1996, 2001).

Segundo Zukerman e Albrecht (2001) a determinação do perfil do visitante em um *site* pode ser baseada em duas abordagens:

a) Aprendizagem Baseada em Conteúdo: o comportamento do visitante em suas visitas passadas define o seu comportamento futuro no *site*;
b) Aprendizagem Colaborativa: o comportamento do visitante se assemelha com o comportamento de um grupo de visitantes, podendo fazer previsões de seu comportamento baseado nesse grupo.

Ainda que várias técnicas possam ser utilizadas, deve-se destacar que a determinação do perfil do visitante requer uma grande quantidade de informações armazenadas e disponíveis à aplicação de técnicas de mineração de dados na *Web*

(*Web Mining*)(CHANG *et al.*, 2001), com o objetivo de descobrir padrões nelas existentes.

Quanto às utilizações da mineração de dados na *Web* podem ser destacadas:

a) Mineração do Conteúdo na *Web*: descoberta automática de padrões nos conteúdos dos documentos *Web*;
b) Mineração do Uso na *Web*: descoberta automática de padrões de acesso ao servidor *Web*;
c) Mineração da Estrutura na *Web*: descoberta automática de padrões de estrutura de hipertexto e/ou *hyperlinks*.

Segundo Cooley, Mobasher e Srivastava (1997), o uso de agentes *Web* tem sido aplicado no campo de mineração de dados na *Web* com o objetivo de fornecer uma organização melhor dos dados semi-estruturados, tais como as páginas *Web* e os padrões de interesse nas navegações em um *site*.

Nesse contexto, Pazzani e Billsus (1999) utilizam agentes como auxiliadores na navegação de *sites*, nos quais o visitante recebe recomendações de páginas *Web* relacionadas à página *Web* visitada.

Russell e Norvig (1995) afirmam que "um agente é tudo o que pode ser considerado capaz de perceber seu ambiente por meio de sensores e de agir sobre esse ambiente por intermédio de atuadores". Para Lieberman (1997), agentes são programas que atuam como assistentes ou ajudantes de usuários na interação com os sistemas.

Dessa forma, vale lembrar que os agentes apresentam uma ou mais das seguintes propriedades (FRANKLIN; GRAESSER, 1996):

a) Autonomia: operam sem a necessidade de serem guiados por humanos e têm certo controle sobre suas ações;
b) Mobilidade: têm a capacidade de poder se mover através de uma rede de computadores;
c) Cooperação: podem trabalhar em conjunto de forma a concluírem tarefas de interesse comum;

d) Comunicabilidade: são capazes de se comunicar com outros agentes ou pessoas;
e) Aprendizagem: possuem habilidades de avaliar as variações de seu ambiente externo e escolher a ação mais correta;
f) Reatividade: reagem a mudanças no seu ambiente;
g) Pró-atividade: não reagem simplesmente em resposta ao ambiente, mas são capazes de exibir um comportamento baseado em metas, tomando a iniciativa.

Dentro desse cenário, esta dissertação apresenta o *Fast Navigation Agent* (FNA), que é um assistente com características de autonomia, mobilidade, comunicabilidade, reatividade e pró-atividade. O FNA tem a finalidade de auxiliar os visitantes na navegação dentro de um *site* exibindo *hyperlinks* personalizados das páginas *Web* resultantes da mineração do uso da *Web*.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

Esta dissertação tem por objetivo desenvolver um agente, FNA (*Fast Navigation Agent*), destinado a aumentar a facilidade e rapidez de acesso às páginas *Web* de maior interesse do visitante, em *sites* que possuem uma grande quantidade de informações e serviços.

1.1.2 Objetivos Específicos

Os objetivos específicos desta dissertação são:

a) Fazer uma revisão da literatura sobre mineração de dados na *Web*;

b) Identificar um método de mineração de dados na *Web* adequado;
c) Especificar e modelar o agente;
d) Demonstrar a viabilidade da aplicação através do desenvolvimento do agente;
e) Testar o agente desenvolvido;
f) Redigir o trabalho.

1.2 JUSTIFICATIVA

A obtenção de informações sobre interesses de um visitante em *sites* que apresentam uma grande quantidade de páginas *Web* e serviços pode ser uma tarefa frustrante e cansativa. Frustrante pelo fato do visitante navegar pelo *site*, tendo a sensação de estar perdido, ou não encontrar o que deseja. Quando o visitante atinge seu objetivo a navegação futura muitas vezes poderá tornar-se cansativa, pois ele deverá percorrer todo o eventualmente longo caminho até chegar a seu destino.

Além disso, freqüentemente os *sites* fazem atualizações em seu conteúdo e sua estrutura. Quando isso ocorre, o visitante que estava acostumado a percorrer um determinado caminho de navegação tem que gastar novamente algum tempo para descobrir o novo caminho que o conduzirá às páginas *Web* que são do seu interesse.

Na medida do possível deve-se evitar que o visitante tenha a sensação de estar perdido dentro de um *site*. Para isso, o usuário precisa saber responder a três questões. São elas: “De qual página *Web* eu vim?”, “Em que página *Web* estou?” e “Para onde eu posso ir?”. Respondendo a essas perguntas, o visitante saberá sua localização em um *site*.

Compreende-se, assim, que facilitar a navegação do visitante, ou seja, chegar em seu destino rapidamente, representa uma necessidade. Dessa forma, o FNA auxilia o visitante em sua passagem por um *site*, exibindo os *hyperlinks* relevantes baseados em seu perfil.

É importante destacar que o FNA está presente no conteúdo da página *Web*, não havendo a necessidade de criar uma nova. Por todas as características

apresentadas, o agente desenvolvido oferece maior facilidade ao acesso através dos *hyperlinks* personalizados fazendo a verificação de atualizações no conteúdo das páginas *Web* e atuando a partir do interesse do visitante.

1.3 METODOLOGIA

Inicialmente, será feita uma revisão bibliográfica enfocando os conceitos e métodos de mineração de dados na *Web*. Essa revisão deve embasar a identificação do método de mineração de dados na *Web*. A partir daí será feita a especificação, modelagem e implementação do agente.

A parte do agente que diz respeito ao método de mineração de dados na *Web* será implementado na liguagem JAVA. A outra parte, que diz respeito à interação do agente com os visitantes será implementada nas linguagens HTML, PHP e JavaScript. As informações de navegabilidade serão coletadas e armazenadas em um banco de dados MySql para futuras minerações.

Os testes com o agente serão realizados através de um *site* construído para esse fim. Os resultados desses testes permitirão analisar a interação do visitante com o agente e o suporte a navegação.

1.4 ESTRUTURA DO TRABALHO

O presente trabalho é composto de cinco capítulos.

O capítulo 2 apresenta os conceitos de mineração de dados na *Web* (*Web Mining*) e suas abordagens e, também, as tarefas utilizadas na mineração de dados na *Web*.

O capítulo 3 explana o conceito e a aplicação da tarefa de Regras de Associação e exemplifica a sua aplicação dentro do contexto da *Web*.

O agente *Fast Navigation Agent* (FNA) é descrito no capítulo 4, que mostra sua arquitetura, funcionalidade e aplicação, unindo os conceitos de mineração de uso da *Web* e Regras de Associação vistos nos capítulos anteriores.

Finalmente, o capítulo 5 apresenta as considerações finais ilustrando alguns resultados obtidos a partir do FNA, além de oferecer possíveis direções para trabalhos futuros.

2 MINERAÇÃO DE DADOS NA *WEB*

2.1 INTRODUÇÃO

Este capítulo aborda a mineração de dados na *Web*, que inclui a mineração do conteúdo da *Web*, mineração do uso da *Web* e mineração da estrutura da *Web*, suas respectivas definições e aplicações. Em seguida, são apresentadas e definidas as principais tarefas de mineração de dados para descoberta de padrões no ambiente *Web*.

2.2 DA FONTE DE DADOS À DESCOBERTA DE PADRÕES

Servidores *Web* armazenam uma grande variedade de tipos de documentos. Esses tipos podem ser imagens, vídeos, animações, textos e áudios, inseridos em páginas que são exibidas para os usuários em suas visitas aos *sites* na *Internet*.

Durante as visitas dos usuários a um *site*, os servidores *Web* registram seus acessos às páginas *Web* em arquivos de registros de acesso ou em banco de dados, dependendo de como esses servidores são configurados. Os registros de acesso contêm as informações sobre a navegação do usuário no *site* (figura 1), tais como, endereços IP, datas e horas de acesso, páginas *Web* ou tipos de documentos solicitados, navegadores utilizados pelos visitantes e etc.

ClientIP	Time	URL	Referrer	Agent
201.64.62.74	2006-11-18 17:13:28	http://www.fna.kit.net/a.html	-	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 2.0.50727)
201.64.62.74	2006-11-18 17:16:26	http://www.fna.kit.net/b.htm	http://www.fna.kit.net/a.html	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 2.0.50727)
201.64.62.74	2006-11-18 17:17:03	http://www.fna.kit.net/c.htm	http://www.fna.kit.net/b.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 2.0.50727)
201.64.62.74	2006-11-18 17:17:34	http://www.fna.kit.net/d.htm	http://www.fna.kit.net/c.htm	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 2.0.50727)

Figura 1 - Exemplo de Registros de Acesso a um Servidor *Web*

No quadro 1 é descrito cada tipo de informação obtida a partir da navegação do visitante em um *site* ilustrado na figura 1.

Campos	Descrição
ClientIP	Endereço IP do visitante
Time	Data e hora do acesso à página *Web*
URL	Página *Web* acessada;
Referrer	Página *Web* referenciadora da página *Web* acessada
Agent	Navegador utilizado pelo visitante

Quadro 1 - Descrição dos campos do registro de acesso ao servidor Web

Com as constantes visitas que um *site* recebe de vários usuários, o número de registros de acesso tende a aumentar e, como as informações dentro do *site* são dinâmicas, ocorrem atualizações e inserções de novas páginas *Web* crescendo a quantidade de documentos dentro dos servidores *Web*.

Devido a esse crescimento significativo, tanto nas páginas *Web* como nos registros de acesso, podem-se obter informações pertinentes relacionadas a cada usuário que visita o *site*. A obtenção dessa informação é feita através da aplicação de mineração de dados na *Web* (*Web Mining*).

Vale destacar que o primeiro a sugerir o termo *Web Mining* foi Etzioni (1996) ao apontar “[...] o uso de técnicas de mineração de dados para descobrir e extrair informação automaticamente de documentos e serviços da WWW”.

Para Cooley, Mobasher e Srivastava (1997, p. 558)

> *Web Mining* pode ser amplamente definido como a descoberta e análises de informações úteis a partir da *World Wide Web*. Isto descreve a busca automática de informações de recursos disponíveis, isto é, Mineração do Conteúdo na *Web*, e descoberta de padrões de acesso dos usuários a partir de servidores *Web*, isto é, Mineração do Uso na *Web*.

Segundo essa afirmação, a mineração de dados na *Web* é classificada em dois segmentos: Mineração do Conteúdo na *Web* e Mineração do Uso na *Web* (figura 2).

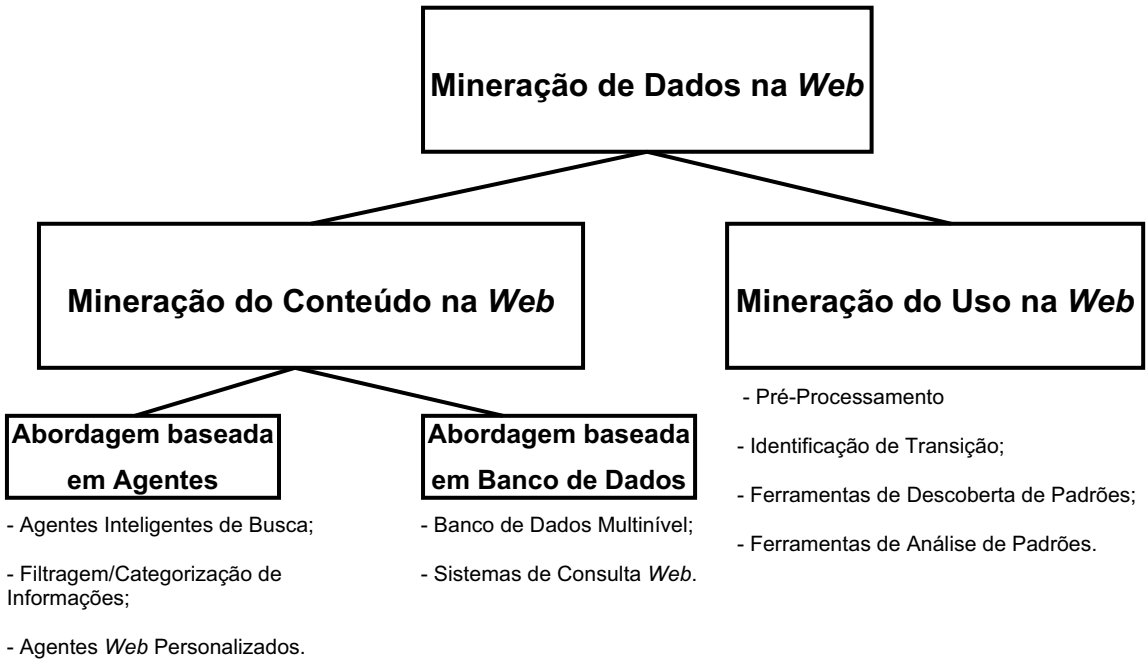

Figura 2- Taxonomia da Mineração de Dados da *Web*

Fonte: COOLEY, MOBASHER e SRIVASTAVA, 1997

E Zaiane (2000) acrescenta,

> As estruturas de *hyperlink* contêm uma vasta quantidade de informação escondida que pode ser alvo de mineração. Mineração da Estrutura na *Web* constitui-se no processo de extração de conhecimento a partir das interconexões de hipertextos de documentos da *Web*.

A seguir são definidos os tipos de mineração de dados na *Web* mostrando suas funções e aplicações.

2.3 MINERAÇÃO DO CONTEÚDO NA WEB

A Mineração do Conteúdo na *Web* tem a finalidade de fazer a descoberta do conhecimento a partir dos conteúdos, dados, documentos e serviços *Web*. Fazer mineração de conteúdo não é tarefa fácil, pois as fontes de informação na *Web* apresentam uma forma semi-estruturada (COOLEY; MOBASHER e SRIVASTAVA, 1997).

A figura 3 ilustra o código fonte de uma página *Web* e nela pode-se observar a estrutura da página através das *tags* representadas por <...>. Nota-se também que as informações não estruturadas nas linhas 1 a 4 apresentam diversos tipos de dados. Por exemplo, na linha 1 há uma imagem e na linha 4 temos um dado tipo texto (Telefone:) e outro tipo numérico (13123412345). Assim, a *Web* não possui controle ou uma definição sobre a estrutura ou o tipo de dados e/ou documentos armazenados nos servidores *Web*.

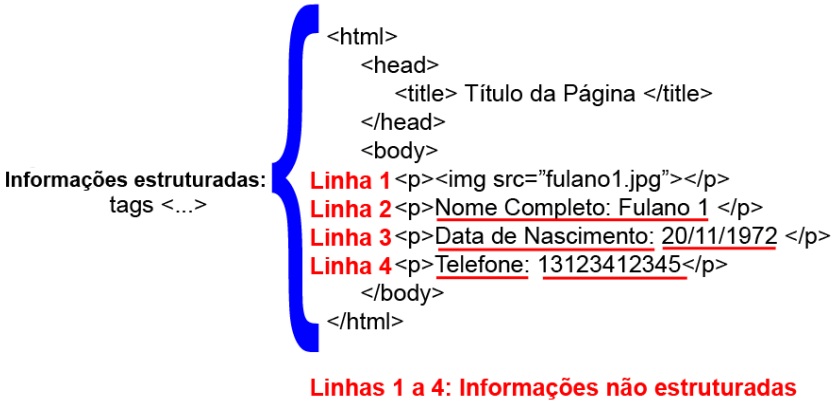

Figura 3 - Exemplo de origem da informação semi-estruturada

Quanto à obtenção e organização das informações, Cooley, Mobasher e Srivastava (1997) citam duas importantes técnicas:

a) O uso de agentes que envolvem o desenvolvimento de sistemas sofisticados atuando de forma automática para um determinado fim;

- Agentes inteligentes de busca se referem a um grupo de sistemas que buscam informações relevantes utilizando características de um determinado domínio;
- A filtragem e categorização de informações podem também ser realizadas automaticamente por agentes que observam a estrutura das interligações de páginas *Web* e seus conteúdos para criar agrupamentos hierárquicos de documentos;
- Agentes personalizados obtêm e analisam a preferência de um visitante e buscam fontes de informações na *Web* que satisfaçam essa preferência, ou ainda, de visitantes com interesses semelhantes.

b) Abordagens baseadas em banco de dados visam integrar e organizar o conteúdo heterogêneo e semi-estruturado dos dados da *Web* em coleções de dados mais estruturadas que possam disponibilizar uma maior quantidade de recursos. As seguintes técnicas são apresentadas:

- Bancos de Dados Multinível armazenam em seu nível mais baixo informações semi-estruturadas presentes em vários bancos de dados *Web*, tais como, documentos hipertexto. No nível mais alto são armazenados os meta dados obtidos do nível mais baixo, mantendo-os em coleções estruturadas de dados, de forma relacional ou orientada a objetos, por exemplo;
- Sistemas de Consulta *Web* utilizam a linguagem de consulta a banco de dados, por exemplo, a linguagem SQL, que faz a estruturação das informações sobre os documentos *Web* e, até mesmo, processamento de linguagem natural para a busca de informações.

2.4 MINERAÇÃO DO USO NA WEB

Na Mineração de Uso na *Web* é feita a descoberta de perfis ou padrões de acesso dos visitantes em um *site* a partir de arquivos de registro de acesso do servidor *Web* (COOLEY, MOBASHER, SRIVASTAVA, 1997).

De acordo com Ebecken, Lopes e Costa (2003, p. 368),

> Os registros são analisados de forma a apresentar uma tendência de acesso, que pode levar a uma organização do *site* apropriada a um acesso mais rápido e eficiente. Podem-se utilizar procedimentos de mineração de dados nos arquivos de acesso a páginas, depois de devidamente pré-processados.

Os sistemas e técnicas para a descoberta e análise de padrões podem ser classificados em dois grupos, são eles (COOLEY, MOBASHER e SRIVASTAVA, 1997):

a) Ferramentas de Descoberta de Padrões usam técnicas de inteligência artificial, mineração de dados e estatística para minerar conhecimento a partir de dados coletados;
b) Ferramentas de Análise de Padrões são usadas para entender, visualizar e interpretar os padrões de acesso descobertos a partir dos registros de acesso.

A Mineração do Uso na *Web* envolve três tarefas principais, conforme mostra a figura 4 (SRIVASTAVA *et al*., 2000), a seguir:

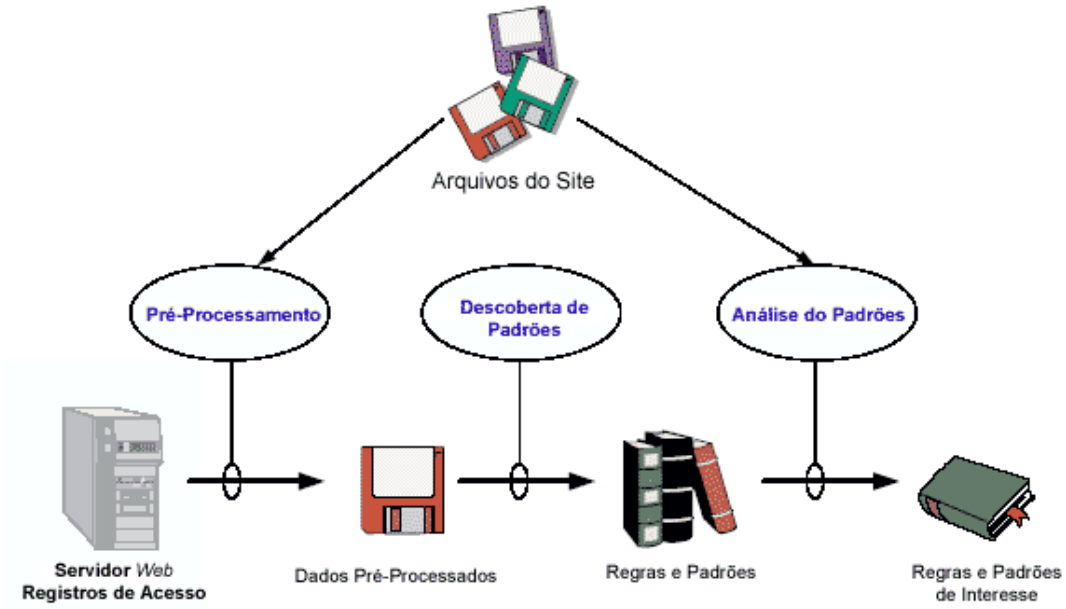

Figura 4 - Visão Geral dos Processos da Mineração do Uso na *Web*
Fonte: SRIVASTAVA et al., 2000, p. 15

O esquema anterior mostra que, a partir dos arquivos do *site* armazenados em um servidor *Web* e acessados pelos seus visitantes são gerados registros de acesso desses arquivos. A fase do pré-processamento tem o objetivo de fazer a limpeza dos dados, ou seja, obter dados relevantes dos registros de acesso dos visitantes no *site* para futura análise.

Sobre os dados pré-processados, são aplicadas tarefas e técnicas de mineração de dados, tais como, regras de associação, classificação, agrupamento e padrões de seqüências entre outros, com o objetivo de descobrir padrões pertinentes sobre os registros de acesso pré-processados. Por fim é realizada a análise dos padrões descobertos, fazendo uma filtragem nos padrões e eliminando regras ou padrões de menor importância.

Após a execução desses processos, podem-se determinar perfis de visitantes dentro de um *site*, por exemplo, saber quais páginas *Web* são de interesse para um determinado visitante dentro de um *site* com base em sua navegação pelo mesmo.

2.5 MINERAÇÃO DA ESTRUTURA NA WEB

A navegação dos visitantes em um *site* é feita através de *hyperlinks* que interconectam várias páginas *Web* (navegação entre páginas) ou *hyperlinks* que permitem uma navegação na mesma página *Web* (figura 5). Esse conjunto de *hyperlinks* gera a estrutura do *site*.

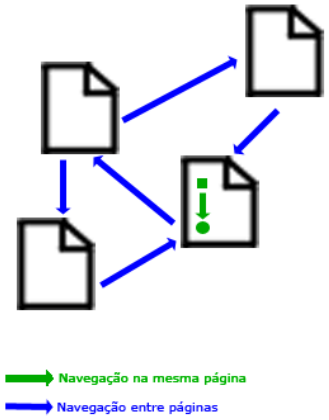

Figura 5 - Exemplo de navegação

Na Mineração da Estrutura na *Web* são aplicadas técnicas para calcular a qualidade ou relevância das páginas da *Web*. Uma das regras utilizadas é que quanto mais páginas *Web* estiverem apontando para uma determinada página *Web*, mais relevante ela será.

2.6 TAREFAS DE MINERAÇÃO DE DADOS NA WEB

Na mineração de dados na *Web* são adaptadas as tarefas de mineração de dados utilizados em bancos de dados tradicionais (SRIVASTAVA et al., 2000).

Para Fayyad, Shapiro e Smyth (1996a, 1996b), as principais tarefas de mineração de dados são:

a) Classificação: aprendizado de uma função que mapeia um registro em uma das várias classes conhecidas;
b) Regressão: aprendizado de uma função que mapeia um registro em um valor real previsto;
c) Agrupamento: identificação de grupos de registros com características semelhantes entre si;
d) Sumarização: fornece uma descrição compacta para um subconjunto de dados;
e) Modelagem de Dependência: descreve dependências significantes entre variáveis.
f) Análise de Ligação: determina relações entre dados em uma base de dados gerando regras de associação;
g) Análise de Seqüências: modela padrões de seqüências em que a ordem dos dados apresenta uma significância.

A seguir são apresentados trabalhos de pesquisa em que utilizam algumas das tarefas de mineração de dados no ambiente *Web*.

2.7 TRABALHOS RELACIONADOS

Alguns trabalhos de pesquisa apresentam abordagens para a aplicação da mineração de dados no ambiente da *Web* que resultaram na construção de ferramentas para esse fim. Dentre estes destacam-se:

a) *Access Miner* (BRUSSO, 2000): aplicação de regras de associação sobre o banco de dados de registros de acesso do servidor *Web* para determinar correlações entre as páginas *Web* visitadas e descobrir padrões de navegação em um *site*;
b) *PagePrompter* (YAO, HAMILTON e WANG, 2002): utiliza um agente inteligente para ajudar o visitante em sua navegação pelo *site*. Esse agente descobre regras de associação e grupos de páginas *Web* a partir do arquivo de registro do servidor *Web*;

c) *WEBMINER* (COOLEY, MOBASHER e SRIVASTAVA, 1997): descobre regras de associação e padrões seqüenciais a partir de registros de acesso ao servidor *Web*.

2.8 CONCLUSÃO

A *Web* é um grande repositório de dados e informações semi-estruturadas podendo ser acessada por vários visitantes através de diversos *sites*. Os acessos e a obtenção de informações são realizados de uma maneira imprevisível e não uniforme. Com isso há a necessidade de se desenvolver sistemas com o objetivo de auxiliar tanto os visitantes, como os proprietários dos *sites*. Uma das maneiras de auxiliá-los é aplicando a mineração de dados na *Web*, seja para ajudar os desenvolvedores de *sites* a entender melhor o comportamento navegacional dos visitantes, seja para ajudar os visitantes a alcançar com maior precisão as informações que buscam.

Dentro do contexto de mineração de dados na *Web*, o próximo capítulo mostra o uso das regras de associação bem como sua aplicação no ambiente de mineração do uso na *Web*.

3 REGRAS DE ASSOCIAÇÃO

3.1 INTRODUÇÃO

O presente capítulo apresenta a definição e demonstra a aplicação das regras de associação dentro do ambiente *Web*, com o objetivo de descobrir relações entre páginas *Web* acessadas por um visitante em um *site*, utilizando o algoritmo Apriori. Em seguida, é exemplificada, passo a passo, a utilização desse algoritmo.

3.2 VISÃO GERAL DAS REGRAS DE ASSOCIAÇÃO

A cada acesso de um visitante a um *site* é estabelecida uma sessão. Em uma sessão o visitante pode acessar uma ou mais páginas *Web*. Ao navegar pelas várias páginas *Web* em um *site* é deixado um “rastro digital” nos servidores *Web*, que armazenam toda a trajetória do visitante no *site*.

Segundo Han e Kamber (2001) "a tarefa de regras de associação descobre correlações de interesse em um grande conjunto de elementos de dados".

Logo, no contexto de mineração de dados na *Web* pode-se aplicar a tarefa de regras de associação para descobrir relações significativas entre páginas *Web* nas diversas sessões de um visitante.

Por exemplo, ao aplicar regras de associação em um conjunto de sessões de um visitante, as seguintes regras poderiam ser obtidas:

a) O visitante X que acessou a página A.html tem a chance de 75% de acessar a página B.html;
b) O visitante X que acessou a página A.html tem a chance de 20% de acessar a página C.html.

As regras desse exemplo mostram que quando o visitante X acessar a página A.html ele tem maior chance de visitar a página B.html do que a página C.html.

Vale destacar que existem vários algoritmos de regras de associação, tais como, Apriori, AprioriTID, AIS, SETM (AGRAWAL; SRIKANT, 1994). Dentre esses algoritmos, será utilizado, neste trabalho, o algoritmo Apriori por apresentar melhor desempenho no tempo de execução na obtenção de elementos freqüentes em um conjunto de dados, segundo Agrawal e Srikant (1994).

3.3 DEFINIÇÃO FORMAL DAS REGRAS DE ASSOCIAÇÃO

Dentro do contexto de mineração de dados na *Web* e baseado na definição formal das Regras de Associação (AGRAWAL; SRIKANT, 1994), dado um conjunto de páginas *Web* $\mathbf{P = \{p_1, p_2, p_3, ..., p_n\}}$ e um conjunto de sessões em uma Base de Dados **D**, em que cada sessão **S** é um conjunto de páginas *Web* visitadas em **P** tal que $\mathbf{S \subseteq P}$. Uma regra de associação é uma expressão do tipo $\mathbf{X \Rightarrow Y}$, na qual $\mathbf{X \subset P}$, $\mathbf{Y \subset P}$ e os conjuntos **X** e **Y** não possuem páginas *Web* em comum. **X** é denominado **antecedente** e **Y** é denominado **conseqüente** da regra. Tanto o antecedente quanto o conseqüente de uma regra de associação podem ser formados por conjuntos contendo uma ou mais páginas *Web*. A quantidade de páginas *Web* pertencentes a um conjunto é chamada de cardinalidade do conjunto. Um conjunto de cardinalidade **k** costuma ser referenciado como um ***k-itemset***.

O suporte de um conjunto de páginas *Web* **W**, **Suporte(W)**, representa a porcentagem de sessões da base de dados **D** que contêm as páginas *Web* de **W**.

$$\mathbf{Suporte(W)} = \frac{\mathbf{freqüência\ em\ que\ W\ ocorre}}{\mathbf{número\ total\ de\ sessões\ em\ D}} \times \mathbf{100} \qquad (1)$$

O suporte de uma regra de associação $\mathbf{X \Rightarrow Y}$, **Suporte(X $\Rightarrow$ Y)**, é dado por $\mathbf{Suporte(X \cup Y)}$**.** Quanto à confiança desta regra, **Confiança(X $\Rightarrow$ Y)**, ela representa, dentre as sessões que contêm **X**, a porcentagem de sessões que também contêm **Y**.

$$\textbf{Confiança}(\mathbf{X \Rightarrow Y}) = \frac{\textbf{Suporte}(\mathbf{X \cup Y})}{\textbf{Suporte}(\mathbf{X})} \times \mathbf{100} \qquad (2)$$

Sendo assim, a mineração de regras de associação em bases de dados consiste em encontrar todas as regras que possuam suporte e confiança maiores ou iguais, respectivamente, a um suporte mínimo e uma confiança mínima.

O processo de descobrir todas as regras de associação, pode ser decomposto em duas etapas:

a) Encontrar todos os conjuntos de páginas *Web* freqüentes, ou seja, conjuntos com suporte maior ou igual ao suporte mínimo;
b) Utilizar as páginas *Web* freqüentes obtidas para gerar as regras de associação com confiança maior ou igual a confiança mínima.

3.4 ALGORITMO APRIORI

A figura 6 ilustra a primeira parte do algoritmo Apriori (AGRAWAL; SRIKANT, 1994). A primeira etapa do algoritmo determina o número de ocorrências para cada subconjunto de candidatos de cardinalidade igual a 1 resultando o conjunto $\mathbf{L_1}$.

```
1)   L1 = {large 1-itemsets};
2)   for (k = 2; Lk-1 ≠ ∅; k++) do begin
3)       Ck = apriori-gen(Lk-1); // novos candidatos
4)       forall sessões s ∈ D do begin
5)           Ct = subset(Ck, s); // Candidatos contidos em s
6)           forall candidatos c ∈ Ct do
7)               c.count++;
8)       end
9)       Lk = {c ∈ Ck | c.count ≥ minsup};
10)  end
11)  Resposta = ⋃k Lk ;
```

Figura 6 - Algoritmo Apriori

Fonte: AGRAWAL; SRIKANT, 1994, p.489

O suporte de cada um dos elementos dos subconjuntos que fazem parte do conjunto $\mathbf{L_1}$ é maior ou igual ao suporte mínimo pré-estabelecido.

No passo seguinte, inicia-se o processo de repetição para geração de novos candidatos $\mathbf{C_2}$ a partir de $\mathbf{L_1}$, resultando em $\mathbf{L_2}$ e assim por diante. O processo de repetição termina quando o $\mathbf{L_{k-1}}$ não possuir mais candidatos.

A geração dos candidatos $\mathbf{C_k}$ é feita aplicando o procedimento **apriori-gen** (figura 7) e passando como parâmetro o conjunto $\mathbf{L_{k-1}}$. Esse procedimento é composto por duas etapas. Na primeira é feito o produto cartesiano $\mathbf{L_{k-1}}$ **x** $\mathbf{L_{k-1}}$ gerando os subconjuntos de cardinalidade **k**, cujos elementos são ordenados de forma crescente.

```
insert into C_k
select p.item_1, p.item_2, ..., p.item_k-1, q.item_k-1
from L_k-1 p, L_k-1 q
where p.item_1 = q.item_1, ..., p.item_k-2 = q.item_k-2, p.item_k-1 < q.item_k-1;
```

Figura 7 - Primeira etapa do procedimento apriori-gen

Fonte: AGRAWAL; SRIKANT, 1994, p.489

Na segunda etapa (figura 8) é executado o processo de corte de todos os candidatos **c** pertencente a $\mathbf{C_k}$ tal que algum subconjunto **c** de tamanho **k-1** não está presente em $\mathbf{L_{k-1}}$.

```
forall itemsets c ∈ C_k do
    forall (k-1)-subsets s of c do
        if (s ∉ L_k-1) then
            delete c from C_k;
```

Figura 8 - Segunda etapa do procedimento apriori-gen

Fonte: AGRAWAL; SRIKANT, 1994, p.489

Após a criação dos candidatos $\mathbf{C_k}$, é realizada uma análise dos candidatos sobre o conjunto de todas as sessões **D** determinando o número de ocorrências de cada candidato no conjunto de todas sessões **D**.

Se o número de ocorrências do candidato for maior ou igual ao suporte mínimo, então, esse candidato fará parte do conjunto $\mathbf{L_k}$.

No fim dessa etapa encontra-se a unicidade de todos os conjuntos candidatos válidos em $\mathbf{L_k}$.

Dessa forma, o próximo passo do algoritmo é gerar as regras de associação a partir do conjunto de candidatos $\mathbf{L_k}$, com **k** maior ou igual a 2, aplicando o procedimento **genrules** (figura 9).

```
procedure genrules(l_k: large k-itemset, a_m: large m-itemset)
1)   A = {(m-1)-itemsets a_m-1 | a_m-1 ⊂ a_m}
2)   forall a_m-1 ∈ A do begin
3)       conf = support(l_k) / support(a_m-1);
4)       if (conf ≥ minconf) then begin
5)           output the rule a_m-1 ⇒ (l_k - a_m-1),
                        with confidence = conf and
                         support = support(l_k);
6)           if (m - 1 > 1) then
7)               call genrules(l_k, a_m-1)
8)       end
9)   end
```

Figura 9 - Procedimento genrules para geração de regras

Fonte: AGRAWAL; SRIKANT, 1994, p.489

O procedimento **genrules** (figura 9) recebe o conjunto freqüente $\mathbf{L_k}$ por duas vezes, a primeira como $\mathbf{l_k}$ e a segunda como $\mathbf{a_m}$, conjunto freqüente com **m** elementos, no qual inicialmente **m** é igual **k**, e $\mathbf{l_k}$ é igual a $\mathbf{a_m}$.

Em seguida, encontram-se todos os subconjuntos não nulos de $\mathbf{a_m}$ com $\mathbf{a_{m-1}}$ elementos. Esses subconjuntos são armazenados em um conjunto chamado de **A**. Para cada subconjunto $\mathbf{a_{m-1}}$ de $\mathbf{a_m}$ pertencente a **A**, escreve-se a regra da forma $\mathbf{a_{m-1} \Rightarrow (l_k - a_{m-1})}$, se a razão $\mathbf{conf = support(l_k) / support(a_{m-1})}$ for maior ou igual a confiança mínima. Consideram-se todos os subconjuntos de $\mathbf{a_m}$ para gerar as regras com múltiplos tamanhos.

Assim, o algoritmo divide o conjunto freqüente $\mathbf{a_m}$ em vários subconjuntos de **m-1** elementos em busca das regras válidas.

No próximo passo, o algoritimo testa se o valor de **m-1** é maior que 1, e, sendo a condição verdadeira chama-se outra vez a função **genrules**, num processo recursivo, passando o conjunto $\mathbf{l_k}$ e o conjunto $\mathbf{a_{m-1}}$.

Novamente, a função **genrules** recebe o conjunto $\mathbf{l_k}$ e o conjunto $\mathbf{a_{m-1}}$ como $\mathbf{a_m}$, e repete todo o processo até que o conjunto $\mathbf{a_{m-1}}$ contenha somente um elemento.

3.5 EXEMPLO DA APLICAÇÃO DO ALGORITMO APRIORI

Dada a base de dados **D** com 9 sessões (quadro 2) e um suporte mínimo de 2 sessões (22,23%).

SessãoID	Páginas Web Visitadas
1	a.html, b.html, e.html
2	b.html, d.html
3	b.html, c.html
4	a.html, b.html. d.html
5	a.html, c.html
6	b.html, c.html
7	a.html, c.html
8	a.html, b.html, c.html, e.html
9	a.html, b.html, c.html

Quadro 2 - Exemplo da base de dados de 9 sessões

Na primeira iteração do algoritmo, cada página *Web* visitada é elemento do conjunto $\mathbf{L_1}$. O algoritmo (figura 6) simplesmente faz uma varredura de todas sessões para contar o número de ocorrências de cada página *Web* visitada calculando o suporte da mesma (quadro 3)

Páginas *Web*	Suporte
{a.html}	6 / 9 = 66,67%
{b.html}	7 / 9 = 77,78%
{c.html}	6 / 9 = 66,67%
{d.html}	2 / 9 = 22,23%
{e.html}	2 / 9 = 22,23%

Quadro 3 - Candidatos L_1

Para descobrir o conjunto de candidatos $\mathbf{C_2}$, o algoritmo usa o produto $\mathbf{L_1}$ **x** $\mathbf{L_1}$ aplicando os critérios do **apriori-gen** (figura 7) e novamente é feita a varredura em **D** para determinar o número de ocorrências dos candidatos $\mathbf{C_2}$ (quadro 4).

Páginas *Web*	**Suporte**
{a.html, b.html}	4 / 9 = 44,45%
{a.html, c.html}	4 / 9 = 44,45%
{a.html, d.html}	1 / 9 = 11,12%
{a.html, e.html}	2 / 9 = 22,23%
{b.html, c.html}	4 / 9 = 44,45%
{b.html, d.html}	2 / 9 = 22,23%
{b.html, e.html}	2 / 9 = 22,23%
{c.html, d.html}	0 / 9 = 0,0%
{c.html, e.html}	1 / 9 = 11,12%
{d.html, e.html}	0 / 9 = 0,0%

Quadro 4 - Candidatos C_2

A partir de $\mathbf{C_2}$, eliminam-se as páginas *Web* com o suporte menor que o suporte mínimo gerando $\mathbf{L_2}$ (quadro 5).

Páginas *Web*	**Suporte**
{a.html, b.html}	4 / 9 = 44,45%
{a.html, c.html}	4 / 9 = 44,45%
{a.html, e.html}	2 / 9 = 22,23%
{b.html, c.html}	4 / 9 = 44,45%
{b.html, d.html}	2 / 9 = 22,23%
{b.html, e.html}	2 / 9 = 22,23%

Quadro 5 - Candidatos L_2

A geração do conjunto de candidatos $\mathbf{C_3}$ (quadro 6) é feita passando $\mathbf{L_2}$ para o procedimento **apriori-gen** (Figura 7).

Páginas *Web*	Suporte
{a.html, b.html, c.html}	2 / 9 = 22,23%
{a.html, b.html, e.html}	2 / 9 = 22,23%
{a.html, c.html, e.html}	1 / 9 = 11,12%
{b.html, c.html, d.html}	0 / 9 = 0,0%
{b.html, c.html, e.html}	1 / 9 = 11,12%
{b.html, d.html, e.html}	0 / 9 = 0,0%

Quadro 6 - Candidatos C_3

A seguir, aplica-se o segundo passo do **apriori-gen** (figura 8) que afirma que todos os subconjuntos não vazios de um conjunto $\mathbf{C_k}$ pertencem ao conjunto $\mathbf{L_{k-1}}$.

Com base nesse aspecto, verifica-se cada candidato em $\mathbf{C_3}$ e elimina-se o candidato cujo subconjunto de dois elementos não é freqüente em $\mathbf{L_2}$:

a) Os subconjuntos de dois elementos de {a.html, b.html, c.html} são {a.html, b.html}, {a.html, c.html} e {b.html, c.html} e pertencem ao conjunto de candidatos $\mathbf{L_2}$, portanto, mantém-se {a.html, b.html, c.html} em $\mathbf{C_3}$;

b) Os subconjuntos de dois elementos de {a.html, b.html, e.html} são {a.html, b.html}, {a.html, e.html} e {b.html, e.html} pertencendo ao conjunto de candidatos $\mathbf{L_2}$, portanto mantém-se {a.html, b.html, e.html} em $\mathbf{C_3}$;

c) Os subconjuntos de dois elementos de {a.html, c.html, e.html} são {a.html, c.html}, {a.html, e.html} e {c.html, e.html}, entretanto, {c.html, e.htm} não pertence ao conjunto de candidatos $\mathbf{L_2}$, portanto, {a.html, c.html, e.html} é eliminado de $\mathbf{C_3}$;

d) Os subconjuntos de dois elementos de {b.html, c.html, d.html} são {b.html, c.html}, {b.html, d.html} e {c.html, d.html}, entretanto,

{c.html, d.htm} não pertence ao conjunto de candidatos $\mathbf{L_2}$, portanto, {b.html, c.html, d.html} é eliminado de $\mathbf{C_3}$;

e) Os subconjuntos de dois elementos de {b.html, c.html, e.html} são {b.html, c.html}, {b.html, e.html} e {c.html, e.html}, entretanto, {c.html, e.htm} não pertence ao conjunto de candidatos $\mathbf{L_2}$, portanto, {b.html, c.html, e.html} é eliminado de $\mathbf{C_3}$;

f) Os subconjuntos de dois elementos de {b.html, d.html, e.html} são {b.html, d.html}, {b.html, e.html} e {d.html, e.html}, entretanto {d.html, d.htm} não pertence ao conjunto de candidatos $\mathbf{L_2}$, portanto {b.html, d.html, e.html} é eliminado de $\mathbf{C_3}$;

Após essa fase de corte dos candidatos $\mathbf{C_3}$ cujos subconjuntos não pertencem ao conjunto $\mathbf{L_2}$, têm-se os novos candidatos $\mathbf{C_3}$ (quadro 7).

Páginas *Web*	**Suporte**
{a.html, b.html, c.html}	2 / 9 = 22,23%
{a.html, b.html, e.html}	2 / 9 = 22,23%

Quadro 7 - Candidatos C_3 após o processo de corte

Como todos os elementos do conjunto $\mathbf{C_3}$ têm suporte maior ou igual ao suporte mínimo, tem-se $\mathbf{L_3}$ (quadro 8) igual a $\mathbf{C_3}$.

Páginas *Web*	**Suporte**
{a.html, b.html, c.html}	2 / 9 = 22,23%
{a.html, b.html, e.html}	2 / 9 = 22,23%

Quadro 8 - Candidatos L_3

O algoritmo usa $\mathbf{L_3}$ para gerar um conjunto candidato com 4 elementos $\mathbf{C_4}$ (quadro 9).

Páginas *Web*	Suporte
{a.html, b.html, c.html, e.html}	1 / 9 = 11,12%

Quadro 9 - Candidatos C_4

Aplicando o processo de corte (figura 8) nota-se que o subconjunto {b.html, c.html, e.html} não pertence ao conjunto de candidatos $\mathbf{L_3}$, logo, o conjunto {a.html, b.html, c.html, e.html} é eliminado, resultando $\mathbf{C_4 = \varnothing}$ e, conseqüentemente, $\mathbf{L_4 = \varnothing}$. Portanto, o algoritmo Apriori termina seu processo de geração de candidatos.

Com o conjunto $\mathbf{L_k}$ dos candidatos válidos estabelecido, inicia-se o processo de geração de regras de associação passando o conjunto $\mathbf{L_k}$, com $\mathbf{k \geq 2}$ para a função **genrules**.

Basicamente a função **genrules** (figura 9) executa os seguintes passos:

a) Para cada elemento freqüente **l** em $\mathbf{L_k}$, gerar todos os subconjuntos não vazios de **l**;

b) Para todo subconjunto **s** não vazio de **l**, gerar a regra $\mathbf{s \Rightarrow (l - s)}$ se $\frac{\textbf{suporte(l)}}{\textbf{suporte(s)}} \geq$ **minconf**, na qual **minconf** é a confiança mínima estipulada.

Por exemplo, suponha o candidato **l** = {a.html, b.html, e.html} em $\mathbf{L_3}$. Os subconjuntos não vazios de **l** são {a.html, b.html}, {a.html, e.html}, {b.html, e.html}, {a.html}, {b.html} e {e.html}. O quadro 10 mostra regras de associação resultantes da aplicação do Algoritmo Apriori.

Regra (s ⇒ (I - s))	Confiança
{a.html, b.html} ⇒ {e.html}	$\frac{\text{suporte}(\{a.html, b.html, e.html\})}{\text{suporte}(\{a.html, b.html\})} = \frac{2}{4} = 50\%$
{a.html, e.html} ⇒ {b.html}	$\frac{\text{suporte}(\{a.html, b.html, e.html\})}{\text{suporte}(\{a.html, e.html\})} = \frac{2}{2} = 100\%$
{b.html, e.html} ⇒ {a.html}	$\frac{\text{suporte}(\{a.html, b.html, e.html\})}{\text{suporte}(\{b.html, e.html\})} = \frac{2}{2} = 100\%$
{a.html} ⇒ {b.html, e.html}	$\frac{\text{suporte}(\{a.html, b.html, e.html\})}{\text{suporte}(\{a.html\})} = \frac{2}{6} = 33,3\%$
{b.html} ⇒ {a.html, e.html}	$\frac{\text{suporte}(\{a.html, b.html, e.html\})}{\text{suporte}(\{b.html\})} = \frac{2}{7} = 28,5\%$
{e.html} ⇒ {a.html, b.html}	$\frac{\text{suporte}(\{a.html, b.html, e.html\})}{\text{suporte}(\{e.html\})} = \frac{2}{2} = 100\%$

Quadro 10 - Regras de Associação para {a.html, b.html, e.html}

Estabelecendo uma confiança mínima, por exemplo, de 50% têm-se as regras {a.html, b.html} ⇒ {e.html}, {a.html, e.html} ⇒ {b.html}, {b.html, e.html} ⇒ {a.html} e {e.html} ⇒ {a.html, b.html} com confiança maior ou igual à confiança mínima.

Observa-se que para a regra {a.html, e.html} ⇒ {b.html} significa que um visitante que acessou as páginas *Web* **a.html** e **e.html** tem a chance de 100% de visitar a página *Web* **b.html**. Assim o *site* poderia informar o *hyperlink* da página **b.html** para o visitante e, portanto, facilitando o acesso direto a essa página *Web*.

3.6 CONCLUSÃO

A tarefa de descobrir regras de associação permite encontrar relações entre *hyperlinks* das páginas *Web* acessadas pelo visitante em um *site* determinando probabilidades de acesso as essas páginas *Web*.

No próximo capítulo é apresentado um agente de suporte à navegação que utiliza a tarefa de regras de associação e critérios de grau de interesse.

4 AGENTE DE SUPORTE À NAVEGAÇÃO

4.1 INTRODUÇÃO

Dentro do contexto de mineração do uso na *Web* apresenta-se neste capítulo um agente de suporte à navegação, o *Fast Navigation Agent* (FNA), ilustrando sua arquitetura, funcionamento e interfaces com usuário (visitantes).

4.2 ARQUITETURA DO FNA

Com a finalidade de auxiliar e facilitar o visitante em sua navegação, o *Fast Navigation Agent* (FNA) exibe os *hyperlinks* personalizados baseados nas navegações passadas feitas pelo visitante no *site*.

O FNA é composto por dois módulos: O primeiro módulo, denominado FNA coletor (FNAc) presente nas páginas *Web* dos *sites*, tem a função de exibir os *hyperlinks* personalizados para o visitante e enviar as URLs das páginas *Web* visitadas para um banco de dados. O segundo módulo, localizado externamente aos *sites*, denominado FNA minerador (FNAm), tem o objetivo de descobrir padrões de navegação de cada visitante nos *sites*.

A figura 10 mostra a arquitetura do FNA. No *site* do FNA ficam hospedados o FNAc, o banco de dados e todas as informações relacionadas ao FNA. Observa-se que o FNAc é carregado nas páginas *Web* dos *sites*, cujas páginas *Web* farão parte no processo de descoberta de padrões feita pelo FNAm. O módulo FNAm atua diretamente sobre o banco de dados, consultando as páginas *Web* acessadas pelos visitantes nos *sites*, minerando e descobrindo padrões de navegação com a aplicação do algoritmo Apriori e enviando os resultados da mineração para o banco de dados, para que o visitante possa ver, a partir do FNAc, seus *hyperlinks* personalizados nos *sites* visitados.

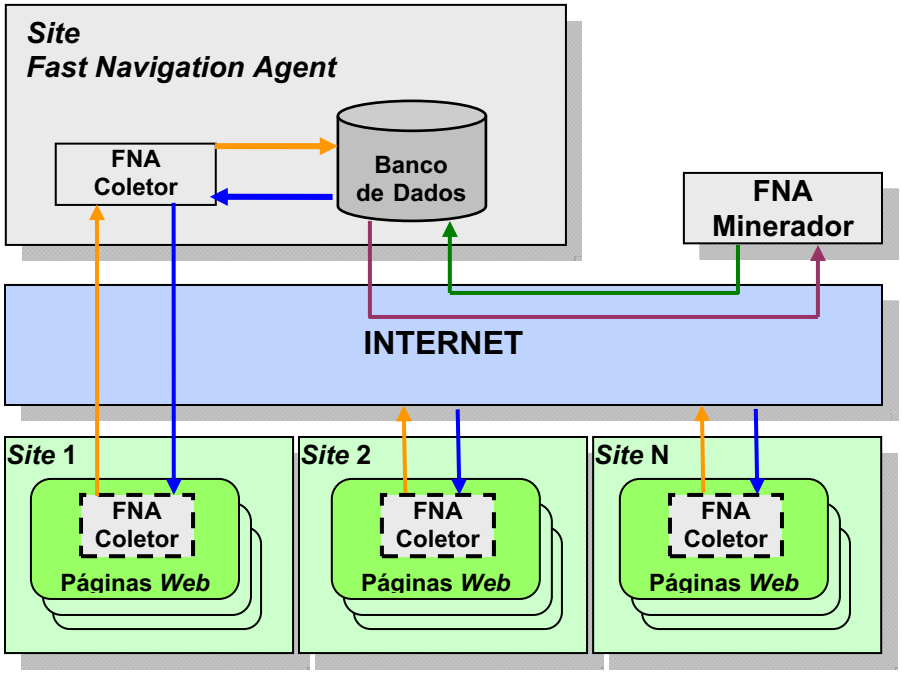

Figura 10 - Visão geral da aquitetura do FNA

A seguir são apresentados os detalhes do funcionamento do FNAc e FNAm.

4.3 FNA COLETOR

O FNAc encontra-se presente nas páginas *Web* dos *sites*, cujas URLs das páginas *Web* são armazenadas no banco de dados hospedado no *site Fast Navigation Agent*.

O uso do FNAc não requer recursos adicionais, ou seja, pode ser usado tanto em *sites* estáticos e simples quanto em *sites* dinâmicos e mais elaborados. A sua implantação e utilização são bem simples.

Entretanto, para utilizá-lo, o *site* deve estar cadastrado no banco de dados do *Fast Navigation Agent*, pois após o cadastro é gerado e atribuído um código de identificação (cid) que deve ser usado no *script* (figura 11) que é implementado nas páginas *Web* do *site* solicitante.

```
<script language="JavaScript" src="http://www.fastnavigationagent.com/agente/fnac.php?cid=XXXXXX"></script>
```

Figura 11 - *Script* que carrega o FNAc

O código de identificação (cid) serve para verificar se o *site* solicitante tem permissão para utilizar o FNAc. Caso negativo, o FNAc fica desabilitado, impossibilitando o uso do mesmo para os visitantes. A verificação e a permissão são feitas no próprio FNAc com o objetivo de:

a) Identificar os *sites*;
b) Saber quais visitantes estão utilizando o agente nos *sites* identificados;
c) Não permitir o uso do agente em *sites* que não estejam cadastrados no banco de dados do *Fast Navigation Agent* e, assim, a base de dados não armazena informações irrelevantes.

Sendo assim, nos sites identificados e autorizados, o ícone do FNA é exibido. Quando o visitante clicar no ícone do FNA é exibida a interface do agente, na qual o visitante deve se identificar (figura 12).

Após a identificação do visitante, os *hyperlinks* personalizados são exibidos (figura 13).

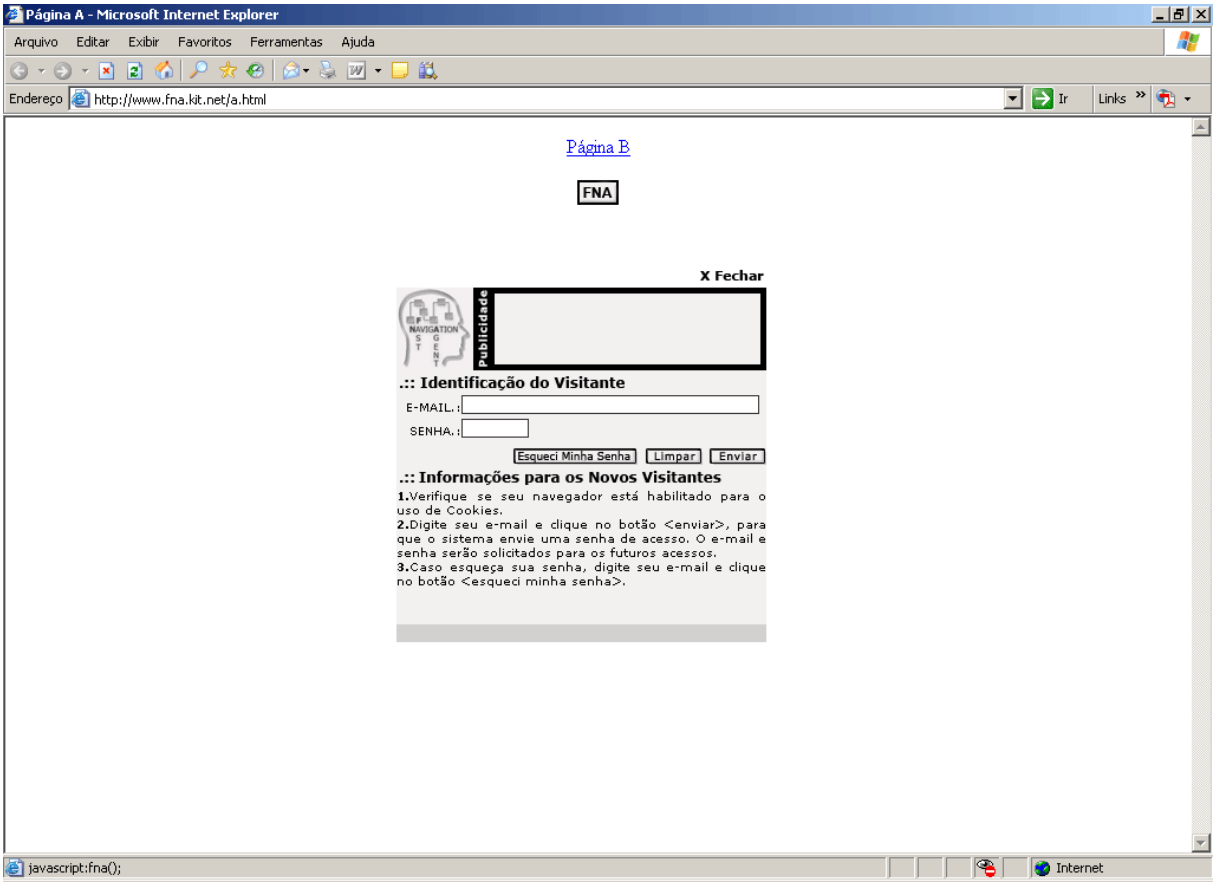

Figura 12- Interface do agente com o visitante

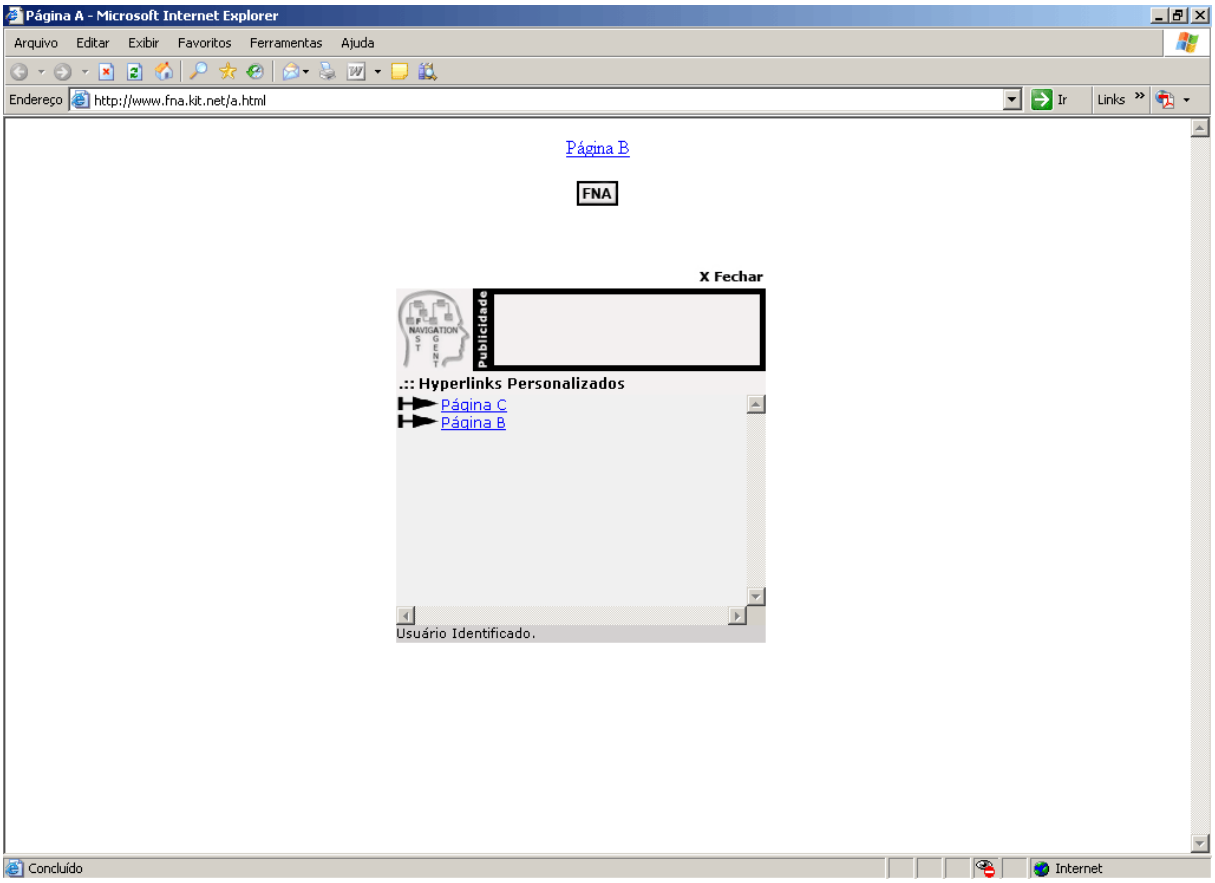

Figura 13 - Exibição dos *hyperlinks* personalizados

A figura 14 ilustra, através de um diagrama de estados, o funcionamento do FNAc de forma mais detalhada. Observa-se que antes de solicitar a identificação do visitante, o FNAc verifica se existe um *cookie* no computador do visitante com o valor de identificação. Caso exista, o FNAc não solicita mais a identificação do visitante durante a sua sessão. Esse *cookie* fica presente até o momento em que o visitante encerrar seu navegador.

Ao usar o agente pela primeira vez, o visitante deve digitar seu *e-mail* e clicar no botão *enviar*. Assim, o agente faz uma consulta à base de dados a procura do *e-mail* do visitante. Como o *e-mail* não existe, é gerada uma senha e esta é enviada para o *e-mail* informado pelo visitante. Com a posse da senha, ele preenche o campo a ela reservado e clica novamente no botão *enviar*. Com o visitante autenticado, um *cookie* é gravado no computador do visitante com o valor de identificação. Em toda página *Web* com o agente presente, esse *cookie* será consultado e informará o agente que o visitante está autenticado.

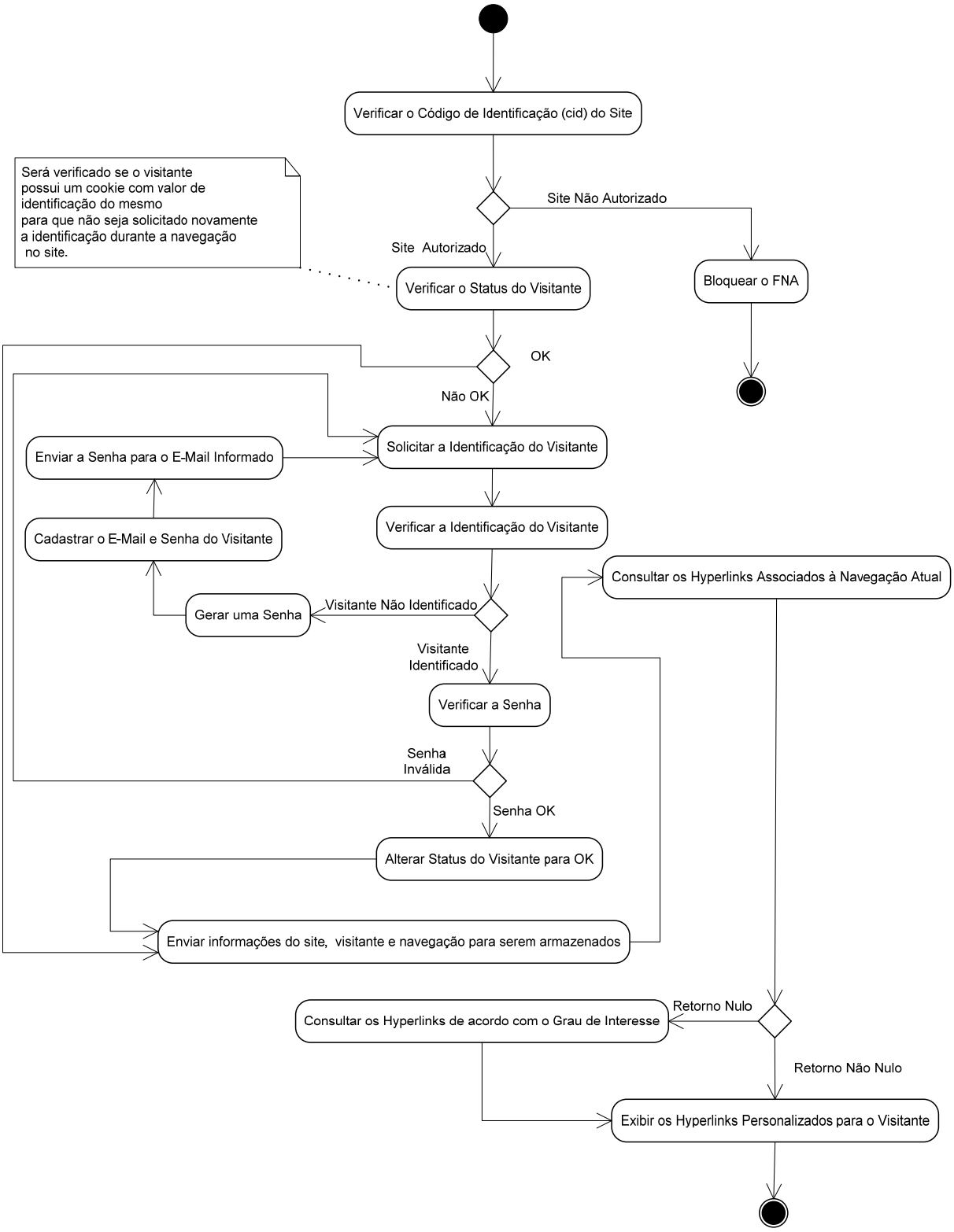

Figura 14 - Diagrama de atividades do FNAc

Após a autenticação com sucesso do visitante, o agente envia para o servidor as seguintes informações:

a) Código de Identificação da Sessão;
b) Código de Identificação do *Site* (cid);
c) Código de Identificação do Visitante (armazenado no *cookie*);
d) Endereço IP do Visitante;
e) A URL da página *Web* em que o visitante se encontra;
f) A URL de origem da página *Web* na qual o visitante se encontra;
g) e outras informações que serão utilizadas para análises estatísticas.

Todas essas informações são armazenadas no banco de dados para que o *FNAm* possa analisar e gerar padrões a partir delas.

Após o envio dessas informações, é realizada uma consulta à base de dados com a finalidade de obter as URLs associadas (utilização das regras de associação geradas) à navegação atual do visitante. Caso o resultado dessa consulta não retorne um *hyperlink* para alguma navegação, é feita uma consulta das URLs das páginas *Web* seguindo os critérios de grau de interesse em cada uma delas. Esses critérios são (CHAN, 2000):

a) Freqüência de visita às páginas *Web*;
b) Páginas *Web* recentemente visitadas;
c) Tempo de permanência nas páginas *Web*;.

Chan (2000) utiliza também como critério o armazenamento da URL das páginas *Web* nos favoritos do navegador do visitante, porém, neste trabalho, foi substituído pelo critério em que o visitante acessa as páginas *Web* através do FNAc.

A partir desses critérios são gerados os graus de interesse para cada URL. O visitante recebe os *hyperlinks* com base nesses graus de interesse em ordem decrescente. A determinação do grau de interesse é o resultado da expressão (3).

$$\text{Grau de Interesse} = \text{Freqüência(Página Web)} \times \left(1 + \frac{\sum \text{AcessoPeloAgente(PáginaWeb)}}{\text{Freqüência(PáginaWeb)}} + \frac{\sum \text{DuraçãoDaVisita(PáginaWeb)}}{\text{DuraçãoTotalDaVisitaNoSite}} + \frac{\text{UltimoAcesso(PáginaWeb)} - \text{PrimeiroAcessoNoSite}}{\text{TempoAtual} - \text{PrimeiroAcessoNoSite}} \right) \quad (3)$$

$$\text{onde Freqüência(PáginaWeb)} > 0, \text{DuraçãoTotalDaVisitaNoSite} > 0 \text{ e (TempoAtual} - \text{PrimeiroAcessoNoSite)} > 0$$

Finalmente, destaca-se que o FNAc não interfere na estrutura de *hyperlinks* do *site* e nem no *design* da página *Web*, pois toda vez que o visitante clicar sobre o ícone FNA sempre abrirá uma pequena interface sobre a página *Web* ativa exibindo os *hyperlinks* personalizados.

4.4 FNA MINERADOR

O FNAm, localizado externamente ao *site Fast Navigation Agent* e também aos *sites* que utilizam o FNAc, pode ser executado em um computador com acesso à *Internet*. O FNAm atua diretamente no banco de dados no qual todas as informações dos visitantes e *sites* cadastrados no sistema FNA (anexo A) estão armazenados.

A figura 15 ilustra a tabela do banco de dados em que são armazenados os dados referentes ao acesso do visitante em um *site*.

fnaRegistrosAcessos	
PK	RegistroAcessoID
FK	SiteID
FK	VisitanteID
	ClienteIP
	SessaoID
	DataHoraAcesso
	DataHoraSaída
	AgenteReferrer
	URLOrigem
	URL
	TituloPagina
FK	NavegadorID
FK	SistemaOperacionalID
	ResolucaoVideo
FK	PaisID

Figura 15 - Tabela de registros de acesso dos visitantes nos *sites*

O FNAm é implementado com base no algoritmo Apriori, encontrando relações entre as URLs das páginas *Web* acessadas, para cada visitante nos respectivos *sites* armazenados na base de dados. O FNAm envia para o banco de dados as associações entre as URLs das páginas *Web* encontradas (figura 16). Assim, através do FNAc, os visitantes visualizam os *hyperlinks* de acordo com seus perfis.

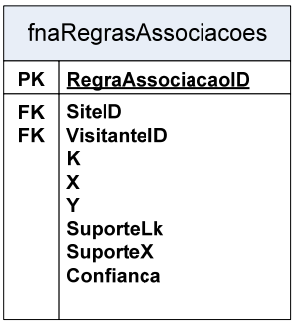

Figura 16 - Tabela das regras de associação

Para determinar quais associações são armazenadas, o FNAm calcula um nível de confiança mínima (c_m) e insere na base de dados todas as relações com o nível de confiança (c_i) acima ou igual à confiança mínima. O cálculo do nível de confiança mínima é feito através da razão da soma de todos os níveis de confiança encontrados em cada associação pelo número total dos níveis de confiança (n).

$$c_m = \frac{\sum_{i=1}^{n} c_i}{n} \quad (4)$$

Se o nível de confiança mínima estiver próximo a 100% significa que as associações encontradas têm relações fortes entre as *URLs* das páginas *Web* resultantes da mineração e o perfil do visitante está bem traçado. Caso contrário, se o nível de confiança mínima tender a 0%, mais *hyperlinks* das páginas *Web* são exibidos para o visitante permitindo que o mesmo aponte quais páginas *Web* são de seu interesse, assim o agente "aprende" fazendo com que seu nível de confiança mínima aumente.

Para observar todas as etapas de mineração e descoberta das regras de associação foi desenvolvida uma interface (figura 17) que exibe as seguintes informações:

a) As URLs das páginas *Web* acessadas em cada sessão pelo visitante no *site*;
b) As URLs das páginas *Web* candidatas com seus respectivos valores de suporte;
c) As URLs das páginas *Web* com seus respectivos valores de suporte acima do suporte mínimo;
d) Todas as possíveis regras de associação baseadas nas URLs das páginas *Web* do item c;
e) As regras de associação com nível de confiança acima do nível de confiança mínima.

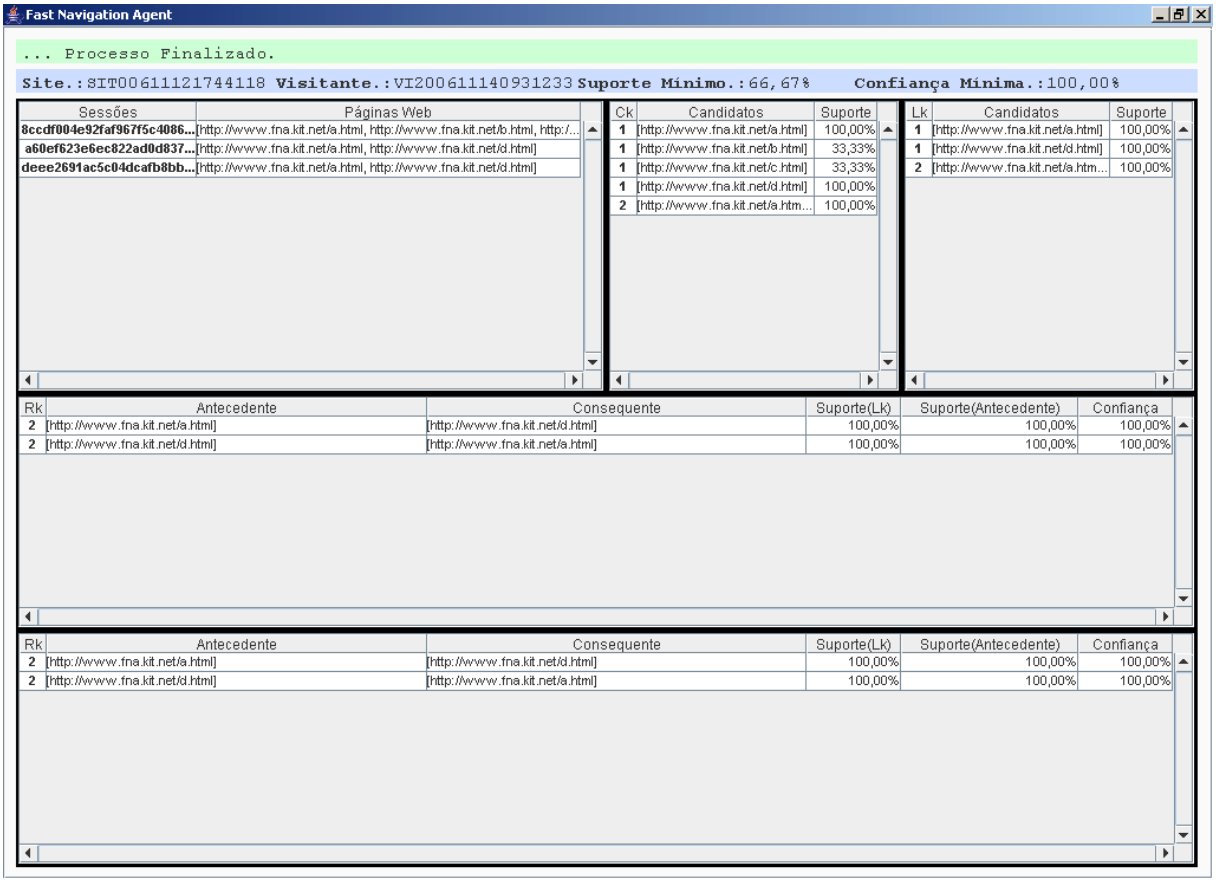

Figura 17 - Interface do FNAm

Essa interface do FNAm não está disponível para os visitantes e nem para os desenvolvedores dos *sites*. Ela foi implementada com a finalidade de observar a geração das regras de associação resultante da aplicação do algoritmo Apriori.

4.5 SIMULAÇÕES E RESULTADOS DO FNA

A fim de observar o comportamento do suporte a navegação provido pelo FNA e o comportamento dos agentes FNAc e FNAm foram desenvolvidos dois *sites*-teste em que são possíveis simular a navegação de um usuário.

A seguir são apresentados os dois exemplos feitos com os *sites*-teste.

4.5.1 1ª Exemplo: Utilização de um *site*-teste com uma estrutura simples

A figura 18 ilustra a estrutura de *hyperlinks* de um *site*-teste utilizado neste primeiro exemplo. Observa-se que para alcançar a página *Web* D a partir da página *Web* A, o visitante deverá acessar as páginas *Web* B e C, respectivamente.

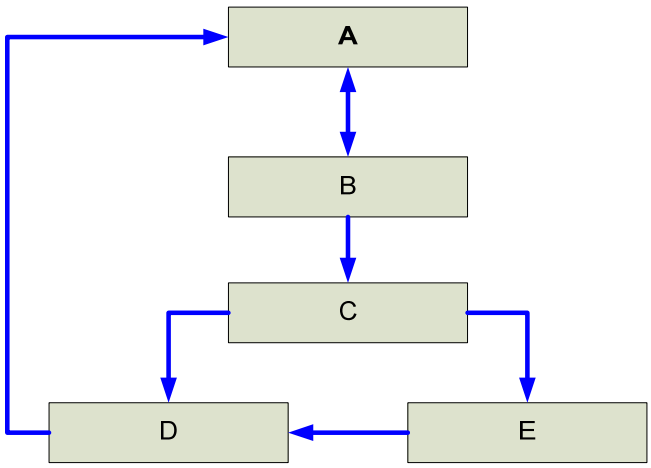

Figura 18 - Estrutura de *hyperlinks* do *site*-teste

A seguir são apresentados alguns testes do agente realizados nesse *site*-teste.

4.5.1.1 1º Teste: O primeiro acesso

Ao identificar um visitante que não tem cadastro e, portanto, um visitante que acessa o *site* pela primeira vez, o FNA não apresenta nenhum *hyperlink* personalizado uma vez que não possui nenhum registro de navegações passadas desse visitante até o momento (figuras 19 e 20).

Assim, ao acessar as páginas *Web* subseqüentes, o agente exibe os *hyperlinks,* com base no grau de interesse (equação 3) de cada página *Web*, pelas quais o visitante passou.

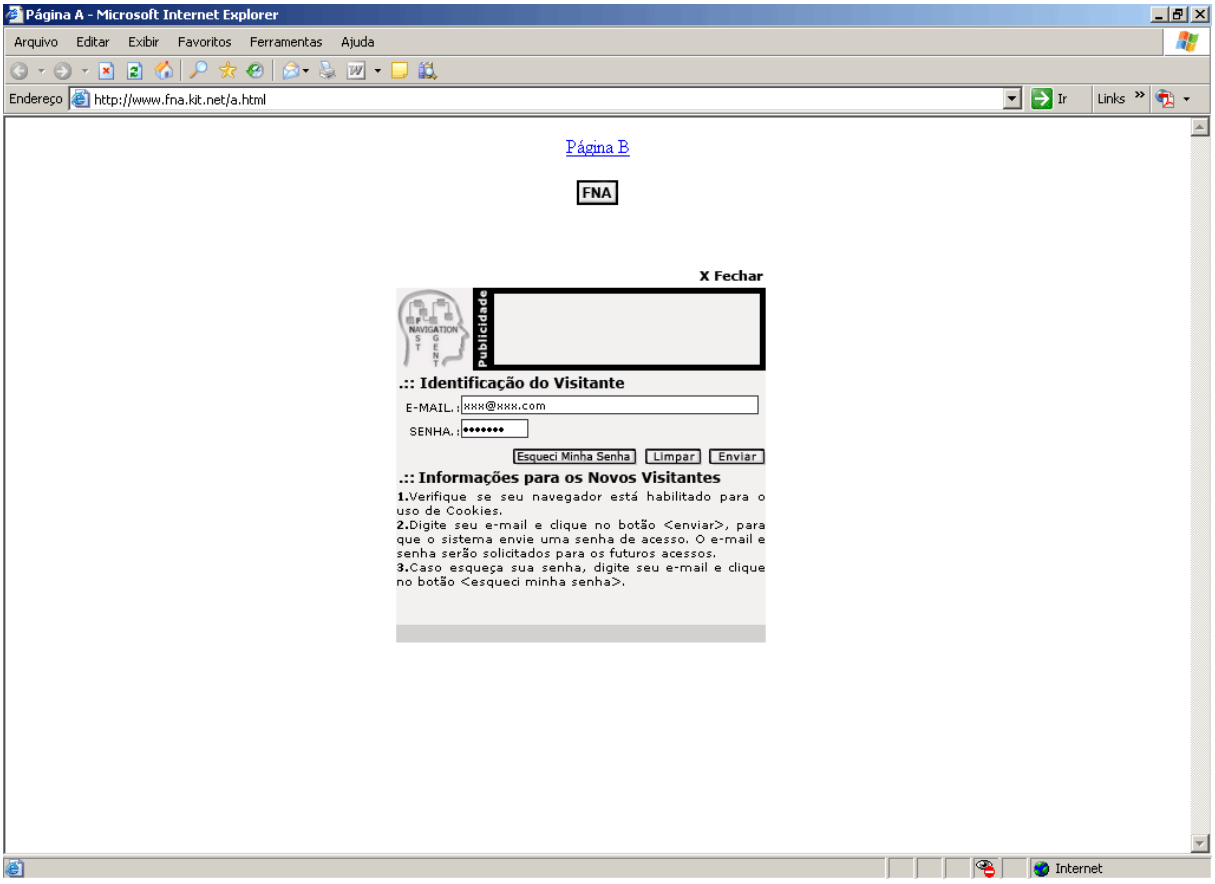

Figura 19 - 1º Teste: Identificação do visitante

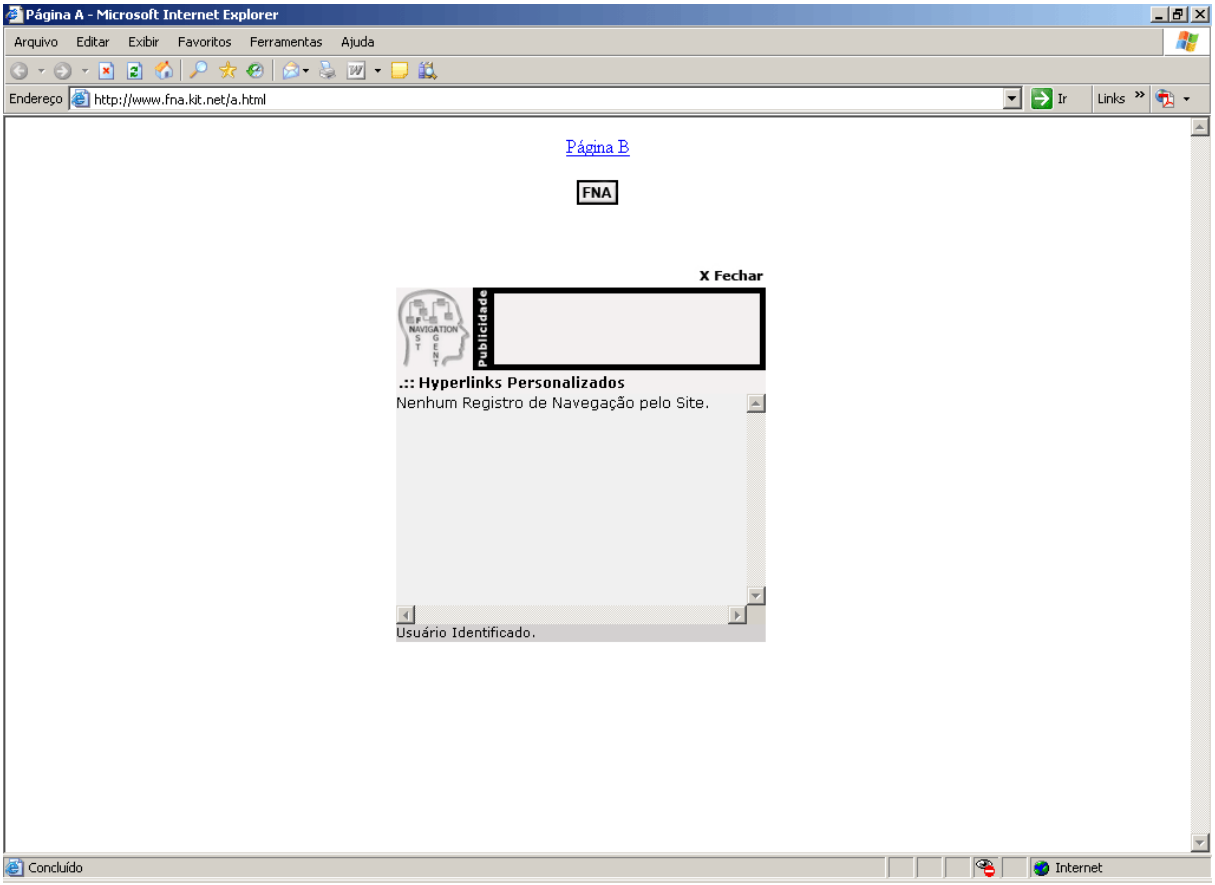

Figura 20 - 1º Teste: Exibição da mensagem que não há registro de acesso

Vale lembrar que após a identificação do visitante o agente além de exibir os *hyperlinks* das páginas *Web* da sessão atual, também registra toda navegação do visitante pelo *site*.

Quando o visitante acessa a Página B, é exibido, então, o *hyperlink* da Página A, pois agora o agente sabe que o mesmo acessou a Página A (figura 21).

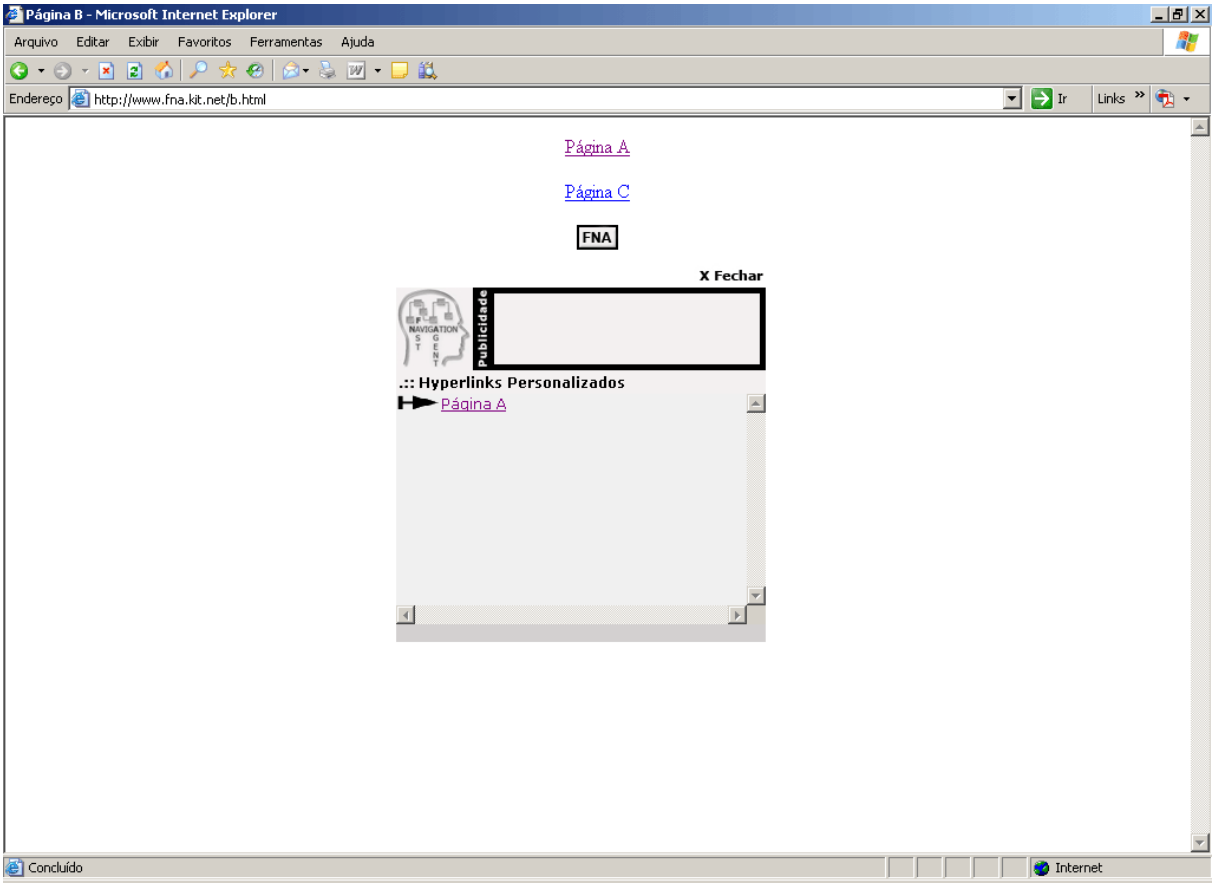

Figura 21 - 1º Teste: Exibição do *hyperlink* da última página *Web* acessada

Conseqüentemente, quando o visitante acessa a Página C, o agente exibe os *hyperlinks* das páginas *Web* B e A (figura 22). Finalmente, quando o visitante acessa a Página D, o visitante recebe os *hyperlinks* das páginas *Web* C, B e A, nesta ordem (figura 23). A ordem dos *hyperlinks* é baseada no grau de interesse (equação 3) cujo grau é um número real. Quanto maior o valor do grau de interesse, maior interesse na página *Web*.

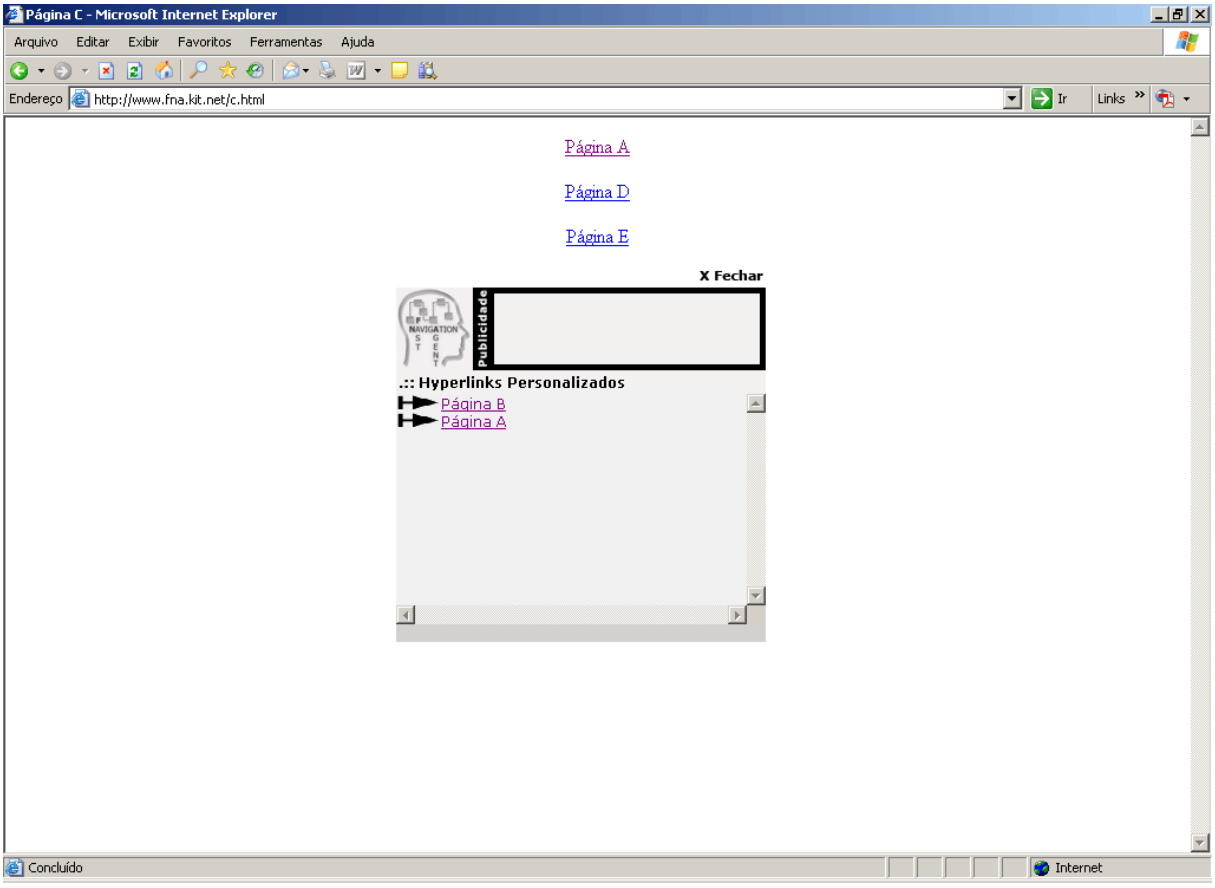

Figura 22 - 1º Teste: Exibição dos *hyperlinks* das páginas *Web* B e A a partir de C

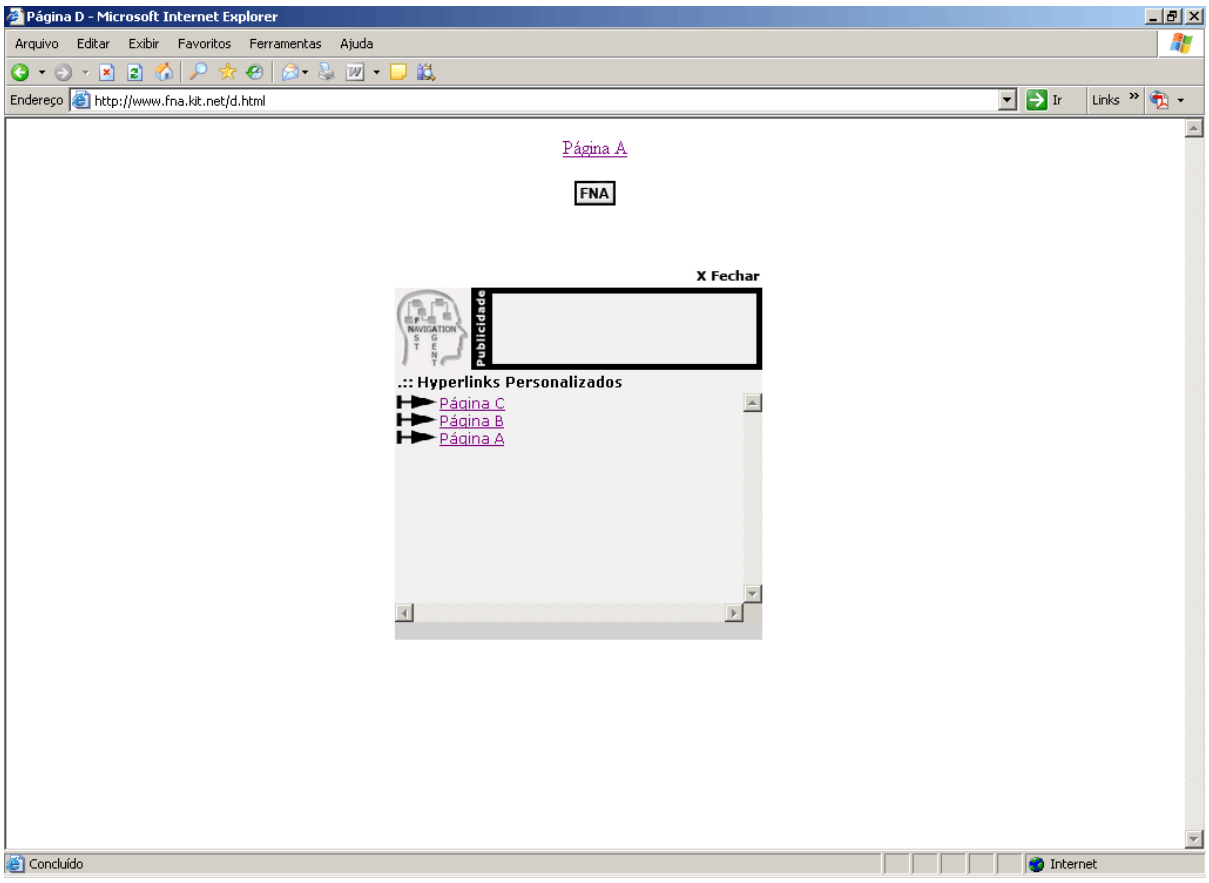

Figura 23 - 1º Teste: Exibição dos *hyperlinks* das páginas *Web* C, B e A a partir de D

4.5.1.2 2º Teste: O uso do agente pela segunda vez

Quando o visitante retornar ao mesmo *site* e, portanto for identificado pelo FNA pela segunda vez, o agente apresenta os *hyperlinks* das páginas *Web* da última passagem do visitante pelo *site* com base, também, no grau de interesse de cada página *Web* (figura 24). Isso ocorre, pois, até o momento, não foram geradas regras de associação. Ou seja, o banco de dados do FNA possui o registro de apenas uma sessão do visitante no *site*.

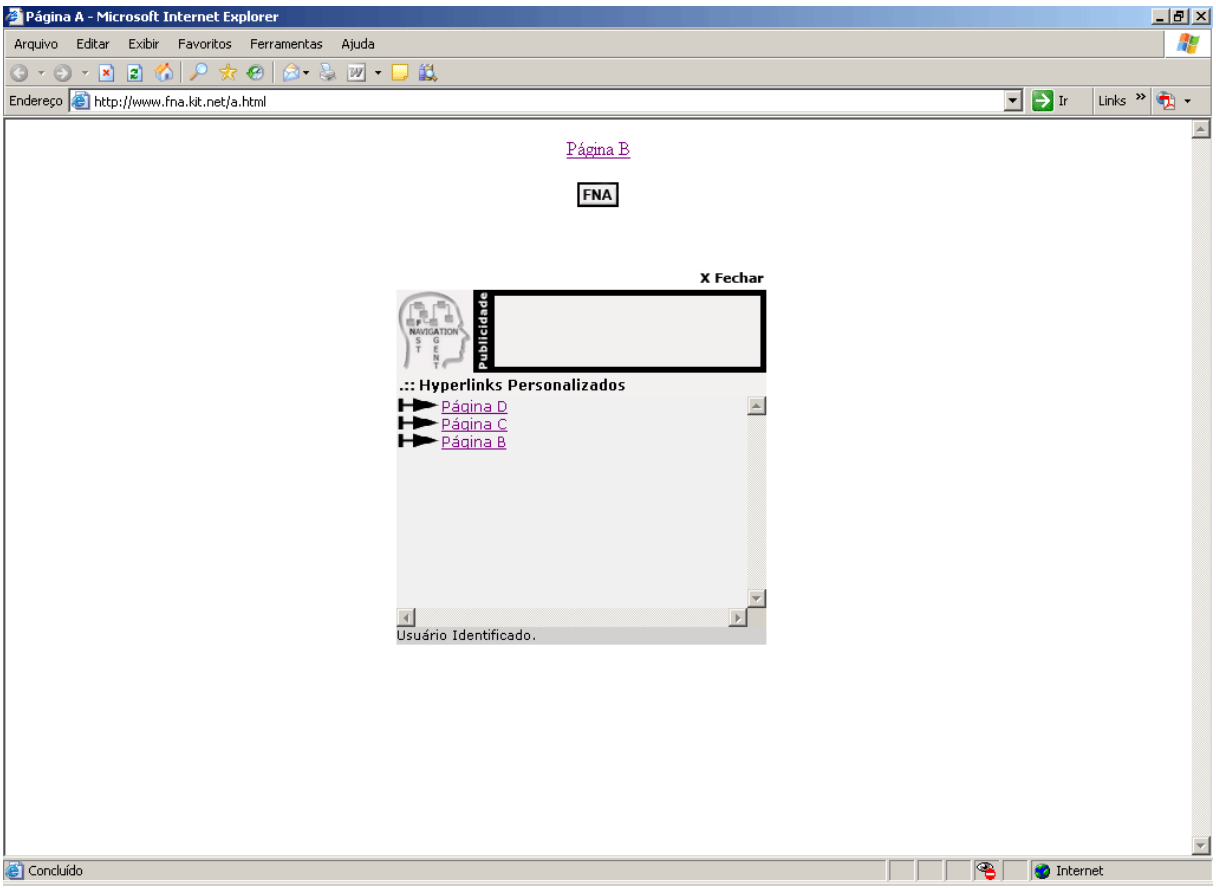

Figura 24 - 2º Teste: Exibição de *hyperlinks* com base no grau de interesse de cada página *Web*

Nota-se que o visitante estando na Página A, tem a possibilidade de acessar a Página D sem precisar passar pelas Páginas B e C.

4.5.1.3 3º Teste: Utilização das regras de associação

Com base nos 1º e 2º testes apresentados anteriormente (seções 4.5.1.1 e 4.5.1.2, respectivamente), o quadro 11 apresenta registros dos *hyperlinks* das páginas *Web* acessadas até o momento pelo visitante no *site*.

Sessões*	Páginas *Web*
8b79ef1829e2369c0220a4a06f997840	http://www.fna.kit.net/a.html, http://www.fna.kit.net/b.html, http://www.fna.kit.net/c.html, http://www.fna.kit.net/d.html
7742c670065d3ce91e725177d611ace1	http://www.fna.kit.net/a.html, http://www.fna.kit.net/d.html

***** a identificação da sessão é gerada pelo servidor *Web*.

Quadro 11 - Sessões do Visitante

Com base nos registros armazenados e um suporte mínimo de duas sessões, ou seja, pelo menos uma URL de uma página *Web* aparece em duas visitas, tem-se:

$$\textbf{Suporte Mínimo} = \frac{\textbf{2}}{\textbf{número total de sessões}} \times \textbf{100} = \frac{\textbf{2}}{\textbf{2}} \times \textbf{100} = \textbf{100\%}$$

Determinando os candidatos com suporte maior ou igual ao suporte mínimo, obtêm-se os candidatos freqüentes (ilustrado no quadro 12).

CANDIDATOS		
K	Candidato	Suporte
1	http://www.fna.kit.net/a.html	100.0%
1	http://www.fna.kit.net/b.html	50.0%
1	http://www.fna.kit.net/c.html	50.0%
1	http://www.fna.kit.net/d.html	100.0%
2	http://www.fna.kit.net/a.html, http://www.fna.kit.net/d.html	100.0%

Suporte ≥ Suporte Mínimo →

CANDIDATOS FREQÜENTES		
K	Candidato	Suporte
1	http://www.fna.kit.net/a.html	100.0%
1	http://www.fna.kit.net/d.html	100.0%
2	http://www.fna.kit.net/a.html, http://www.fna.kit.net/d.html	100.0%

Quadro 12 - Geração dos Candidatos Freqüentes

Nota-se que a URL *http://www.fna.kit.net/a.html* apresenta um suporte de 100,0%, ou seja, essa URL foi acessada em todas as visitas feitas pelo visitante no *site*. Em comparação a URL *http://www.fna.kit.net/b.html* resultou em um suporte de 50,0% devido ao acesso a esta URL ter ocorrido em uma sessão.

Com o resultado dos candidatos freqüentes são geradas todas as regras de associação possíveis (X $\Rightarrow$ Y) (quadro 13).

Rk	X*	Y**	Confiança
2	http://www.fna.kit.net/a.html	http://www.fna.kit.net/d.html	100.0%
2	http://www.fna.kit.net/d.html	http://www.fna.kit.net/a.html	100.0%

* Conjunto antecedente
** Conjunto conseqüente

Quadro 13 - Todas as regras de associação

O grau de confiança mínima é determinado pela expressão (4) resultando, nesse teste, em 100,0%. Com isso, todas as regras com confiança maior ou igual à confiança mínima são válidas.

Neste caso, ao comparar o quadro 13 com o quadro 11 pode-se verificar que um visitante que acessou a URL *http://www.fna.kit.net/a.html* sempre acessou a URL *http://www.fna.kit.net/d.html*.

Nesse mesmo teste, quando o visitante acessa a Página A, o FNA exibe somente o *hyperlink* para Página D. Conseqüentemente, o visitante acessa diretamente a Página D sem precisar acessar as Páginas B e C (figura 25). Na Página D o agente exibe os *hyperlinks* das últimas sessões com base nos graus de interesse de cada um deles em ordem inversa (figura 26), pois não existem regras de associação baseadas na navegação atual do visitante no *site*, isto é, para o conjunto antecedente {http://www.fna.kit.net/a.html, http://fna.kit.net/d.html} (quadro 13).

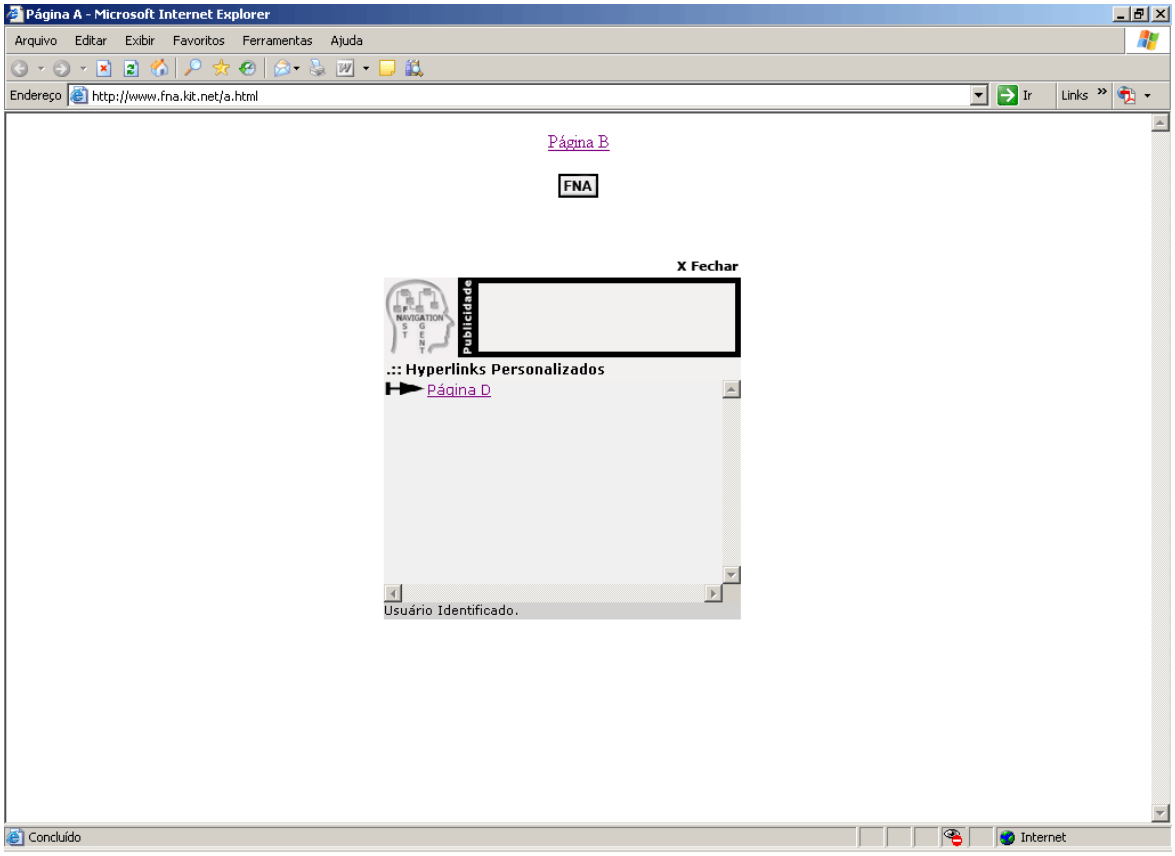

Figura 25 - 3º Teste: Exibição do *hyperlink* da página *Web* D a partir da página A

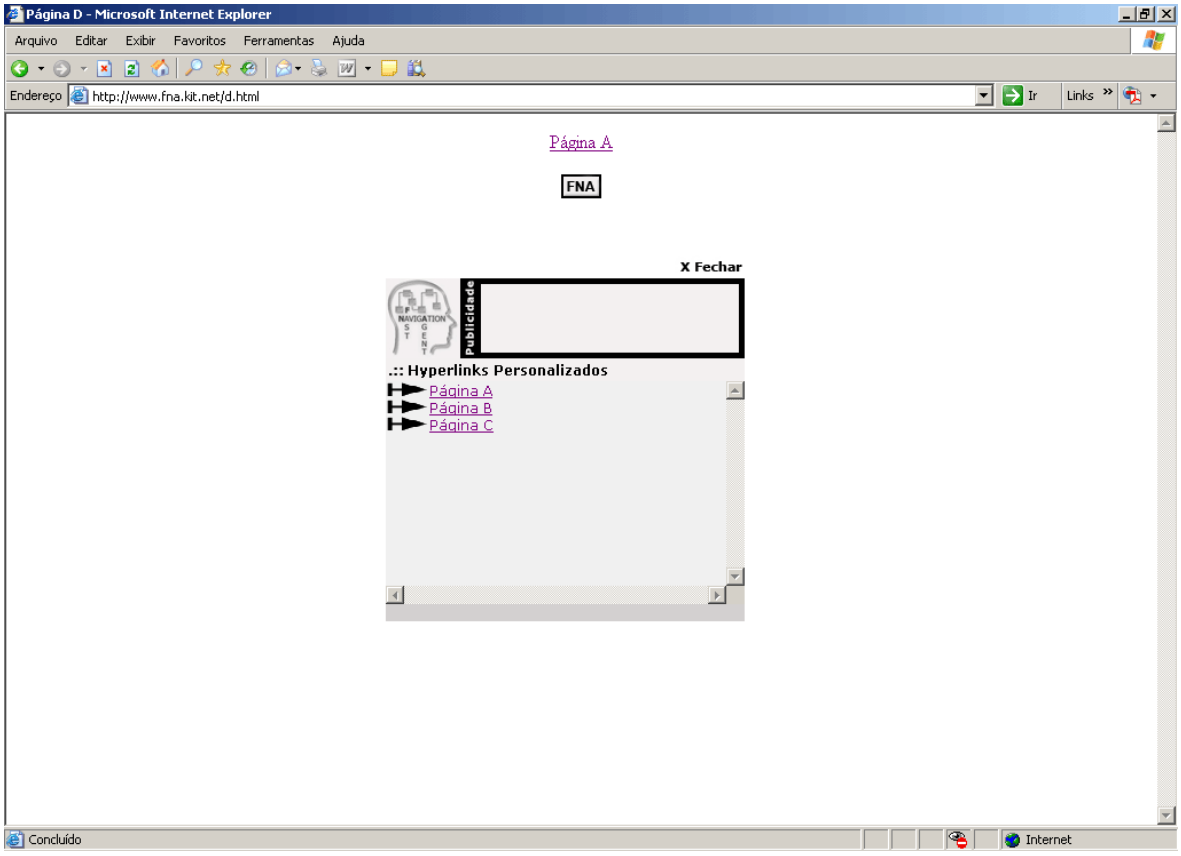

Figura 26 - 3º Teste: Exibição dos *hyperlinks* das páginas *Web* com base no grau de interesse

4.5.2 2ª Exemplo: Aplicação do FNA em um *site*-teste com uma estrutura mais complexa

O segundo exemplo utilizou um outro *site*-teste que apresenta uma estrutura de *hyperlinks* mais complexa (figura 27) que a estrutura usada no primeiro exemplo.

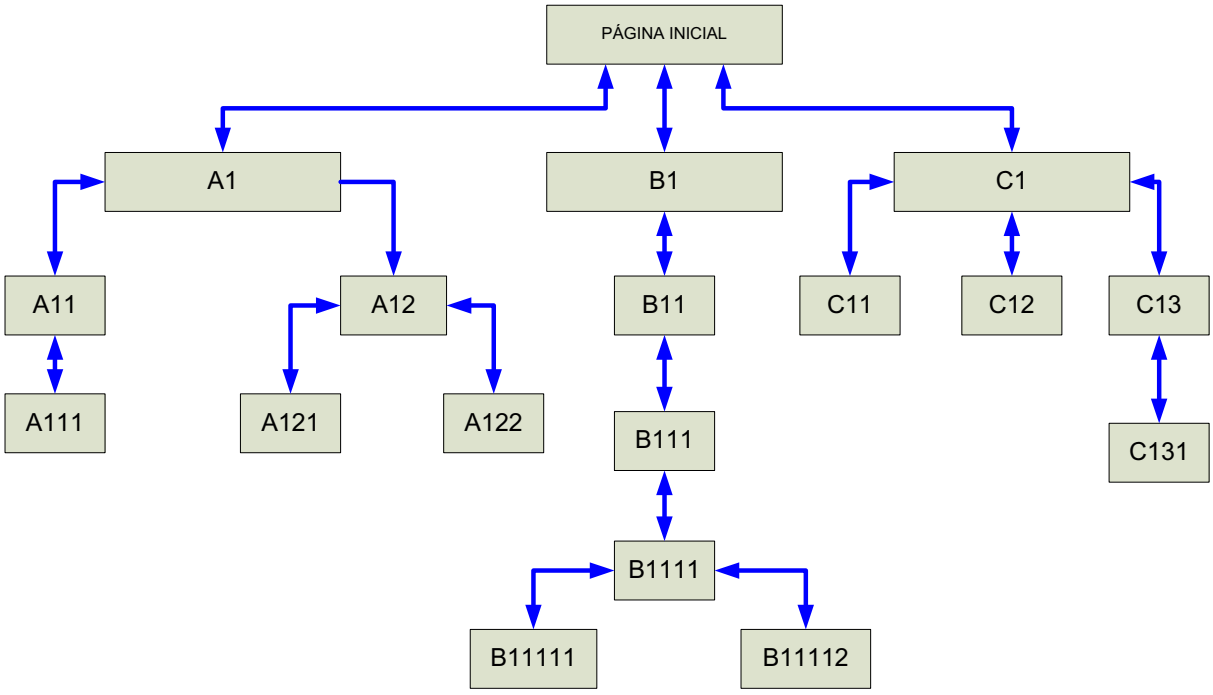

Figura 27 - Estrutura de *hyperlinks* mais complexa

Nota-se que no mapa de navegação desse *site* o visitante que se encontra na página *Web* A111 não tem acesso diretamente, por exemplo, à página *Web* B11112. Entretanto, o visitante deverá voltar para página inicial e seguir outro caminho para alcançar a página *Web* desejada.

A seguir são apresentados os testes utilizando o grau de interesse e regras de associação com a finalidade de oferecer uma navegação rápida e fácil.

4.5.2.1 1º Teste: Utilização do grau de interesse

Após o visitante se identificar e acessar a página *Web* A111, verifica-se que o visitante tem a possibilidade de voltar diretamente para página inicial, através do FNA, sem a necessidade de acessar as páginas *Web* A11 e A1 (figura 28).

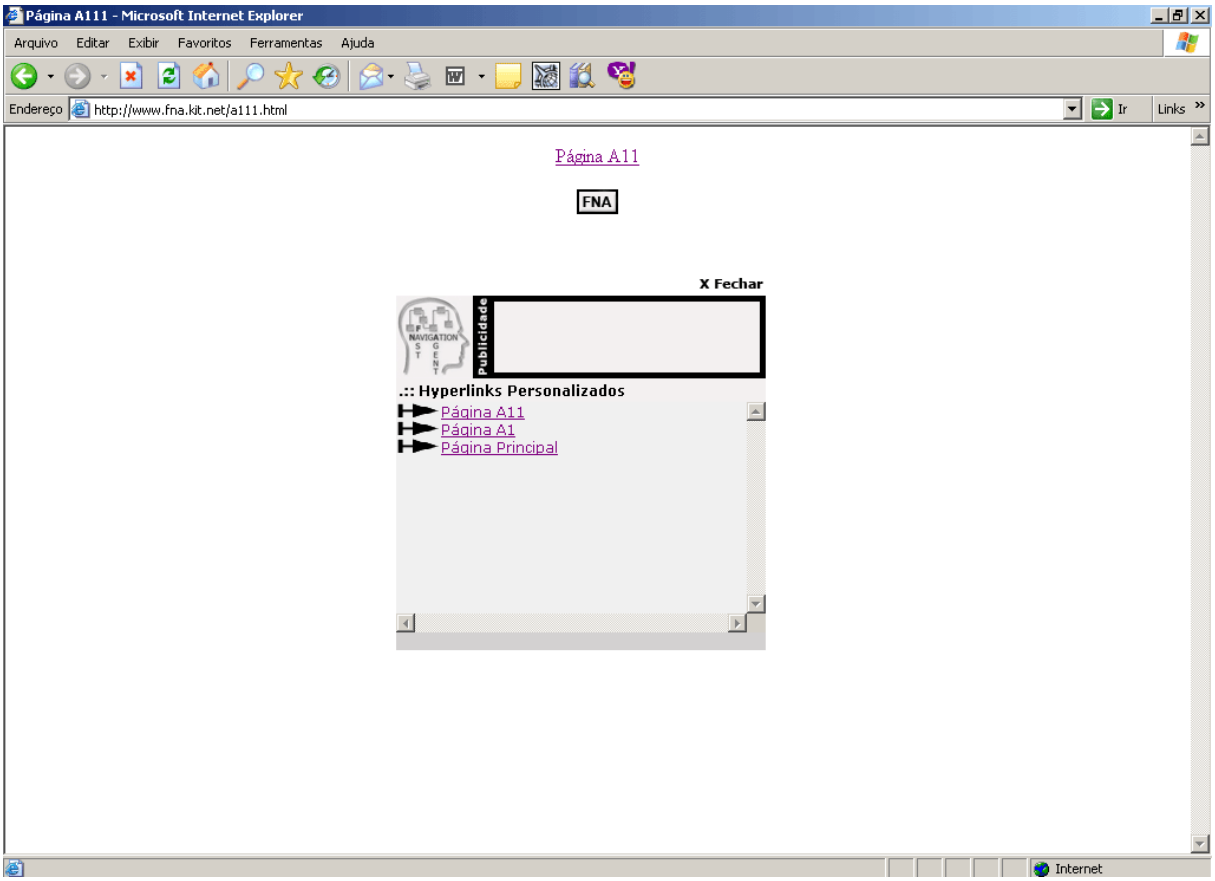

Figura 28 - 1º Teste: Exibição de *hyperlinks* com base no grau de interesse

Observa-se que a ordem dos *hyperlinks* apresentados na interface do agente vai da página *Web* recentemente acessada às páginas *Web* visitadas anteriormente. Essa ordem de *hyperlinks* pode ser vista também na figura 29.

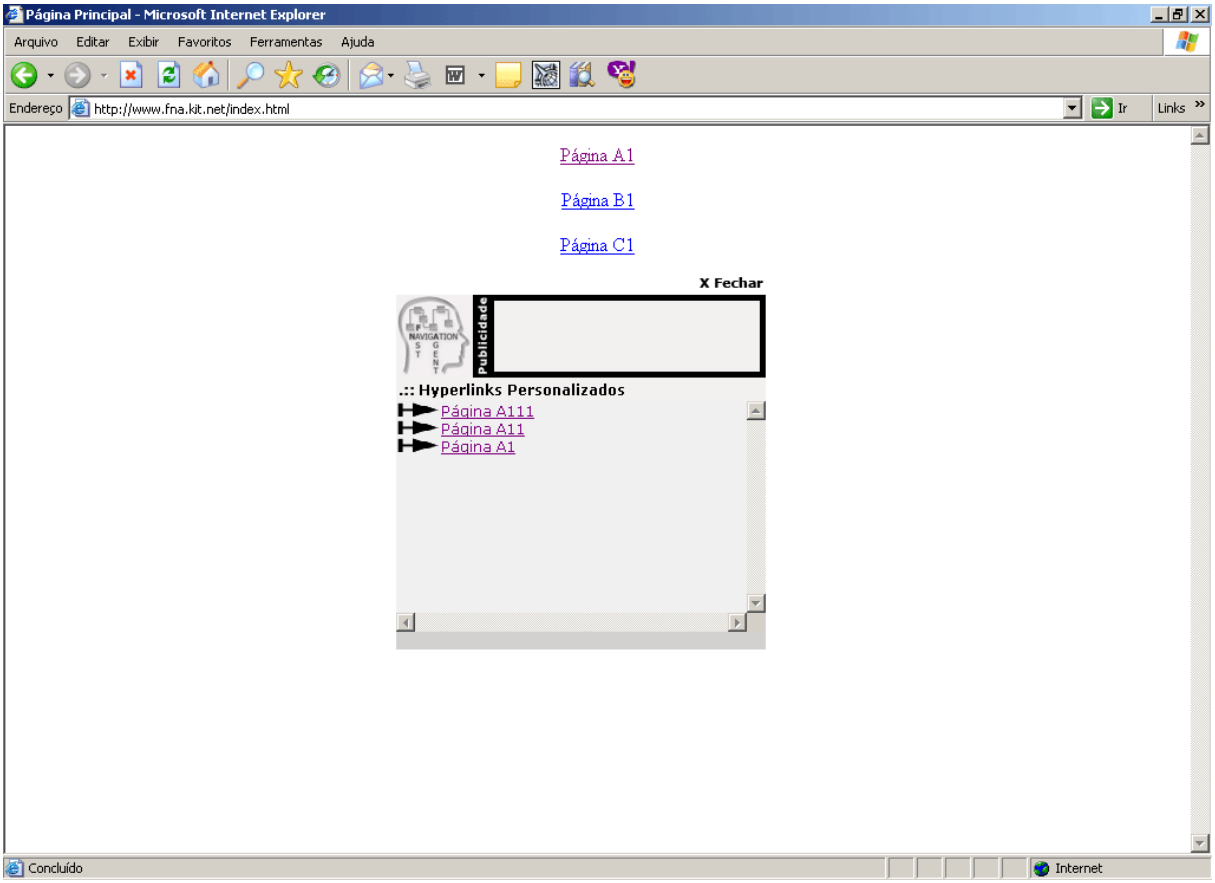

Figura 29 - 1º Teste: Exibição de *hyperlinks* personalizados na página inicial

Nota-se que a partir da página inicial o visitante consegue, após a atuação do agente, acessar diretamente a página *Web* A111.

Vale lembrar que esse teste foi realizado no primeiro acesso do visitante ao *site*-teste. Nesse primeiro acesso o visitante também acessou a página *Web* B11112. A figura 30 ilustra a exibição dos *hyperlinks* das páginas *Web* em que o visitante não tem acesso diretamente da página *Web* B11112, por exemplo, as páginas Web A1, A11, A111 e a página inicial.

Em um segundo acesso ao *site*-teste, o visitante recebe os *hyperlinks* personalizados conforme a figura 31.

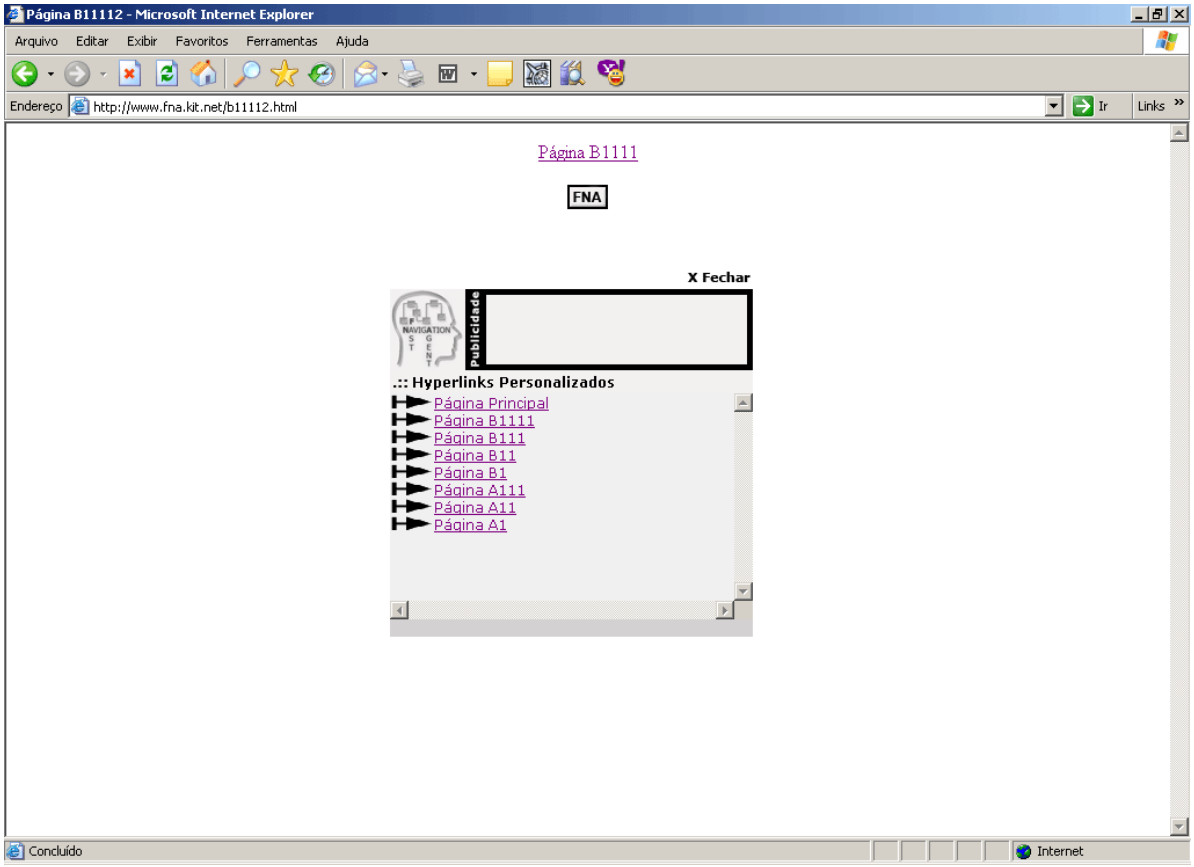

Figura 30 - 1º Teste: Exibição dos *hyperlinks* personalizados a partir da página *Web* B11112

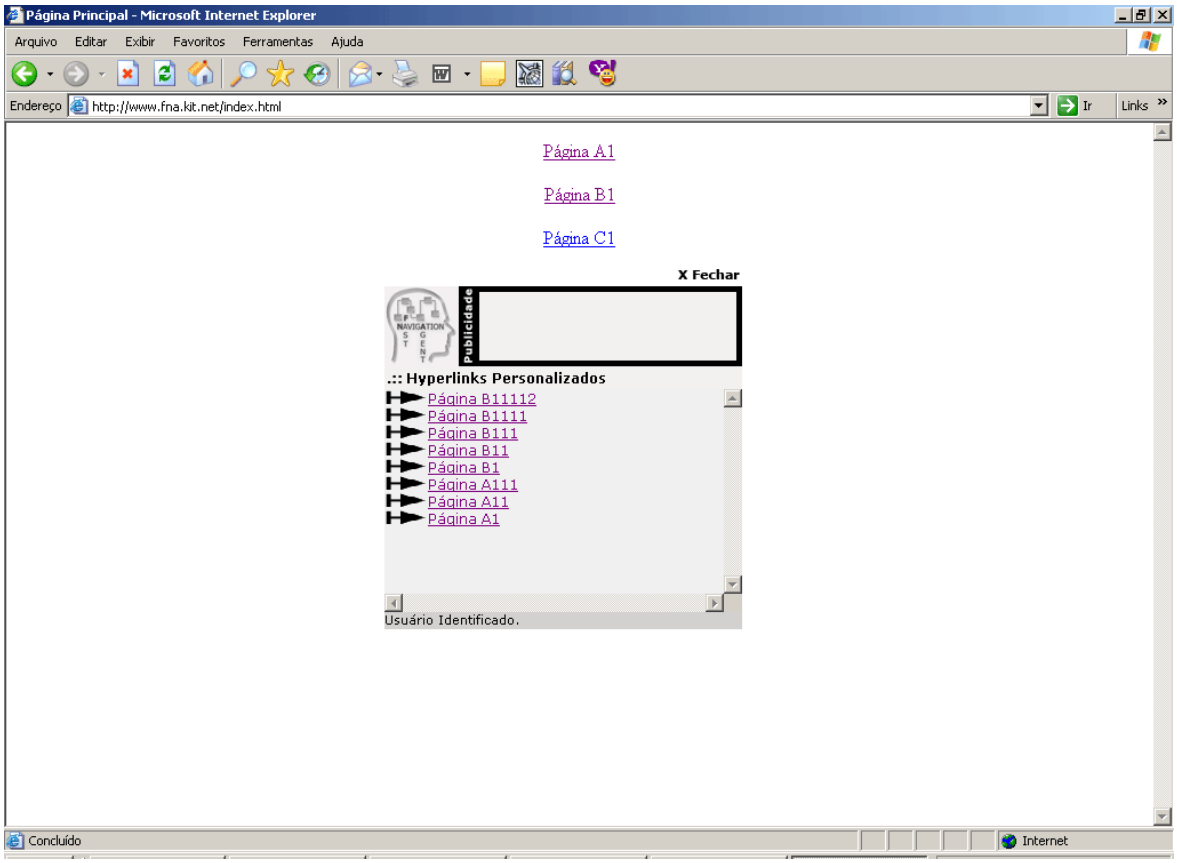

Figura 31 - 1º Teste: Segundo acesso ao *site*-teste

O quadro 14 mostra as URLs com seus respectivos graus de interesse resultante da aplicação da equação (3) para exibição dos *hyperlinks* personalizados na interface do agente.

Titulo da Página	URL	Grau de Interesse
Página Principal	http://www.fna.kit.net/index.html	3.862745
Página B11112	http://www.fna.kit.net/b11112.html	1.876226
Página B1111	http://www.fna.kit.net/b1111.html	1.650735
Página B111	http://www.fna.kit.net/b111.html	1.598039
Página B11	http://www.fna.kit.net/b11.html	1.518382
Página B1	http://www.fna.kit.net/b1.html	1.465686
Página A111	http://www.fna.kit.net/a111.html	1.322304
Página A11	http://www.fna.kit.net/a11.html	1.186274
Página A1	http://www.fna.kit.net/a1.html	1.127451

Quadro 14 - 1º Teste: URLs com os respectivos Graus de Interesse

Na segunda passagem pelo *site* o visitante tem a possibilidade de acessar a página *Web* A111 e a partir desta a página *Web* B11112 (figuras 32 e 33).

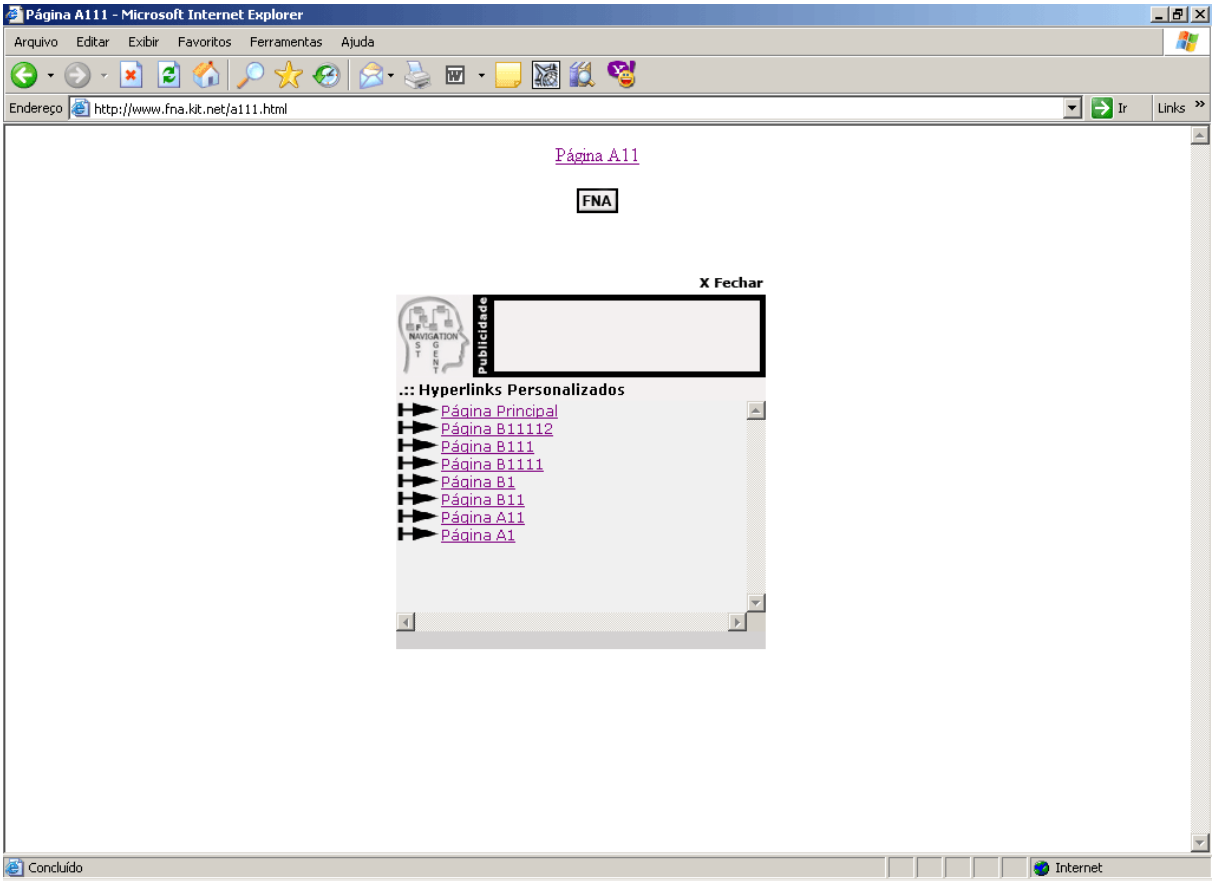

Figura 32 - 1º Teste: Acesso à página *Web* A111 a partir da página inicial

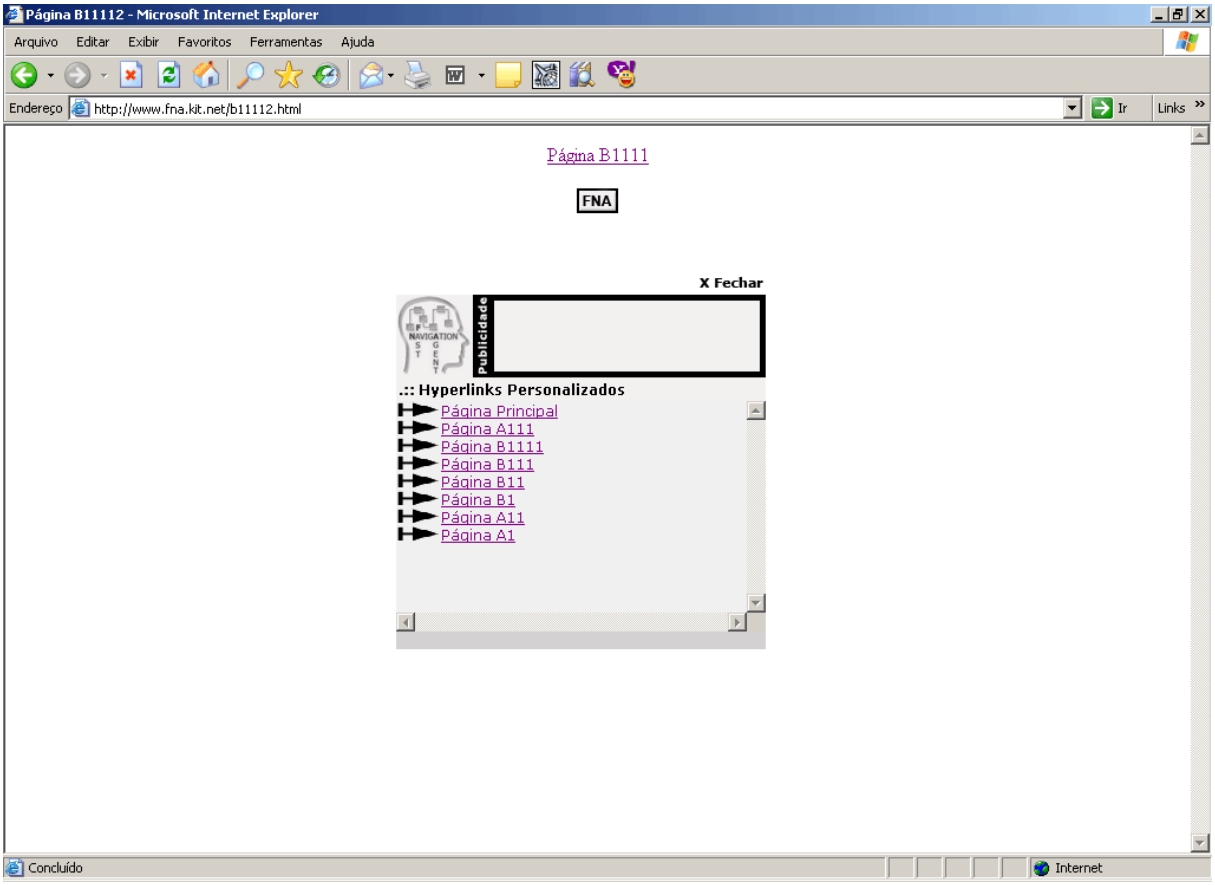

Figura 33 - 1º Teste: Acesso à página *Web* B11112 a partir da página *Web* A111

Observa-se na figura 33 que o *hyperlink* da página *Web* A111 é apresentado em segunda posição. Isso ocorre devido a um dos critérios apresentado na equação (3) no qual o visitante acessa uma página *Web* através do FNA.

A figura 34 mostra as ligações geradas pelo FNA a partir da página *Web* B11112.

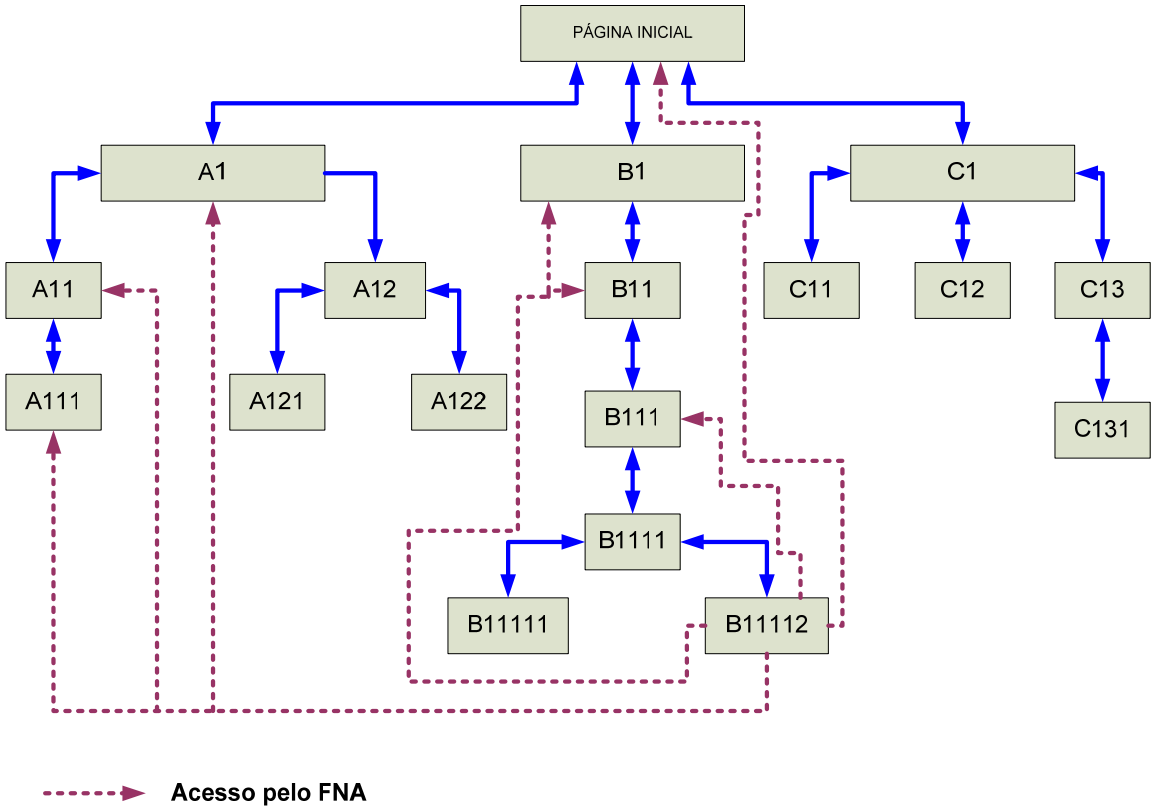

Figura 34 - 1º Teste: Novos *hyperlinks* gerados pelo FNA a partir da página *Web* B11112

4.5.2.2 2º Teste: Utilização das regras de associação geradas

Com base nas duas sessões do visitante registradas no banco de dados (quadro 15) foram geradas as seguintes regras de associação (X ⇒ Y) (quadro 16).

SessaoID	URL
e213ea5a0525ed2e098bbfc629f483ad	http://www.fna.kit.net/**index.html**
e213ea5a0525ed2e098bbfc629f483ad	http://www.fna.kit.net/**a1.html**
e213ea5a0525ed2e098bbfc629f483ad	http://www.fna.kit.net/**a11.html**
e213ea5a0525ed2e098bbfc629f483ad	http://www.fna.kit.net/**a111.html**
e213ea5a0525ed2e098bbfc629f483ad	http://www.fna.kit.net/**index.html**
e213ea5a0525ed2e098bbfc629f483ad	http://www.fna.kit.net/**b1.html**
e213ea5a0525ed2e098bbfc629f483ad	http://www.fna.kit.net/**b11.html**
e213ea5a0525ed2e098bbfc629f483ad	http://www.fna.kit.net/**b111.html**
e213ea5a0525ed2e098bbfc629f483ad	http://www.fna.kit.net/**b1111.html**
e213ea5a0525ed2e098bbfc629f483ad	http://www.fna.kit.net/**b11112.html**
17992be42c429f61bb7fa341ffaf18b0	http://www.fna.kit.net/**index.html**
17992be42c429f61bb7fa341ffaf18b0	http://www.fna.kit.net/**a111.html**
17992be42c429f61bb7fa341ffaf18b0	http://www.fna.kit.net/**b11112.html**

Quadro 15 - 2º Teste: Registros de Acesso ao *site*-teste

Rk	X	Y	Confiança
2	http://www.fna.kit.net/**a111.html**	http://www.fna.kit.net/**b11112.html**	100%
2	http://www.fna.kit.net/**b11112.html**	http://www.fna.kit.net/**a111.html**	100%
2	http://www.fna.kit.net/**a111.html**	http://www.fna.kit.net/**index.html**	100%
2	http://www.fna.kit.net/**index.html**	http://www.fna.kit.net/**a111.html**	100%
2	http://www.fna.kit.net/**b11112.html**	http://www.fna.kit.net/**index.html**	100%
2	http://www.fna.kit.net/**index.html**	http://www.fna.kit.net/**b11112.html**	100%
3	http://www.fna.kit.net/**a111.html**, http://www.fna.kit.net/**b11112.html**	http://www.fna.kit.net/**index.html**	100%
3	http://www.fna.kit.net/**a111.html**	http://www.fna.kit.net/**b11112.html**, http://www.fna.kit.net/**index.html**	100%
3	http://www.fna.kit.net/**b11112.html**	http://www.fna.kit.net/**a111.html**, http://www.fna.kit.net/**index.html**	100%
3	http://www.fna.kit.net/**a111.html**, http://www.fna.kit.net/**index.html**	http://www.fna.kit.net/**b11112.html**	100%
3	http://www.fna.kit.net/**index.html**	http://www.fna.kit.net/**a111.html**, http://www.fna.kit.net/**b11112.html**	100%
3	http://www.fna.kit.net/**b11112.html**, http://www.fna.kit.net/**index.html**	http://www.fna.kit.net/**a111.html**	100%

Quadro 16 - 2º Teste: Regras de associação geradas pelo FNA com confiança mínima de 100%

Com o retorno do visitante identificado ao *site*-teste, o agente consultará o banco de dados em busca de páginas *Web* relacionadas à página inicial. Nesse teste retornará as páginas *Web* A111 e B11112 conforme mostram as figuras 35 e 36.

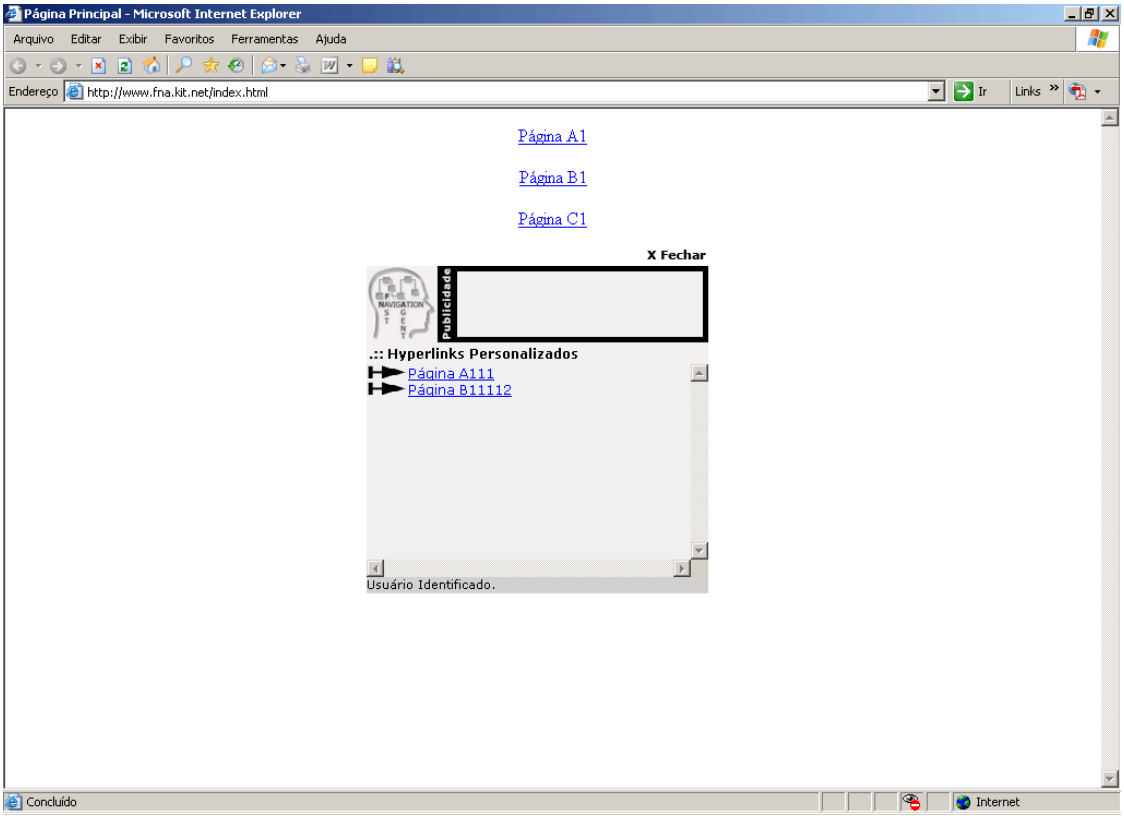

Figura 35 - 2º Teste: *Hyperlinks* personalizados resultantes da geração da regras de associação

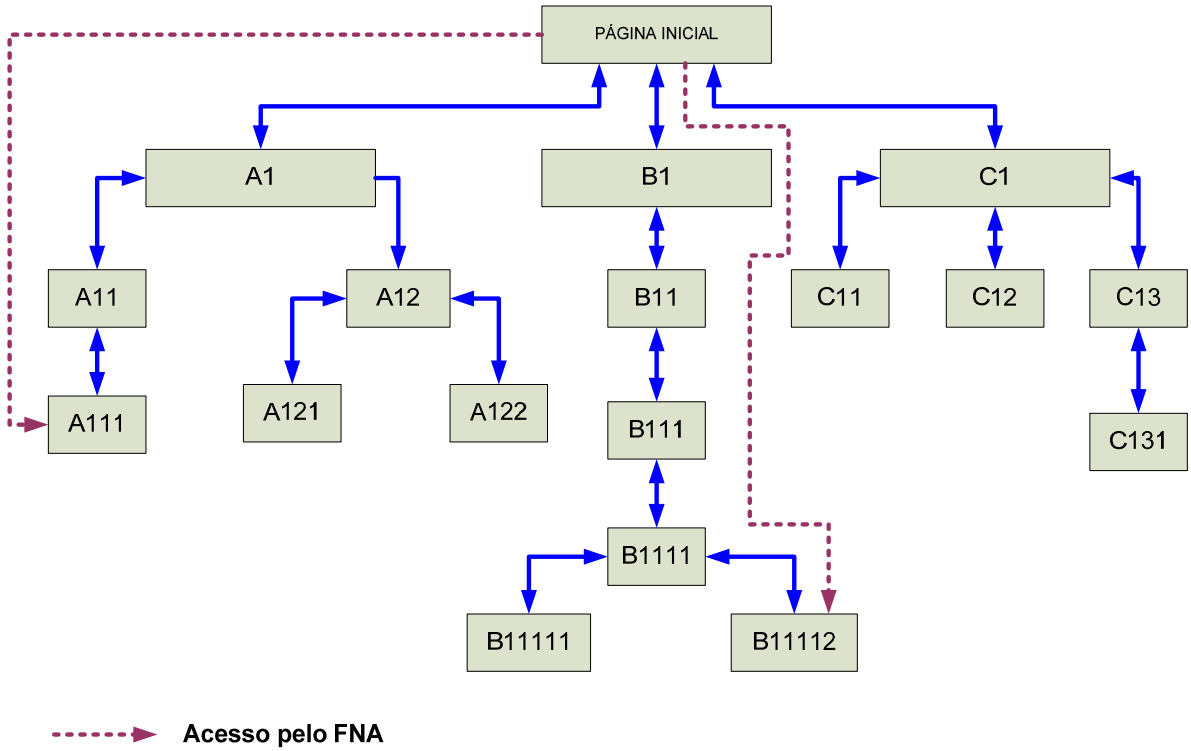

Figura 36 - 2º Teste: Novos *hyperlinks* gerados a partir da página inicial

Com relação às páginas *Web* apresentadas na seção anterior estas foram excluídas devido os seus suportes resultarem em 50% e serem menor que o suporte mínimo calculado de 100%.

Quando o visitante acessa a página *Web* A111, a partir da página inicial, o agente exibe o *hyperlink* da página *Web* B11112 (figuras 37 e 38).

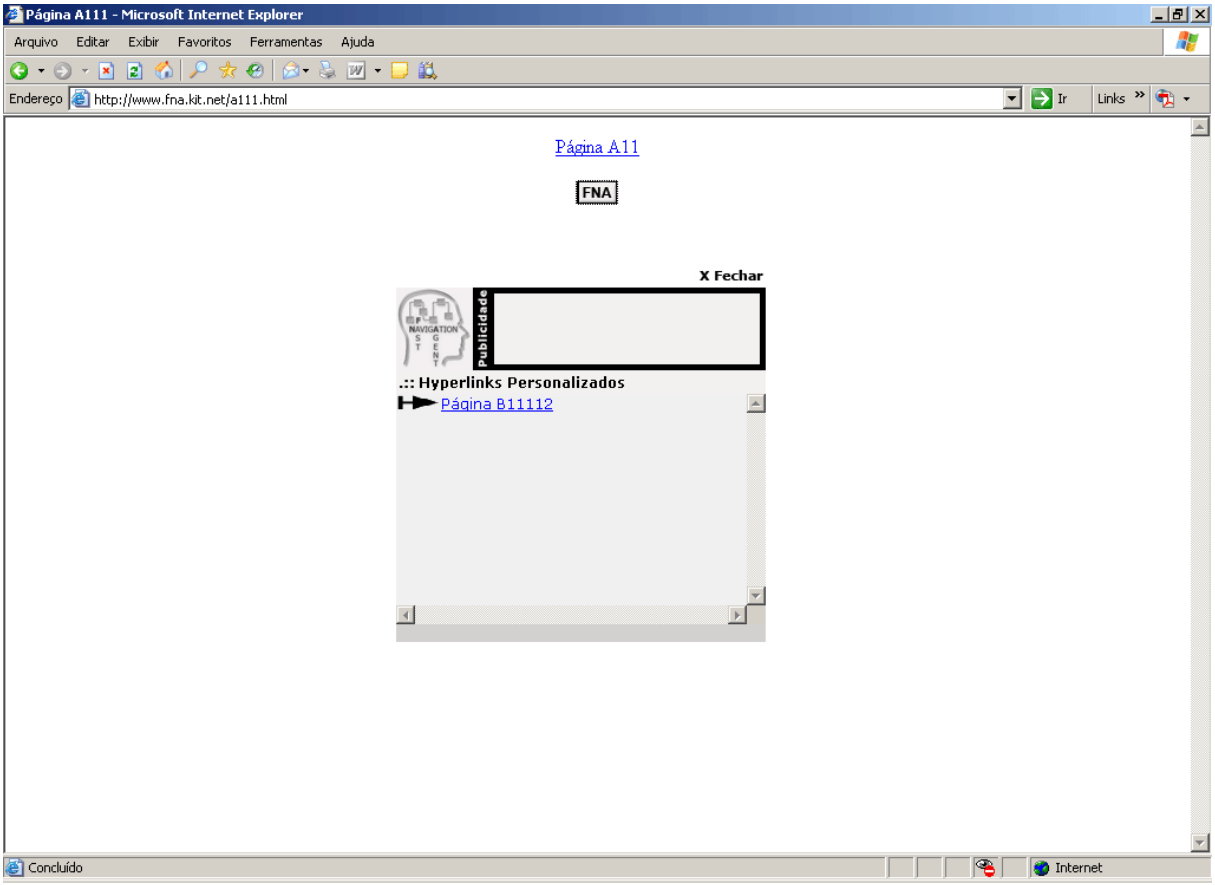

Figura 37 - 2º Teste: Exibição do *hyperlink* da página *Web* B11112 a partir da página *Web* A111

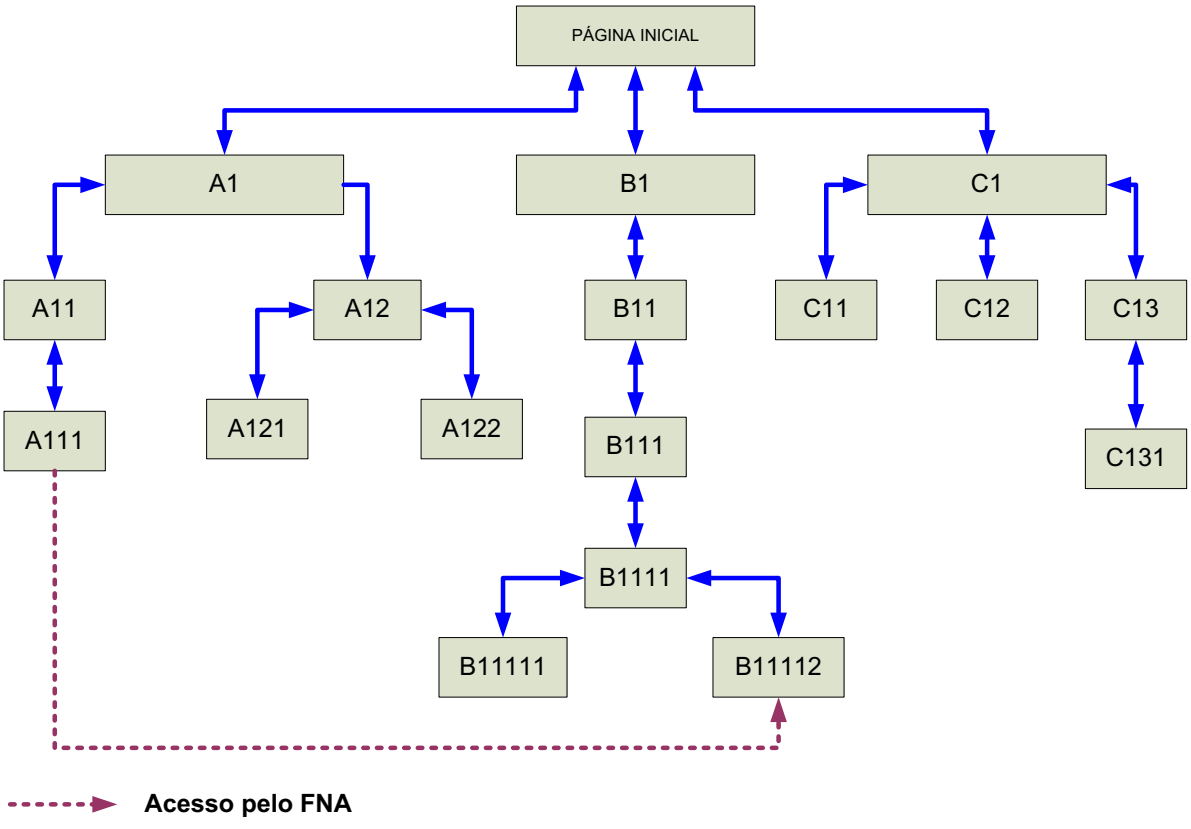

Figura 38 - 2º Teste: Novo hyperlink gerado a partir da página *Web* A111

Quando o visitante acessa a página *Web* B11112 o FNA exibe os *hyperlinks* com base nos graus de interesse das páginas *Web* relacionadas às visitas passadas registradas no banco de dados (figuras 39 e 40) (quadro 17). Isso se deve ao fato de não existir um conseqüente (Y) para o conjunto antecedente (X) {http://www.fna.kit.net/**index.html**, http://www.fna.kit.net/**a111.html**, http://www.fna.kit.net/**b11112.html**}.

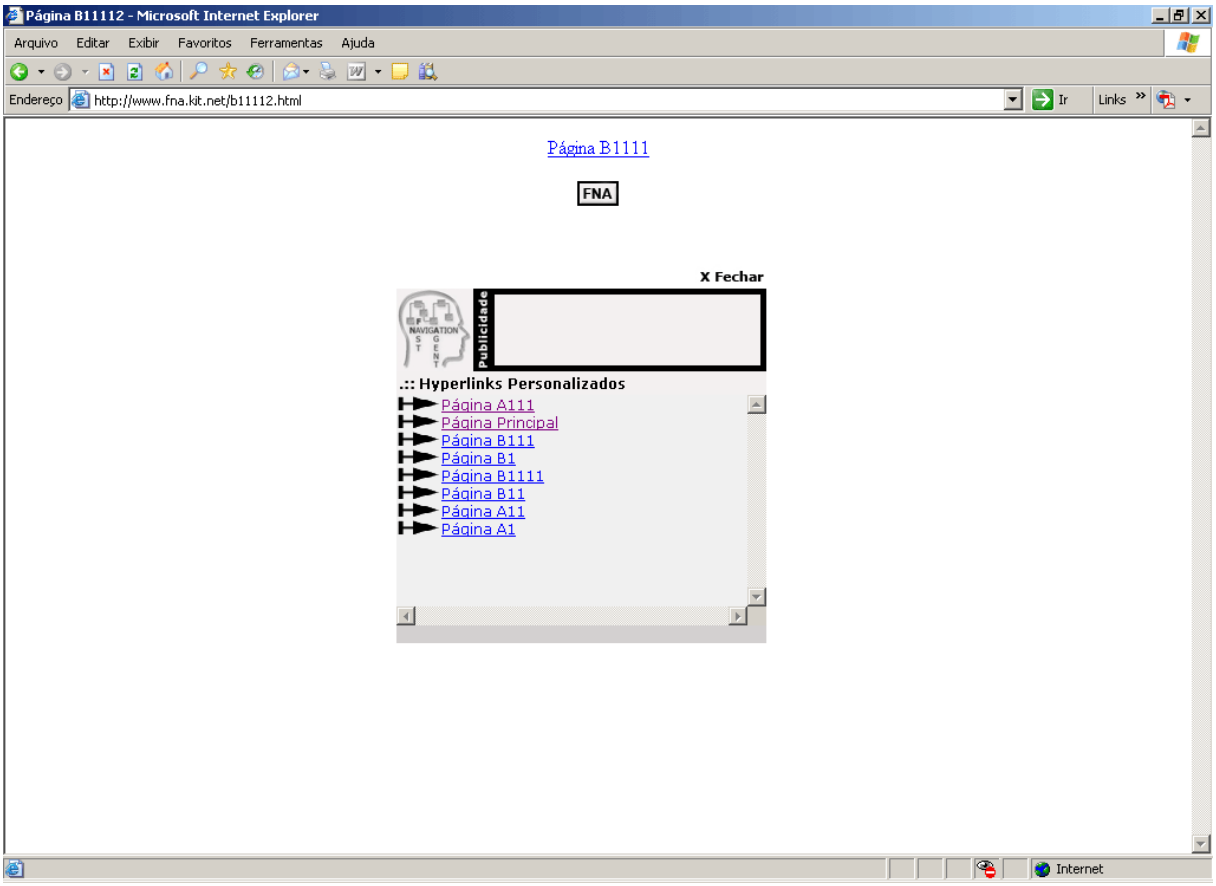

Figura 39 - 2ºTeste: *Hyperlinks* gerados com base no grau de interesse

Titulo da Página	URL	Grau de Interesse
Página A111	http://www.fna.kit.net/a111.html	9,801824
Página Principal	http://www.fna.kit.net/index.html	9,033296
Página B11112	http://www.fna.kit.net/b11112.html	8,475627
Página B111	http://www.fna.kit.net/b111.html	1,005988
Página B1	http://www.fna.kit.net/b1.html	1,005495
Página B1111	http://www.fna.kit.net/b1111.html	1,005246
Página B11	http://www.fna.kit.net/b11.html	1,004752
Página A11	http://www.fna.kit.net/a11.html	1,004453
Página A1	http://www.fna.kit.net/a1.html	1,004234

Quadro 17 - 2º Teste: Graus de interesse das páginas *Web*

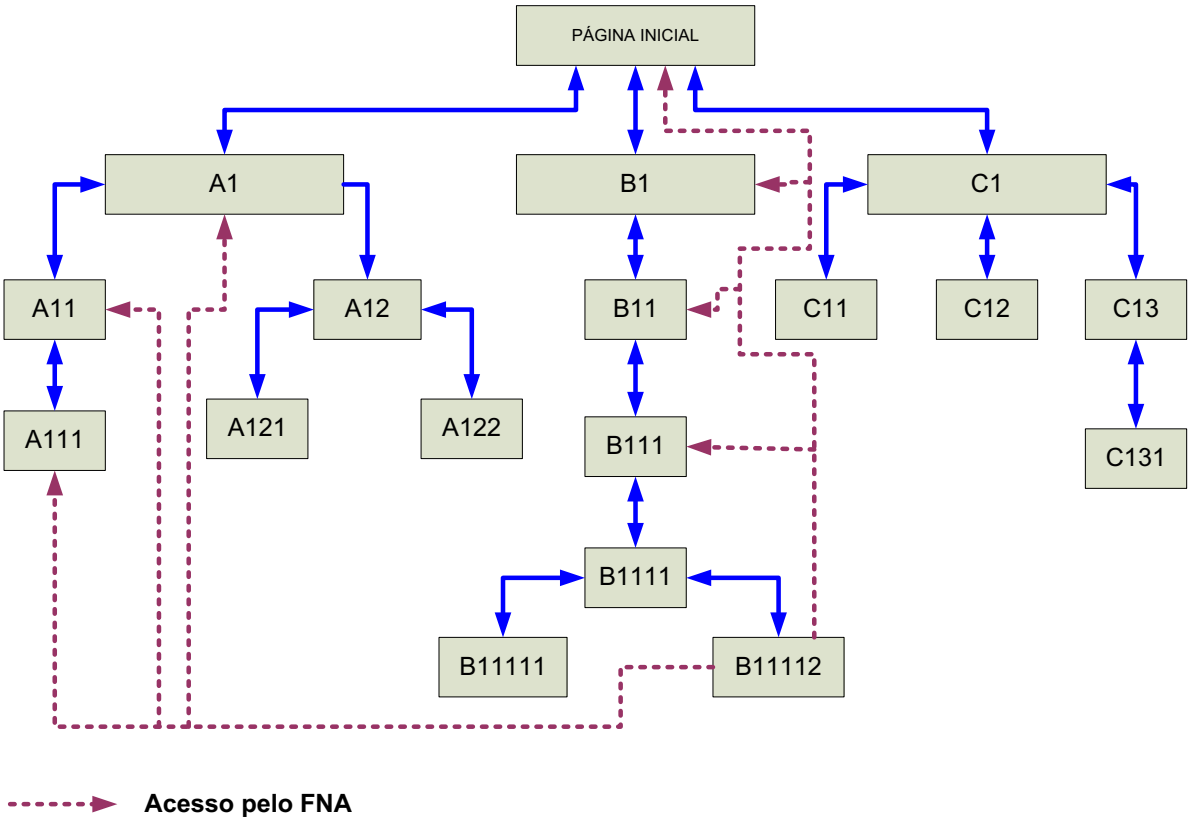

Figura 40 - 2º Teste: Novos *hyperlinks* gerados a partir da página *Web* B11112

4.6 CONCLUSÃO

O *Fast Navigation Agent* auxilia o visitante em sua navegação por um *site* exibindo *hyperlinks* personalizados baseados em seu histórico de navegação pelo mesmo. Estes permitem ao visitante alcançar o seu destino mais rapidamente.

Para o desenvolvedor do *site*, a implementação do agente é simples e não há necessidade de recursos tecnológicos adicionais. O FNA não interfere na estrutura do *site* permitindo que o visitante navegue livremente pelo mesmo e explore novas páginas Web do *site*.

Como o FNA não intefere na estrutura do *site* devido à técnica de Suporte a Navegação Adaptativa com o tipo de adaptação Orientada Direta com *Links* Não Contextuais, os *hyperlinks* personalizados são exibidos em forma de uma lista para o visitante independente do conteúdo das páginas *Web*.

O FNA foi testado nos sistemas operacionais Microsoft Windows XP Profissional e Microsoft Windows 2000 Profissional, com os seguintes navegadores:

a) Microsoft Internet Explorer versão 6;
b) Mozilla Firefox versão 1.5.0.7;
c) Netscape versão 8.1.

Em todos esses ambientes o FNA funcionou perfeitamente sem apresentar problemas de compatibilidade.

5 CONSIDERAÇÕES FINAIS

Esta dissertação apresentou a arquitetura e o funcionamento do agente *Fast Navigation Agent*. O agente tem a finalidade de observar e auxiliar os visitantes em sua navegação pelos *sites* que o utilizam, buscando oferecer, aos seus visitantes, *hyperlinks* personalizados.

O processo de geração dos *hyperlinks* personalizados é realizado pela aplicação da tarefa de regras de associação e, também, da determinação de grau de interesse nas páginas *Web* acessadas. O resultado desse processo possibilita que o visitante chegue ao seu destino rapidamente, reduzindo ou eliminando a navegação por páginas *Web* que não são de seu interesse.

A mineração de dados na *Web* permite extrair conhecimentos úteis e valiosos a partir de um ambiente heterogêneo no qual as informações são armazenadas de forma semi-estruturada. Tais conhecimentos contribuem para que os visitantes consigam interagir melhor com os *sites* e desenvolvedores compreendam melhor o comportamento dos visitantes.

A princípio esta dissertação seguiu a abordagem baseada em conteúdo com a finalidade de conhecer o perfil do visitante individualmente com base em suas navegações passadas.

Finalmente, espera-se que esta pesquisa sirva como base a trabalhos futuros que serão desenvolvidos a partir de implementações de sistemas empregando outros métodos de mineração de dados seguindo a abordagem colaborativa e, assim possibilitando estudos comparativos sempre com a finalidade de obter resultados positivos tanto para os desenvolvedores de *sites*, quanto para os visitantes.

REFERÊNCIAS

AGRAWAL, R., SRIKANT, R. **Fast Algorithms for Mining Associations Rules**. Proceedings of 20th International Conference on Very Large Data Bases, p. 487 – 499, Santiago, 1994.

BRUSILOVSKY, P. **Adaptive Hypermedia.** User Modeling and User-Adapted Interaction, v. 11, n. 1-2, p 87-110, 2001.

BRUSILOVSKY, P. **Methods and Techniques of Adaptive Hypermedia.** User Modeling and User-Adapted Interaction, v. 6, n. 2-3, p 87-129, 1996.

BRUSSO, M. J. **Access Miner: Uma proposta para a extração de regras de associação aplicada à mineração do uso da Web**. 2000. 96 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal do Rio Grande do Sul, Rio Grande do Sul.

CHAN, P. K. **Constructing Web User Profiles. A Non-Invasive Learning Approach**. In: MASAND, B.; SPILIOPOULOU, M. Web Usage Analysis and User Profiling - International WEBKDD´99 Workshop, p. 39-55. Berlim: Springer-Verlag, 2000.

CHANG, G.; HEALEY, M.J.; MCHUGH, J. A. M.; WANG, J.T.L. **Mining the World Wide Web An Information Search Approach**. p. 93-104. Massachusetts: Kluwer Academic Publishers, 2001.

COOLEY, R.; MOBASHER, B; SRIVASTAVA, J. **Web Mining: Information and Pattern Discovery on the World Wide Web**. Proceedings of the 9th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'97), p.558. Newport Beach: IEEE, 1997.

EBECKEN, N. F.; LOPES, M. C. S.; COSTA, M. C. A. **Mineração de Textos**. In: REZENDE, S. O. Sistemas Inteligentes - Fundamentos e Aplicações (Rede Cooperativa de Pesquisa em Inteligência Artificial), p. 364-369. Barueri: Manole, 2003.

ETZIONI, O. **The World Wide Web: quagmire or gold mine?** Communication of rhe ACM, v. 39, n. 11, p. 65 – 68, Novembro 1996.

FAYYAD, U; SHAPIRO, G. P; SMYTH, P. **From data mining to knowledge discovery: An overview**. In Advances in Knowledge Discovery and Data Mining, U. FAYYAD, U; SHAPIRO, G. P; SMYTH, P.; UTHURUSAMY, R. (eds). Cambridge: AAAI/MIT Press, 1996a.

FAYYAD, U; SHAPIRO, G. P; SMYTH, P. **The KDD Process for Exracting Useful Knowledge from Volumes of Data**. Communication of the ACM, v. 39, n. 11, p. 27 – 34, Novembro 1996b.

HAN, J.; KAMBER, M. **Data Mining - Concepts and Techniques.** p. 225, San Diego: Academic Press, 2001.

FRANKLIN, S.; GRAESSER, A. **Is it an Agent, or just a Program ?: A Taxonomy for Autonomous Agents**. Proceedings of the Third International Workshop on Agent Theories, Architectures and Languages, Springer-Verlag, 1996.

LIEBERMAN, H. **Autonomous Interface Agents**. Proceedings of the SIGCHI conference on Human factors in computing systems, p. 67-74. New York: ACM press, 1997.

PAZZANI, M. J.; BILLSUS, D. **Adaptive Web site Agents**. Proceedings of the third annual conference on Autonomous Agents, p. 394-395. New York: ACM press, 1999.

ROSATELLI, M. C.; TEDESCO, P. A. **Diagnosticando o usuário para criação de sistemas personalizáveis**. In R. O. Anido & P. C. Masiero (Eds.), Anais do XXIII Congresso da SBC - III Jornada de MCIA, v. 8, p. 153-201. Porto Alegre: SBC, 2003.

RUSSELL, S. J.; NORVIG, P. **Artificial Intelligence: A Modern Approach**. 1. ed. New Jersey: Prentice Hall, 1995.

SRIVASTAVA, J.; COOLEY, R.; DESHPANDE, M.; TAN, P. **Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data**. ACM Special Interest Group on Knowledge Discovery and Data Mining, v.1, n.2, p.12 - 23, Janeiro 2000.

YAO, Y. Y.; HAMILTON, H. J.; WANG, X. **PagePrompter: An Intelligent Web Agent Created Using Data Mining Techniques**. Lecture Notes in Computer Science, v. 2475/2002, p. 506-513. Berlin: Springer, 2002.

ZAIANE, O. R. **Web Mining: Concepts, Practices and Research**. XIV Brazilian Symposium on Databases (SBBD'2000), p. 410-474. Joâo Pessoa: SBC, 2000.

ZUKERMAN, I.; ALBRECHT, D.W. **Predictive Statistical Models for User Modeling.** User Modeling and User-Adapted Interaction, v. 11, n. 1-2, p 5-18, Março 2001.

ANEXO A - MODELO ENTIDADE-RELACIONAMENTO DO FNA